Packet Priority Assignment for Wireless Control Systems of Multiple Physical Systems

Wenchen Wang, Daniel Mosse
Computer Science Department
University of Pittsburgh
Pittsburgh, PA, USA

Alessandro V. Papadopoulos
IDT, Mälardalen University
Västerås, Sweden

***Abstract*—Wireless control systems (WCSs) have gained much attention lately, due to their easy deployment and flexibility compared to wired control systems. However, this comes at the cost of possibly increased network delay and packet losses, that can significantly impact the control system performance, and possibly its stability. Such problems become even more relevant if the network is shared among different control systems, and thus becomes a scarce resource, like in Industrial Internet of Things applications. In this paper, we describe how to assign packet priorities dynamically when there are many physical systems sharing a given network, aiming at minimizing the performance degradation of the WCS. Towards that, we present a network model including both delay and packet losses, both of which are very important for the control system performance. Our solution is evaluated over two different use cases to show the generality of the approach: the WCS for a set of inverted pendula, and the WCS for small modular reactors in a nuclear power plant. The results show that the proposed approach allows for a more stable performance even in presence of highly nonlinear systems, sensitive to time-varying delays, as well as in presence of high network interference.**

I. Introduction

Networked and wireless control systems (WCSs) have been a major topic of research across different communities in the last decade [1]–[4]. A WCS comprises controllers, sensors, relay nodes, and actuators connected via a wireless network. Such a deployment poses significant challenges in the control system and in the implementation of suitable protocols that allow for a predictable behavior of the controlled system, in particular in multi-hop deployments with shared resources [5]–[8]. In fact, the network-induced imperfections, such as network delay and packet losses degrade the control system performance, especially during transients of the physical system, and can even impact the control system stability.

Prior research [6], [8]–[10] focused on the impact of the WCSs when controlling a single physical system, often linear and stable [**?**], [11], [12]. In this work, we focus on the case of wireless real-time control with *multiple physical nonlinear systems*, with a given fixed multi-hop network. The increasing number of IoT (Internet of Things) and IIoT (Industrial IoT) systems calls for new approaches to design and implement WCSs for multiple physical systems [13]–[16]. The main challenges rely in the preservation of the stability of the multiple physical systems, as well as the design of suitable protocols that can support the real-time operation.

To reduce the effect of network-induced imperfections on the overall control system RMSE (root mean square error), we propose an approach to dynamically assign priorities to packets of the different physical systems, choosing the appropriate network paths. To show the generality of our approach, we evaluated the effect of the proposed network control over two very distinct control systems of multiple physical systems: (a) a set of inverted pendula, and (b) a set of of small modular reactors (SMRs) in a nuclear power plant.

The contributions of this paper are the following:

- We propose a mechanism to address for the problem of controlling multiple physical systems over a shared wireless multi-hop network;
- We propose three heuristic cross-layer methods for the packet priority assignment problem for minimizing the effect of delays and packet losses on the different control system performance;
- We present a general network quality model for WCSs, that includes both end-to-end network delay and packet loss, to choose on which route to send packets of different priorities; and
- We evaluate the proposed approach measuring the RMSE of our approaches for two use cases of WCSs sharing a multi-hop wireless network: a set of inverted pendula, and a set of SMRs in a nuclear power plant.

II. Related Work

Although WCSs have several advantages, including an easy deployment and maintenance, one of the biggest challenges is dealing with network-induced imperfections [17]. The solutions of recent research works are typically divided into three categories: (i) control only, (ii) network only, and (iii) control+network co-design solutions.

Control solutions for dealing with network imperfections are promising. In [18], the closed-loop system is modeled and controlled as a switched system, considering both time delays and packet losses at the actuator nodes, and it is sabilized by using an optimal control approach. Other examples include [19]–[21] that use a model predictive control approach, which obtains a

This work was supported by the US Department of Energy, by the US National Science Foundation, and by the Swedish Foundation for Strategic Research under the project "Future factories in the cloud (FiC)" with grant number GMT14-0032.

finite number of future control commands besides the current one for handling both time-varying delays and packet drops. However, these works only consider network as a black box and there is no packet scheduling mechanism taking into account different control system application demands, and they typically focus on a single physical plant to be controlled.

For the network solutions, online dynamic link layer scheduling algorithms have been proposed [22]–[24] to meet the deadline of a rhythmic flow and minimize the number of dropped regular packets in a centralized and distributed way, respectively, based on the rhythmic task model proposed in [25]. However, these two works did not consider different control system application demands (i.e., an SMR may need to increase the power output from 40MW to 43MW within 10 minutes). Also, they assume network external disturbances occur sporadically, which is different from the problem addressed herein. Han et al. [5] propose three types of reliable routing graphs for different communication purposes and generate real-time data link layer communication schedules based on those graphs. Saifullah et al. [26], [27] analyze the worst-case end-to-end delay analysis for source and graph routing based on wirelessHart standard to guarantee the real-time communication in WCS. Alderisi et al. [28] propose a probabilistic scheduling method to provide guarantees on reliable packet delivery in wirelessHART based networks, and they do not account for the current demands of the control systems. In this paper we do not focus on real-time network delay analysis nor on proposing new network communication scheduling, since we assume there is a solid network design that contains different routing paths to transmit several periodic control system packets in parallel. Instead, in this paper we use the network quality model proposed in this paper to choose which routing path is the best for which packet considering control system application demands.

For the co-design solution, which is the closest to ours, the integration of wireless network and control systems performance are studied in [6], [8], [9], [29], [30], [30]–[33]. The co-design of fault-tolerant wireless network and control in nuclear power plants are studied in [8], [31]–[33]. The work in [8] shows that the network delay and reliability both could affect the control system performance. Network reconfiguration schemes to minimize the network-induced error for WCS with a single control system are proposed in [33]. In [6], the authors show how the network reliability affects the control system failure ratio via a water tank case study. In [9], the authors discuss how the routing scheme affects the control system performance. A co-design of network topology conditions and control system stability is explored in [7]. In [29], the author derives a sufficient condition for the random access communication policy of shared wireless medium and design a control-aware random access communication policy. There are important aspects that are not fully covered by these approaches, that are addressed in this paper. First, there is still a gap to describe the relationship between network performance and control system performance, especially when the system to be controlled is nonlinear. There is a model to describe this gap, and thus we propose a general network quality model to describe this gap in terms of network delay and message loss. Second, all these works discuss wireless control system with single physical system, while our approach deals with multiple physical systems. Finally, in this paper we more deeply analyze the interplay between dynamic packet scheduling and the control system performance.

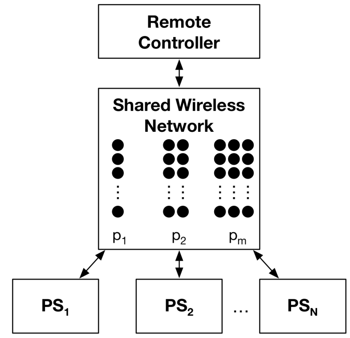


Fig. 1: System overview: N physical systems, are connected to a remote controller, via a shared wireless network.

III. Problem Formulation

Consider N physical systems (PSs) that share one wireless network as shown in Figure 1. We define a series of time steps $t = \{t_0, t_1, \ldots, t_n\}$ and a set of N reference functions $\mathbf{r}(t) = \{r_1(t), r_2(t), \ldots, r_N(t)\}$. Each reference function corresponds to one PS. Reference functions define the desired behavior of the different PSs over time, such as temperature profiles, trajectories to follow, or other similar setpoint changes over different time frames. For simplicity of presentation we consider ramp-like reference signals (linearly increasing or decreasing over time, up to a desired value), that can be characterized by: (i) Requested Change Amount (RCA), or the amount a reference quantity must increase/decrease in a linear way, (ii) Requested Change Duration (RCD), or the amount of time in which such change must take place, and (iii) Start Time (ST), or the initial time instant when such change should start. For example, RCA=1, RCD=10, and ST=5 means that the controlled variable of the control system must increase linearly by 1 unit, within 10 seconds, starting from time 5 seconds.

Each PS periodically sends out one packet of its measurements to the remote controller at a given frequency. We assume that the sensors measuring the PSs performance implement a time synchronization protocol, such as [34], [35], and therefore they share a common time. As in [36], there are m choices of network paths/flows $\{p_1, p_2, \ldots, p_m\}$, each of which has a delay D_j, time given that we consider TDMA networks with the fixed topology (i.e., the number of nodes in a path does not change) and a time-varying delivery ratio $dr_j(t)$, due to possible network noise and interference (the variation of network noise makes delivery ratio vary over time). Each network path delivers the measurements of one physical system to the remote controller at a time. We assume that if a measurement is lost, the controller will use the last received value for computing the control law. Approaches to optimize the control system behavior in presence of delayed or lost messages have been studied [37]–[39], but this issue is beyond the scope of this paper.

Our objective is to minimize the performance degradation induced by the wireless realization of the network, with respect to a wired realization, over a time horizon T_{trans}, when physical systems are experiencing a *transient* induced by a change in their respective reference function $\mathbf{r}(t)$. As usual, for each PS $i \in \{1, \dots, N\}$, the Root Mean Squared Error (RMSE) metric is computed as:

$$RMSE_i = \sqrt{\frac{1}{T_{trans}} \sum_{t=0}^{T_{trans}} \|y_i^W(t) - y_i^{WL}(t)\|^2} \quad (1)$$

where $y_i^W(t)$ is the output vector of system i with a wired network, $y_i^{WL}(t)$ is the output vector of system i with a wireless network, and $\|x\|$ is the 2-norm of the vector x.

The overall quality is assessed as the average performance degradation over the N physical systems, computed as

$$RMSE_{\text{tot}} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} RMSE_i^2} \quad (2)$$

which must be minimized. In the case of heterogeneous physical quantities, a normalized version of the $RMSE_{\text{tot}}$ can be considered.

IV. Solution Approach

Our approach to minimize the $RMSE_{\text{tot}}$ is to create a dynamic priority assignment for the packets and to send these packets through a certain chosen path in the shared wireless network with multiple paths. Each path has different quality levels, based on its reliability and its induced delay (the path quality is formally defined in Section IV-B). The proposed solution is composed of two steps. First, the systems are sorted by decreasing application demand at that particular time, and a priority is assigned accordingly. Then, a mapping between the priority of the control system and the path quality is created, that is, the packets associated with the PS of the highest priority will be sent over the network path with the best quality. In the case of heterogeneous PSs, all the quantities can be normalized between the minimum and maximum demand.

Due to the combinatorial complexity of the problem, an optimal solution cannot be identified unless a brute-force approach is used. Therefore, we propose three *heuristic methods to dynamically assign the priorities* to the N physical systems (Section IV-A), and we propose a *path quality model* (aptly called the *PQModel*) that includes delay and loss to quantify the quality of network path.

A. Priority Assignment of Measurement Packets

The basic idea is to give high priority to measurement packets of the PS that would yield poor performance if its packets were delayed or lost, that is, to avoid unnecessarily increases in RMSE of each PS, and thus of $RMSE_{\text{tot}}$. To determine the packet priority, we propose three heuristic methods with different perspectives.

1) Dynamic Heuristic: To minimize $RMSE_{\text{tot}}$, this heuristic gives the higher priority to the PS with higher RMSE, because it is more necessary for that PS to transmit its message as soon and as reliably as possible (thus reducing the RMSE). Since we cannot get the RMSE of the wired control system at run time, we track the $rRMSE_i(t)$ for the i^{th} PS compared with its reference function $r_i(t)$ for PS$_i$ at run time at each time step, which is computed as:

$$rRMSE_i(t) = \sqrt{\frac{1}{t} \sum_{j=0}^{t} \|r_i(j) - y_i(j)\|^2}, \quad (3)$$

where $y_i(j)$ is the measured power output for PS$_i$ at time j. At each time step, we calculate $rRMSE_i(t)$, sort the rRMSEs of N PSs and assign the highest priority to the measurement packet of the PS with the highest current rRMSE.

2) Static Heuristic: The static approach carries out a thorough offline analysis for all possible parameters, computing $rRMSE_i$ with no message loss for each PS, and assigns priorities before the system starts executing. Priorities are not altered during run-time.

3) PID Dynamic Heuristic: The third heuristic is inspired by the widely used PID controller. In classical PID control [40], the error is defined as

$$e_i(t) = r_i(t) - y_i(t). \quad (4)$$

We define the tracking error for each PS$_i$ as:

$$e_i(t) = |r_i(t) - y_i(t)|. \quad (5)$$

The priority π_i for PS$_i$ follows the PID-like dynamics

$$\pi_i(t) = K_P \ e_i(t) + K_I \sum_{i=1}^{t} e_i(t) + K_D \ (e_i(t) - e_i(t-1)).$$

The first term is the Proportional term (or P-term) and it tracks the latest/current error. The second term is the integral term (or I-term) and it tracks (cumulative) error over time. The last term is the derivative term (or D-term) and it approximates the trend of error in the future (e.g., if this term is negative, it means the tracking error tends to reduce).

The reason to introduce the modified expression of the tracking error as in Equation (5) comes from the fact that a discrepancy between the reference signal r and the system output y must cause the priority to always increase, independently of the sign of such discrepancy. With a classical tracking error as in Equation (4), a discrepancy such that $r < y$ would cause the priority to *decrease*, which is not the objective of the priority assignment mechanism. As a result of such choice, the integral term is a non-decreasing function of time, and cannot forget the past errors in presence of positive errors. Therefore, to tamper the magnitude of the integral term, we select $K_I = \lambda K_P / t$, which divides the value by the timeframe being considered, obtaining the following expression:

$$\pi_i(t) = K_P e_i(t) + \frac{K_P \lambda}{t} \sum_{i=1}^{t} e_i(t) + K_D \left(e_i(t) - e_i(t-1)\right) \quad (6)$$

where the new integral term can be interpreted as the average of the error over time and the tuning parameters are K_P, λ and K_D. Finally, we assign highest priority to the measurement of the physical system with highest $\pi_i(t)$ value at time t.

This heuristic makes the change in the priorities calculation more smooth, and filters potential oscillations in the priority assignment that can occur during the PSs' transients.

B. Network Path Quality Determination

After we determine the priority of the measurement packets, we need to determine on which network path to transmit those measurements. Although previous research discussed how the network reliability and network delay affect the control system performance [6], [8], there is still a gap between network performance (i.e., network delay and message loss) and control system performance for problems with several physical nonlinear systems. Since there is no such a model to describe the gap, we propose a general network quality model. We call it PQModel, include both network delay and losses for a path[1], as described by (7), and quantify how much network imperfections affect the control system.

$$PQ = D_{net} + \alpha n_{loss} \Delta_{csp} \quad (7)$$

where D_{net} is the network end-to-end delay, obtained, e.g., as per [41], [42], Δ_{csp} is the control sampling period (time interval between measurements of the PS), α is a non-negative constant, and n_{loss} is the number of consecutive packet losses. Note that n_{loss} is computed based on the control system perspective, in this case Δ_{csp}. For example, if the network sensing sampling period is 0.5s and $\Delta_{csp} = 1.0$s, the network will send two messages during each Δ_{csp}; if the remote controller did not receive either of the messages, $n_{loss} = 1$. Note that this PQModel quantifies the network imperfection impact to the control system, thus the less the PQ value, the better quality the network.

We use α to adjust the importance between network delay and network reliability. When $\alpha = 1$, network delay and network reliability have the same importance to the control system performance. For example, as shown in Figure 2, the control sampling period and network sampling period are both 0.1s, but when the network delay is 0.2s and measurement M_2 gets lost, PQ_2 is 0.3s because the controller will use measurement M_1 that arrived 0.3s earlier. If measurement M_3 also gets lost, the induced delay PQ_3 becomes 0.4s because the controller will (re-)use measurement M_1.

Parameter α must be tuned according to different control system we are dealing with. When $D_{net} < \Delta_{csp}$, that is, the network delay is smaller than the control system sampling period (e.g., like the water tank system in [6]), α is set to a

[1]We omit the subscripts when no confusion arises.

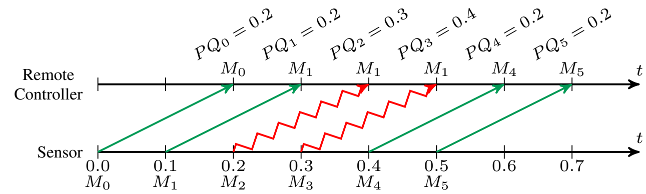


Fig. 2: Network delay and delivery ratio tradeoff illustration, when network delay is greater than control sampling period.

very large number since network reliability is the only factor to affect the control system performance. When $D_{net} \geq \Delta_{csp}$, we set α close to 1. For instance, when the control system uses Kalman filters or any other technique to compensate the message loss reliably, we can reduce the network reliability importance and set α to be small. The value of α also needs to be adjusted under different network situations for the same control system. We will discuss the value of α under different network situations later in Section VI.

V. Case Studies

To make the evaluation of the presented approach uniform, we consider two different case studies that share the same architecture (as shown in Figure 1). In particular, we consider the case of $N = 3$ physical systems and of $m = 3$ paths with 6 hops each. The WCSs considered include 3 inverted pendula (IP) and 3 SMRs of a Nuclear Power Plant (NPP).

For the wireless network, we use the bitvector protocol [43], which uses TDMA scheduling to guarantee real-time transmission with no contention among WCSs within each time slot. We consider the messages sent from sensors to the controller (measurements) and back from controller to the actuator (control signals). We consider a wireless network with three independent paths, each of 6 hops: path 1 (p_1) has no backups but the fastest delivery of packets due to no redundancy; path 2 (p_2) has double the number of sensors and thus higher reliability and higher delay, given the messages have to traverse twice as many nodes; path 3 (p_3) has 3 times as many nodes as p_1, with the highest reliability and highest delay. In other words, the reliability relationship of the three paths is $dr(p_1) < dr(p_2) < dr(p_3)$. We assume each network path can transmit messages independently from the others, that is, all 3 paths can transmit messages in parallel, without interfering with each other (e.g., different channels or physically distant paths). The time slot of TDMA scheduling (Δt) for the IP and NPP are different due to different sensitivity of time delay of the two systems. In particular, $\Delta t = 1$ms for the IP and the delay of p_1, p_2 and p_3 is 0.01s, 0.02s and 0.054s respectively; and $\Delta t = 10$ms for the NPP, and the delay of p_1, p_2 and p_3 is 0.1s, 0.2s and 0.54s respectively. As shown in Figure 3, we combined a state-of-the-art cyber-physical system simulator (WCPS 2.0 [6]) with a Simulink model of the physical system and of its controller. Our simulator allows m wireless network paths running together with m PSs (dark blue block). We implemented the three heuristic

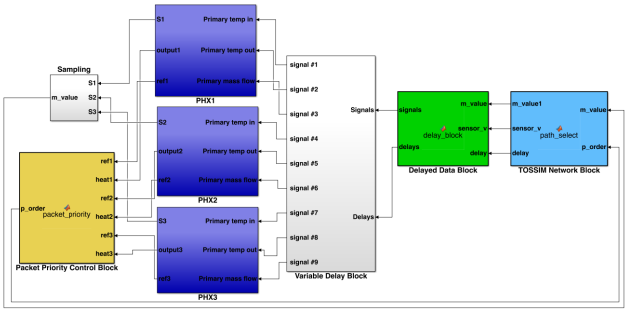


Fig. 3: Simulation overview in Simulink, implemented in conjunction with WCPS for the NPP use case. The Simulink architecture is analogous for the IP use case.

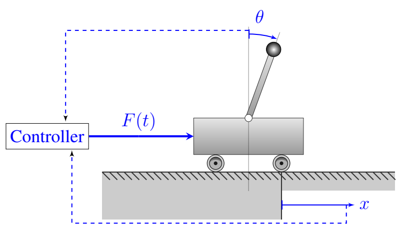


Fig. 4: Inverted pendulum.

methods proposed in Section IV-A to assign priority to the measurement packets in the left-bottom block (yellow). We also implement the network quality model from Section IV-B to quantify the quality of network paths in the rightmost block. We use the TOSSIM network simulator (embedded in WCPS) with wireless traces from a 21-node subset of the WUSTL Testbed[2]. To evaluate the WCSs under a wide range of wireless conditions, similar to [8], we use controlled Received Signal Strength with uniform gaps to simulate various wireless signal strength (RSSI) values to change the quality of network links. Based on [6], we adjust the RSSI values for the average link success ratio (LSR) to be in the range $(0.71, 1.0)$.

A. Inverted Pendula Case Study

Our first use case is typical IIOT application, with three inverted pendula mounted to motorized carts [40], [44], controlled from remote sites through a wireless network. This highly nonlinear dynamics are extremely sensitive to delays and losses induced by the network. A single pendulum scheme is shown in Figure 4. The controller receives the inputs of both the angle θ and the displacement x, and applies a force F to the cart in order to keep the inverted pendulum balanced. The objective of the control system is to stabilize the pendulum in the vertical position ($\theta = 0$), and and to maintain the cart in (or move it to) a specific x position, while keeping the pendulum in the vertical position. With a poor controller, the pendulum will fall down. Real-world examples that relates directly to this inverted pendulum system is the attitude control of a booster rocket at takeoff, segways, etc. In particular, several

[2]http://wsn.cse.wustl.edu/index.php/Testbed

TABLE I: Parameters and values for the simulation of the IP and NPP use cases, with $l = 0, \ldots, 4$ and $j = 1, \ldots, 8$.

Parameters	IP	NPP
Sampling period T_s	0.01s	0.1s
Simulation time T_{sim}	100s	300s
RCA	$(6 + 4l)$m	$(2 + 2l)$MW
RCD	$5j$ sec	$15j$ sec
ST range	[0s, $T_{\text{sim}}-$RCD]	[0s, $T_{\text{sim}}-$RCD]

such systems can be deployed in the same environment, for example, segways that deliver packets in a factory floor.

In this use case, we consider a two-dimensional problem where the pendulum is constrained to move in the vertical plane, and the cart has only one degree of freedom to move (back and forth). In particular, we evaluate this use case over several different scenarios, where the reference signal requires the cart to move of a distance of RCA (while keeping the pendulum stable in the vertical position), within a time interval of RCD. The parameters used for the simulation are specified in Table I (note variables j and l in the caption).

B. Nuclear Power Plant (NPP) Case Study

A modern NPP design considers several SMRs [45], instead of a single large reactor, due to the flexibility and cost-benefit of starting and stopping SMRs. Typically there is one primary heat exchanger system (PHX) and two secondary heat exchangers (SHX) in each SMR. The PHX in the NPP has its main function the exchange of heat from inside of the reactor to the outside. The PHX is typically modeled as a nonlinear system. For each PHX, we focus on three measurements that are sent periodically to the controller, namely outlet hot leg temperature, inlet hot leg temperature, and mass flow rate.

We consider a case study of a NPP with three SMRs (three PHXs and six secondary heat exchangers[3]), each of which transmits measurement data via a shared wireless network (we focus on the 9 measurements sent periodically). Given that there are several SMRs in an NPP, the power output of each SMR may differ and the controller may decide to change the power output (reference function) of each SMR dynamically, based on energy requirements and balance the power required to achieve a certain level of power output. The PHXs are identical systems except for the reference functions. In reality, the reference function is set by the nuclear engineer/operator based on the NPP requirement. To be general in our case study, the RCA, RCD and ST values of each reference function are randomly chosen by uniform distribution from the range of values listed in Table I (note variables j and l in the caption). The parameters in *a set of reference functions* are 3 RCAs, 3 RCDs and 3 STs to set three reference functions. To be general and include all RCD values, we choose simulation time as 300s, taking into account the system settling time (even after the power change duration, the system still needs sometime to settle down to the setpoint). Each PHX will generate

[3]We only modeled one PHX, since the two secondary heat exchangers are backups for safety.

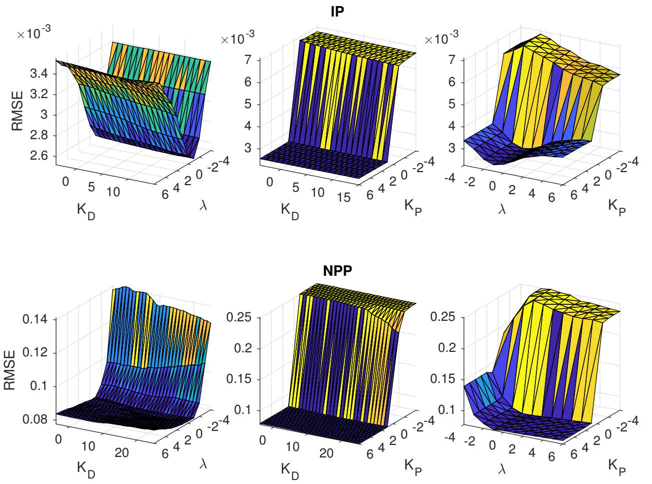
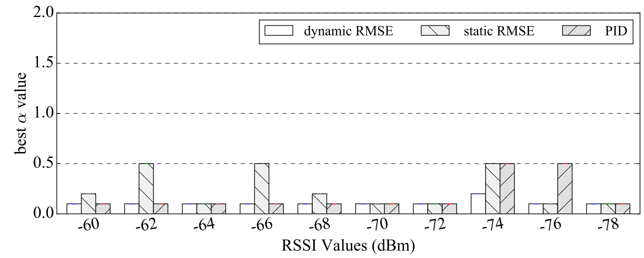
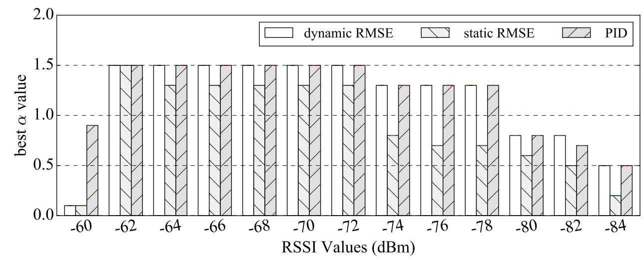


Fig. 5: RMSE computed for the tuning of the PID heuristic.



(a) IP



(b) NPP

Fig. 6: Best values of α as a function of RSSI.

one measurement packet (include three measurements: outlet temperature, inlet temperature and mass flow rate) and send out the packet through the wireless network periodically at the sampling period 0.1s. Recall that, if the measurement packet is lost during the wireless transmission, the system will use the latest received measurement value in the control algorithm.

VI. Quantitative Results from Case Study

In this section, we present how the different parameters of the presented approach are tuned, evaluate our network path quality model, and compare the three heuristic methods of measurement packets priority assignment with respect to the RMSE of the two use cases.

A. Control system results

1) Parameters tuning: A tuning phase is required only for the PID heuristic, and for deciding what is the best value of α of the PQModel in Equation (7).

a) Tuning the PID heuristic: To tune the three parameters of the PID heuristics, i.e., K_P, λ, and K_D of Equation (6), we consider the case where three physical systems are controlled over the network, and the RSSI is -60dBm, because it is the best network condition; the process is the same for different values of RSSI. We select the values of the parameters that minimize the $RMSE_{\text{tot}}$. In particular, we considered the parameters in the following ranges: $K_P \in [-4, 15]$, $\lambda \in [-4, 15]$, and $K_D \in [-4, 15]$. The numerical results of the tuning process for the two use cases is presented in Figure 5. Figure 5 provides a sensitivity analysis of the impact on the control system performance of different choices of control parameters, highlighting that the choice of K_D has little impact on the RMSE value, as well as choices of positive values of K_P. Such analysis suggests that the choice of λ is more critical, since the RMSE value is more sensitive to deviation from its best value. The selected values that minimize the $RMSE_{\text{tot}}$ are therefore $K_P = 1$, $\lambda = -1$, and $K_D = 9$ for the IP case, and $K_P = 1$, $\lambda = 2$, and $K_D = 0$ for the NPP case.

b) Tuning the value of α: To evaluate the network quality model proposed in Section IV-B, we run experiments with different α values from 0.0 to 2.0 for the three heuristic methods proposed in Section IV-A over different RSSI values on 20 sets of reference functions. Each experiment runs 20 times on the network paths given the RSSI value. Figure 6 shows the value of α that minimizes $RMSE_{\text{tot}}$ for different values of RSSI. It is interesting to notice that the values of α for the pendulum case are much lower than the ones for the NPP case. This highlights that the pendulum use case is much more sensitive to large delays. For the pendulum case, we can conclude that almost independently of the value of the RSSI, a low value of α must be selected.

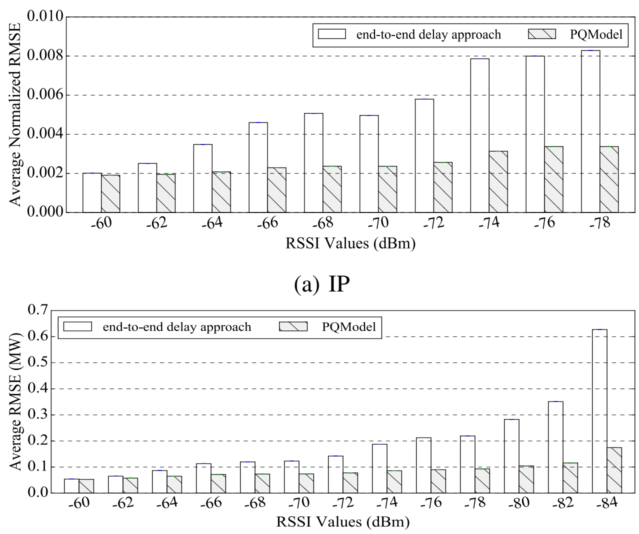
On the other hand, for the NPP case, in all three heuristic methods, the value of α decreases as the interference in the network increases. To figure out the reason, we counted the average number of time steps that a path quality order is selected for the cases of the best α values compared with when $\alpha = 1.0$ for each RSSI value and each heuristic method. Table II shows the impact of parameter α on the RMSE (in MW) for RSSI values of -64dBm and -84dBm, for heuristic method 1. Our method is as follows. The path order number column in the table shows the quality order of the three network paths. For example, 123 means that the highest priority packet will be sent via p_1 path and lowest priority packet will be sent via p_3 path. By comparing the average $RMSE_{\text{tot}}$, we can see that the system with $\alpha = 1.9$, RSSI $= -64$dBm and $\alpha = 0.2$, RSSI $= -84$dBm perform better than that with $\alpha = 1.0$ by 2.6% and 8%, respectively.

For the rest of the experiments, the best values of α are selected for a given RSSI, while the PID-heuristic parameters will be fixed.

	-64dBm		-84dBm	
path order number	$\alpha = 1.0$	$\alpha = 1.9$	$\alpha = 1.0$	$\alpha = 0.2$
123	1943	1244	125	649
132	123	123	108	48
213	2	700	24	257
231	876	877	1408	1937
312	24	2	104	7
321	32	54	1231	102
$RMSE_{\text{tot}}$(MW)	0.039	0.038	0.121	0.112

TABLE II: Comparison of the impact of α value on the control system performance for the NPP use case.

2) End-to-end delay and PQModel comparison: We evaluate the $RMSE_{\text{tot}}$ for end-to-end delay approach and PQModel approach over 100 different sets of the reference functions of three physical systems (both for the IP and for the NPP use cases). For each set of reference functions, we average the results of 20 runs on the three wireless network paths for each RSSI value. The average $RMSE_{\text{tot}}$ is shown in Figure 7. The



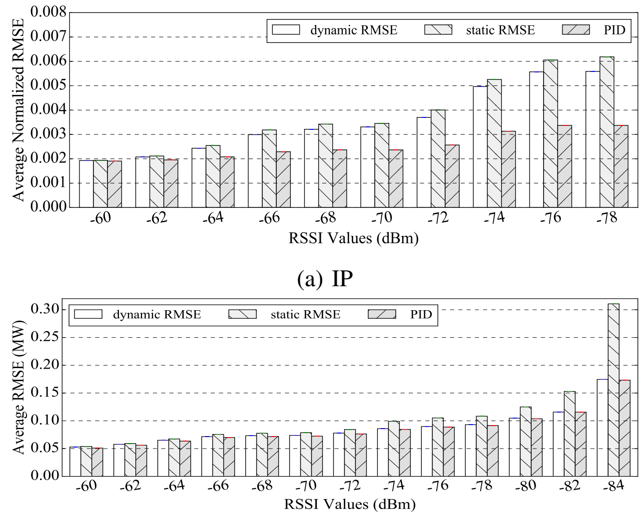
(a) IP

(b) NPP

Fig. 7: Comparison of the two network models as a function of the RSSI.

PQModel performs better (lower RMSE, that is, lower loss of power) than only considering end-to-end delay in all network conditions in both use cases, and provides a much more stable performance over different RSSI conditions. Such a result demonstrates how accounting for both delay and packet losses in the network model can significantly improve the quality of the obtained results, while providing more robust solutions against interference.

3) Comparison of packet priority assignment methods: We evaluated our approaches in the two use cases, for different values of RSSI. Figure 8 shows a comparison of the packet priority assignment methods while changing the RSSI to have different levels of network interference. The two dynamic packet priority assignment methods always perform better than



(a) IP

(b) NPP

Fig. 8: Comparison of the heuristic methods as a function of the RSSI, for the corresponding best values of α.

the static heuristic (center bar) from 6% to as much as 79%. This is because the packet priority of the static heuristic is determined ahead of time and fixed during the execution, not providing flexibility required when network conditions change and when the different PSs have different demands. Note also that dynamic packet priority are preferable when the network quality decreases.

Furthermore, Figure 8 also highlights that the PID heuristics provides a much more uniform performance with respect to an increasing network interference, performing always better than the dynamic RMSE heuristic. Such results suggest that the PID heuristic can provide a more robust priority assignment mechanism with respect to the RSSI than the other considered approaches. This is especially true for the IP use case, due to its higher sensitivity to the delays compared to the NPP case.

VII. Conclusion and Future Work

In this paper, we explored the interaction between dynamic packet scheduling and the control system performance in WCSs with one shared wireless network and multiple physical systems. Motivated by the observation that network delay and packet loss has different effects on control system performance depending on the system application demand, we proposed a dynamic priority assignment mechanism with the goal of minimizing the overall control system error caused by network imperfections, in presence of multiple control systems. Specifically, our solution has two steps: measurement packet priority assignment and network path quality determination taking into account both the network delay and the message loss. We evaluated our solution on two control systems that have multiple controlled systems with a single shared wireless network: (a) three inverted pendula and (b) three SMRs in nuclear power plant.

We came to a counter-intuitive conclusion that when the network has less interference, message loss is more important on quantifying the network quality; **but when the network has more interference, message loss is less important**, because the reliability of lower priority packets can be guaranteed anyway. In addition, our results also highlight the importance of exploring the relationship of network delay and message loss under different network conditions, which can help us reduce the control system performance degradation brought by the network imperfections.

This work also allows us to highlight an interesting control problem, that has not been widely addressed in the control literature, namely seeking for characterization and fundamental bounds of time-varying delays in networked control systems [46], [47], that is typically limited to linear systems. As future work, a theoretical analysis of how to identify such characterization and bounds is envisioned.

References

[1] J. Araújo, M. Mazo, A. Anta, P. Tabuada, and K. H. Johansson, "System architectures, protocols and algorithms for aperiodic wireless control systems," *TII*, vol. 10, no. 1, 2014.

[2] K. Gatsis, A. Ribeiro, and G. J. Pappas, "Optimal power management in wireless control systems," *TAC*, vol. 59, no. 6, 2014.

[3] T. Blevins, D. Chen, M. Nixon, and W. Wojsznis, *Wireless Control Foundation: Continuous and Discrete Control for the Process Industry*, 2015.

[4] A. Saifullah, C. Wu, P. B. Tiwari, Y. Xu, Y. Fu, C. Lu, and Y. Chen, "Near optimal rate selection for wireless control systems," *ACM Trans. Embed. Comput. Syst.*, vol. 13, no. 4s, 2014.

[5] S. Han, X. Zhu, A. K. Mok, D. Chen, and M. Nixon, "Reliable and real-time communication in industrial wireless mesh networks," in *RTAS*, 2011.

[6] B. Li, L. Nie, C. Wu, H. Gonzalez, and C. Lu, "Incorporating emergency alarms in reliable wireless process control," in *ICCPS*, 2015.

[7] M. Pajic, S. Sundaram, G. J. Pappas, and R. Mangharam, "Topological conditions for wireless control networks," in *CDC-ECC*, 2011.

[8] W. Wang, C. D'Angelo, D. Mosse, and D. Cole, "Integrating control and fault-tolerant wireless network design for small modular nuclear reactors," in *IRI*, 2016.

[9] B. Li, Y. Ma, T. Westenbroek, C. Wu, H. Gonzalez, and C. Lu, "Wireless routing and control: a cyber-physical case study," in *ICCPS*, 2016.

[10] C. Lu *et al.*, "Real-time wireless sensor-actuator networks for industrial cyber-physical systems," *Proc. of the IEEE*, vol. 104, no. 5, 2016.

[11] H. Ye, G. C. Walsh, and L. G. Bushnell, "Real-time mixed-traffic wireless networks," *TIE*, vol. 48, no. 5, 2001.

[12] Y. Liu, R. Candell, and N. Moayeri, "Effects of wireless packet loss in industrial process control systems," *ISA Trans.*, vol. 68, 2017.

[13] H. Pei Breivold and K. Sandström, "Internet of things for industrial automation – challenges and technical solutions," in *DSDIS*, 2015.

[14] J. Wurm, K. Hoang, O. Arias, A. Sadeghi, and Y. Jin, "Security analysis on consumer and industrial IoT devices," in *ASP-DAC*, 2016.

[15] S. Mubeen, S. A. Asadollah, A. V. Papadopoulos, M. Ashjaei, H. Pei-Breivold, and M. Behnam, "Management of service level agreements for cloud services in IoT: A systematic mapping study," *IEEE Access*, vol. 6, 2018.

[16] A. Grau, M. Indri, L. L. Bello, and T. Sauter, "Industrial robotics in factory automation: From the early stage to the internet of things," in *IECON*, 2017.

[17] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems – a survey," *IEEE TII*, vol. 9, no. 1, 2013.

[18] H. Li, M.-Y. Chow, and Z. Sun, "Optimal stabilizing gain selection for networked control systems with time delays and packet losses," *IEEE TCST*, vol. 17, no. 5, 2009.

[19] A. Onat, T. Naskali, E. Parlakay, and O. Mutluer, "Control over imperfect networks: Model-based predictive networked control systems," *TIE*, vol. 58, no. 3, 2011.

[20] G. Pin and T. Parisini, "Networked predictive control of uncertain constrained nonlinear systems: recursive feasibility and input-to-state stability analysis," *TAC*, vol. 56, no. 1, 2011.

[21] A. Ulusoy, O. Gurbuz, and A. Onat, "Wireless model-based predictive networked control system over cooperative wireless network," *IEEE TII*, vol. 7, no. 1, 2011.

[22] S. Hong, X. S. Hu, T. Gong, and S. Han, "On-line data link layer scheduling in wireless networked control systems," in *ECRTS*, 2015.

[23] T. Zhang, T. Gong, C. Gu, H. Ji, S. Han, Q. Deng, and X. S. Hu, "Distributed dynamic packet scheduling for handling disturbances in real-time wireless networks," in *RTAS*, 2017.

[24] X. Jin, F. Kong, L. Kong, W. Liu, and P. Zeng, "Reliability and temporality optimization for multiple coexisting wirelesshart networks in industrial environments," *TIE*, vol. 64, no. 8, 2017.

[25] J. Kim, K. Lakshmanan, and R. R. Rajkumar, "Rhythmic tasks: A new task model with continually varying periods for cyber-physical systems," in *ICCPS*, 2012.

[26] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "End-to-end delay analysis for fixed priority scheduling in wirelesshart networks," in *RTAS*, 2011.

[27] A. Saifullah, D. Gunatilaka, P. Tiwari, M. Sha, C. Lu, B. Li, C. Wu, and Y. Chen, "Schedulability analysis under graph routing in WirelessHART networks," in *RTSS*, 2015.

[28] G. Alderisi, S. Girs, L. L. Bello, E. Uhlemann, and M. Björkman, "Probabilistic scheduling and adaptive relaying for WirelessHART networks," in *ETFA*, 2015.

[29] K. Gatsis, A. Ribeiro, and G. J. Pappas, "Control-aware random access communication," in *ICCPS*, 2016.

[30] M. Pajic, S. Sundaram, G. J. Pappas, and R. Mangharam, "The wireless control network: A new approach for control over networks," *TAC*, vol. 56, no. 10, 2011.

[31] W. Wang, D. Mosse, J. G. Pickel, and D. Cole, "Work-in-progress: Cross-layer real-time scheduling for wireless control system," in *RTAS*, 2017.

[32] ——, "Work-in-progress: Wireless network reconfiguration for control systems," in *RTAS*, 2017.

[33] W. Wang, D. Mosse, D. Cole, and J. G. Pickel, "Dynamic wireless network reconfiguration for control system applied to a nuclear reactor case study," in *RTNS*, 2018.

[34] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, "Efficient network flooding and time synchronization with glossy," in *IPSN*, 2011.

[35] F. Terraneo, L. Rinaldi, M. Maggio, A. V. Papadopoulos, and A. Leva, "FLOPSYNC-2: efficient monotonic clock synchronisation," in *RTSS*, 2014.

[36] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Real-time scheduling for wirelesshart networks," in *RTSS*, 2010.

[37] P. Pazzaglia, L. Pannocchi, A. Biondi, and M. Di Natale, "Beyond the Weakly Hard Model: Measuring the Performance Cost of Deadline Misses," in *ECRTS 2018*, vol. 106, 2018.

[38] V. Dolk and M. Heemels, "Event-triggered control systems under packet losses," *Automatica*, vol. 80, 2017.

[39] F.-L. Qu, Z.-H. Guan, T. Li, and F.-S. Yuan, "Stabilisation of wireless networked control systems with packet loss," *IET CTA*, vol. 6, no. 15, 2012.

[40] K. Åström and R. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 2018.

[41] Y. Wang, M. C. Vuran, and S. Goddard, "Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks," in *RTSS*, 2009.

[42] F. Terraneo, A. Leva, S. Seva, M. Maggio, and A. V. Papadopoulos, "Reverse flooding: exploiting radio interference for efficient propagation delay compensation in WSN clock synchronization," in *RTSS*, 2015.

[43] W. Wang, D. Mosse, and D. G. Cole, "Bitvector: Fault tolerant aggregation scheme for monitoring in nuclear power plants," in *ICESS 2015*.

[44] J. Eker and A. Cervin, "A matlab toolbox for real-time and control systems co-design," in *RTCSA*, 1999.

[45] S. R. Greene *et al.*, "Pre-conceptual design of a fluoride-salt-cooled small modular advanced high-temperature reactor (SmAHTR)," Oak Ridge National Laboratory, Oak Ridge, TN, USA, Tech. Rep. ORNL/TM-2010/199, 2010.

[46] K. Gu, J. Chen, and V. L. Kharitonov, *Stability of time-delay systems*. Springer Science & Business Media, 2003.

[47] J.-H. Kim, "Note on stability of linear systems with time-varying delay," *Automatica*, vol. 47, no. 9, 2011.