Wireless Routing and Control: a Cyber-Physical Case Study

Bo Li[1]*, Yehan Ma[1]*, Tyler Westenbroek[2], Chengjie Wu[1], Humberto Gonzalez[2], Chenyang Lu[1]
[1]Department of Computer Science & Engineering, Washington University in St. Louis
[2]Department of Electrical & Systems Engineering, Washington University in St. Louis

ABSTRACT

Wireless sensor-actuator networks (WSANs) are being adopted in process industries because of their advantages in lowering deployment and maintenance costs. While there has been significant theoretical advancement in networked control design, only limited empirical results that combine control design with realistic WSAN standards exist. This paper presents a cyber-physical case study on a wireless process control system that integrates state-of-the-art network control design and a WSAN based on the WirelessHART standard. The case study systematically explores the interactions between wireless routing and control design in the process control plant. The network supports alternative routing strategies, including single-path source routing and multi-path graph routing. To mitigate the effect of data loss in the WSAN, the control design integrates an observer based on an Extended Kalman Filter with a model predictive controller and an actuator buffer of recent control inputs. We observe that sensing and actuation can have different levels of resilience to packet loss under this network control design. We then propose a flexible routing approach where the routing strategy for sensing and actuation can be configured separately. Finally, we show that an asymmetric routing configuration with different routing strategies for sensing and actuation can effectively improve control performance under significant packet loss. Our results highlight the importance of co-joining the design of wireless network protocols and control in wireless control systems.

*The first two authors contributed equally to this work.

ICCPS'16 April 11–14, 2016, Vienna, Austria.

Categories and Subject Descriptors

C.2.4 [**Distributed Systems**]: Distributed applications; C.2.1 [**Network Architecture and Design**]: Wireless communication

Keywords

Cyber-Physical System, Wireless Sensor-Actuator Network, Routing, Control, Process Control

1. INTRODUCTION

Wireless sensor-actuator network (WSAN) technology is gaining rapid adoption in process industries as a means to lower deployment and maintenance costs in challenging industrial environments. Industrial standard organizations such as ISA [3], HART [6], WINA [4] and ZigBee [5] have been actively pushing WSAN for industrial automation. While early success of industrial WSANs primarily focused on monitoring applications, it remains challenging to support actuation applications over WSANs due to their vulnerability to packet loss. To realize the full potential of WSANs for both sensing and actuation, wireless control has received considerable attention in recent research on control theory. For example, state observers have been employed to compensate for packet loss from sensors [30], while previous control inputs can be buffered at actuators for use in case of packet loss from the controller [17]. Despite considerable success in theoretical advancements in wireless control design, however, there are few empirical studies on wireless control systems that combine state-of-the-art control design and standard-based industrial WSANs under realistic wireless conditions.

This paper presents a cyber-physical study on a wireless process control system to systematically explore the interactions between wireless routing and control design, a problem that has received little attention in the literature. On the network side, the WirelessHART standard supports two alternative routing strategies, including single-path source routing and multi-path graph routing, where graph routing reduces packet loss through

path diversity at the cost of additional overhead and energy consumption. On the control side the system integrates an observer based on an Extended Kalman Filter with a model predictive controller and an actuator with a buffer for recent control inputs. The case study is implemented in the Wireless Cyber-Physical Simulator (WCPS) [19, 20] that integrates Simulink with a WirelessHART protocol stack based on a realistic wireless model and on traces collected from a real-world wireless testbed. Our experiments demonstrate that a wireless control system can have different levels of resilience to packet loss for sensing and actuation. Specifically, in our case study, while the state observer is highly effective in mitigating the effects of packet loss from the sensors to the controller, the control performance is more sensitive to packet loss from the controller to the actuators despite the buffered control inputs.

Motivated by this observation, we propose an asymmetric routing approach for WSANs. In contrast to traditional WSANs that employ a uniform routing strategy in the entire network, asymmetric routing can employ different routing strategies for sensing and actuation. This flexible routing approach enables a cyber-physical co-design approach to wireless control design in which routing strategies can be tailored for the characteristics of control design. For example, in our case study, an asymmetric routing configuration (source routing for sensing and graph routing for actuation) effectively improves performance under significant packet loss. Our results highlight the importance of co-joining the design of wireless network protocols and control in wireless control systems.

The contributions of this work are three-fold.

1. A cyber-physical framework that integrates state-of-the-art control techniques such as observers, MPC and buffered actuation, all connected through a standard-based WirelessHART network;
2. An asymmetric WSAN routing approach that enables differentiated redundancies of sensing and actuation under the proposed control framework;
3. A systematic case study that presents in-depth interaction of wireless routing and control in a holistic fashion.

The rest of the paper is organized as follows. Section 2 discusses related works. Section 3 introduces the wireless control design. Sections 4 and 5 present the design and results of the case study. Section 6 concludes the paper.

2. RELATED WORK

WSAN has received significant attentions as a new communication paradigm for industrial cyber-physical systems [23]. Routing is an essential element of industrial WSANs because of its impacts on the reliability, latency and schedulability of wireless communication over multi-hop mesh topologies. Detailed requirements in low-power routing and lossy networks were introduced in [14, 21, 25]. A graph routing algorithm for WirelessHART was presented in [15]. Schedulability analysis under graph routing in WirelessHART networks was introduced in [26]. A conflict-aware real-time routing algorithm for industrial WSAN was presented in [35]. An energy-efficient routing approach was introduced in [34]. Energy-aware routing for real-time and reliable wireless industrial sensor networks was introduced in [16]. These works focus solely on the network without considering the control aspect of a wireless control system.

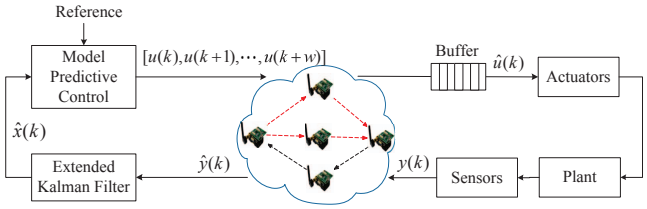


Figure 1: System Architecture: Sensor measurements from sensors are transmitted wirelessly to the observer, which generates estimated system states. The controller takes estimated system states and computes actuation commands, which again are transmitted to actuators wirelessly. We introduce a buffer for actuators such that we can reuse previously buffered actuation inputs when packet drops happen.

Several promising control approaches have been proposed to deal with uncertainties in wireless control systems. Examples include passivity-based control [18], event-triggered [33] and self-triggered control [7], and distributed control [24]. State observers such as Kalman filters have been proposed to enhance a system's resiliency against uncertainties [22, 28] and intermittent observations [29]. While the results are encouraging, these works do not consider routing in a wireless mesh network.

Cyber-physical co-design has emerged as an effective approach for wireless control system design. A co-design of transmission scheduling and control was explored in [12]. Sampling rate selection for wireless control systems was studied in [27]. Etherware in [9, 17] introduced a middleware architecture for wireless control comprised of state observers, MPC and actuation buffers. Etherware was designed based on a WiFi network and did not investigate the issue of routing in industrial wireless mesh networks. To our best knowledge, none of aforementioned works addressed the interaction between routing

and control, which is the focus of our work.

3. WIRELESS CONTROL DESIGN

In this section, we first give an overview of our system architecture. We then introduce the control design. Finally, the wireless network design is introduced.

3.1 System Overview

We consider a wireless control system consisting of a physical plant, a centralized controller, and a shared WSAN. Sensors and actuators communicate through a multi-hop wireless mesh network. In *sensing*, sensors send their measurements to the controller. Actuation commands computed by the controller are sent to actuators during *actuation* through the same WSAN. The system architecture is shown in Fig. 1.

On the control side, we use state-of-art control schemes. We use a state observer in *sensing* to mitigate nuisances caused by the wireless network, and we control the system using a Model Predictive Control (MPC) scheme. Then, in *actuation*, we introduce a buffer, which stores a sequence of control commands computed by the controller.

3.2 Control Design

3.2.1 Sensing with Extended Kalman Filter

State observers such as Kalman filters have been shown to be resilient against uncertainties [22,28,29]. We have implemented an Extended Kalman Filter (EKF) as part of our cyber physical case study to robustly estimate the state of the physical plant under packet loss in the wireless network.

An EKF is in practice a recursive algorithm with two main steps: *prediction* and *update*. In simple terms, the *prediction* step estimates the states of the system associated with the previous step. Then, the *update* step compares the current outputs estimation with newly arrived sensing data, and improves the estimation of the current state variables. Yet, when packets are dropped by the wireless network, the *update* step needs to be modified, as described by Sinopoli et al. in [29]. That is, when the wireless network drops a sensing packet the *update* step is disregarded, therefore returning the same value as the *prediction* step. In our WSAN design, we use a modified EKF with a similar idea.

3.2.2 Model Predictive Control with Buffers

As a controller for our closed-loop WSAN system, we have adopted a Model Predictive Control scheme (MPC, also referred to as Receding Horizon Control) [13]. MPC is implemented by solving a finite-horizon optimal control problem, with horizon $T > 0$, every Δ seconds, where usually Δ is the sampling rate of a discrete-time and $\Delta \ll T$. An ideal MPC has a horizon T as large as possible, to approximate infinite-horizon optimal control, and Δ as small as possible, to observe the state often and therefore react to changes fast. In real-world applications, however, as T increases, so does the computation cost, which in turn forces us to choose a smaller T or a larger Δ.

Although a sequence of control commands, corresponding to the whole time horizon of length T, are computed by the MPC, only the first sample is commonly used for actuation while the rests are discarded. To improve the resilience, instead of transmitting the first sample, we transmit a certain sequence of values in the prediction horizon. This sequence is received by a buffer next to the actuator, and feeds the actuator with one control input per time sample. If the wireless network does not drop packets, then the buffer is completely replaced with a new sequence of control inputs every time sample. But if the wireless network drops a packet, then the buffer simply applies the next available control input, which has already been received in the last packet that arrived successfully. We refer to this approach as a *buffered actuation*. Even though *buffered actuation is* not a new term [10, 11], to our knowledge, this is the first paper studying it in conjunction with a wireless mesh network based on the WirelessHART standard. In particular, we present what we believe to be the first systematic study of the interaction between wireless routing protocols and control.

As in Fig. 1, the state observer takes plant measurements $\hat{y}(k)$to produce estimated system states $\hat{x}(k)$; an MPC takes $\hat{x}(k)$, compares them with the reference signal and generates a sequence of predicted control commands $\big[u(k), u(k+1), \ldots, u(k+w-1)\big]$ with a length of w, which we explain later in more detail. The sequence is later stored in a buffer on the actuator and $\hat{u}(k)$, the control command corresponding to the current time, is applied to the actuator.

3.3 Network Design

3.3.1 WirelessHART Architecture

We adopt a WirelessHART [6] architecture for our WSAN design. WirelessHART uses multiple channels defined in IEEE 802.15.4 physical layer specification, and adopts *channel hopping* for the sake of channel diversity. Any excessively noisy channel will be *blacklisted* by the centralized network manager.

A WirelessHART network is a multi-hop mesh network consisting of a number of field devices connected to a gateway through access points. The network is managed by a centralized network manager. The network manager collects topology information from the field devices, computes routes and transmission schedules,

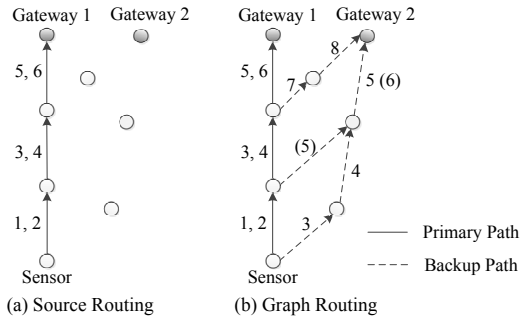


Figure 2: Source and Graph Routing

and disseminates the routing information and schedules among field devices. Transmissions are scheduled based on Time Division Multiple Access (TDMA) comprised of *10ms* time slots. For transmissions between sender/receiver pairs, a time slot can either be *dedicated* or *shared*. In a *dedicated* slot, only one sender is allowed to transmit. In a *shared* slot, more than one sender competes for one transmission opportunity.

3.3.2 Routing Strategies

WirelessHART supports two alternative routing strategies: *graph* routing and *source* routing. *Source* routing provides a single route for each data flow, whereas *graph* routing firstly allocates a primary path from the source to the destination and further adds a backup path from each intermediate node to the destination. As shown in Fig. 2, in *source* routing, only one sender/receiver pair will be considered for the first transmission and the retransmission. In *graph* routing, the network manager allocates a *dedicated* slot for the first transmission, followed by a retransmission slot between the same sender/receiver pair; finally, the network manager allocates a second retransmission in a *shared* slot for the same sender but a different receiver on the routing graph. In other words, *graph* routing supports up to three transmission attempts for a single packet, which as we show later, significantly improves end-to-end delivery ratios.

Despite differentiated redundancies offered by *graph* and *source* routing, there are more tradeoffs between the two in terms of cost. For example, more retransmissions in *graph* routing potentially will cause higher energy costs. Further, in a control system driven by a TDMA network, a design employing *graph* routing will need more time slots, and thus a longer network period, which in turn results in lower control frequency. *Source* routing, on the other hand, features less energy consumption, faster network periods, but worse network reliability. We show results with more details in Section 5.

3.3.3 Asymmetric Routing

Traditional WSAN networks such as WirelessHART employs a uniform routing strategy across an entire network. That is, if the operator chooses graph routing, a graph routing protocol will be used for all flows in the network. However, we observe that the control system has different levels of resiliency to packet loss for sensing and actuation. Indeed, past results in state observers for intermittent measurements [29] prove their effectiveness in mitigating the effects of packet loss from sensors to controller. But there are no similar theoretical results to mitigate losses from controller to actuators; thus we must rely on heuristics such as the use of actuation buffers.

Motivated by this key observation, we propose a novel asymmetric routing approach tailored for wireless control. Under asymmetric routing, the sensing and actuation routes can be configured independently from each other, so that different routing strategies may be used for sensing and actuation. For example, as the control system is less vulnerable to packet loss for sensing than that for actuation, we choose to employ source routing for sensing and graph routing for actuation. Note that the flexibility of asymmetric routing enables a cyber-physical co-design approach by tailoring the routing strategies based on the characteristics of the control systems. As a result, the network can use graph routing to provide reliable communication to the part of the control system that is more vulnerable to packet loss, while employing source routing to the more resilient part of the control system to conserve network resources and energy, thereby combining the benefits of graph routing and source routing. As demonstrated in our case study (see Section 5), asymmetric routing clearly outperforms traditional designs with uniform routing.

While the asymmetric routing approach can be based on any source and graph routing algorithms, our current implementation extends the energy-efficient source routing and graph routing algorithms presented in [34]. Those original algorithms were designed to improve the network lifetime of a WirelessHART network, e.g., the operation time till the first field device's battery runs out. As many WSANs operate on batteries in industrial environments, network lifetime is an important concern for industrial WSANs. Due to the extra paths and transmissions involved in graph routing, the graph routing algorithm results in shorter network lifetime than the source routing algorithm. In [34] the source routing and graph routing algorithms were designed as separate algorithms, each of which was applied to the entire network when used. We extend and combine those algorithms in an asymmetric routing framework, where different routing strategies can be applied to sensing and actuation.

4. CASE STUDY DESIGN

In this section, we first introduce a nonlinear exothermic chemical reaction system as our control plant. We then introduce a detailed design of the Extended Kalman Filter and buffered actuation; the WSAN implementation will be introduced in the end.

All the following design and case studies are performed in WCPS [20], an open-source simulator for holistic simulations of wireless control systems. WCPS supports co-simulation of control systems implemented in Simulink and WSAN implemented in TOSSIM. WCPS has previously been used for realistic case studies of wireless structural control systems [20, 31], of wireless process control systems [19], and it was recently used to build a benchmark problem for wireless structural control released to the civil engineering community [32].

4.1 Physical Plant

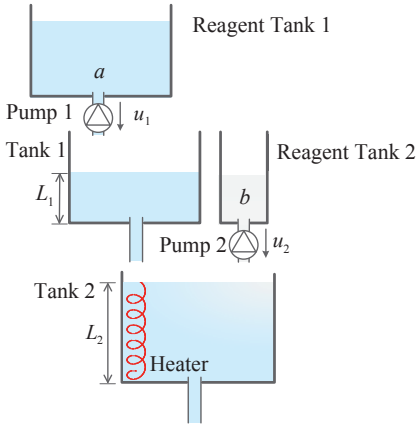


Figure 3: Exothermic Chemical Reaction Plant

A diagram of our dynamical system, consisting of the exothermic chemical reaction of two fluids flowing between a collections of tanks, is shown in Figure 3. Although our dynamical system only considers a few state variables, its dynamic overall behavior is similar to that of many industrial process control systems, such as irrigation networks [7] or oil refineries [36]. Our choice of this plant is also motivated by its rich nature, since the system's state considers a mix of chemical, temperature, and fluid level variables, involving both fast and slow dynamics.

The system is comprised of four tanks: Tank 1 is fed with chemical reagent a via a Reagent Tank connected with a pump, and Tank 2 is fed with chemical reagent b in a similar fashion. Tank 1 is placed higher than Tank 2; then the fluid in Tank 1 flows into Tank 2 due to the gravity. The liquid levels of Tank 1 and Tank 2 are denoted $L_1(t)$ and $L_2(t)$, respectively. The temperature of the solution in Tank 2 is denoted

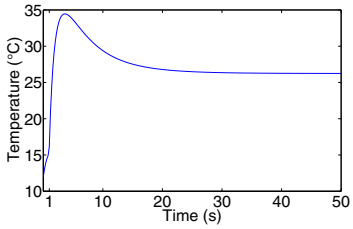


Figure 4: Open-loop Step Response

$T_2(t)$. The concentrations of chemical reagents a and b in Tank 2 are denoted by $A(t)$ and $B(t)$, respectively. There are two actuator inputs in this system consisting of the pumps feeding Tanks 1 and 2, denoted $u_1(t)$ and $u_2(t)$, respectively. We use $L_1(t)$, $L_2(t)$, $T_2(t)$ and outgoing flow speeds of Tank 1, Tank 2, and Reagent Tank 1 as sensing (monitoring) variables, and $u_1(t)$ and $u_2(t)$ as two actuation variables.

Table 1 shows the list of parameters modeling the dynamic behavior of our plant. To simplify our notation, we define:

$$g(L) = \alpha \arctan(\beta L), \tag{1}$$

which we use to approximate the pressure of a fluid in a tank's output pipe with fluid level L.

The level of fluid in Tank 1 is modeled using the following differential equation:

$$A_1 \dot{L_1} = -k_1 g(L_1) + u_1. \tag{2}$$

Similarly, the level of fluid in in Tank 2 is modeled by:

$$A_2 \dot{L_2} = k_1 g(L_1) - k_2 g(L_2) + u_2. \tag{3}$$

The dynamical model for the temperature in Tank 2 is:

$$A_2 L_2 \dot{T_2} = H_c \left(k_1 t_1 g(L_1) - A_2 T_2 g(L_2) + t_2 u_2\right) + h_2 L_2 + H_r r_c k_r A B \exp\left(-\frac{E_a}{R T_2}\right), \tag{4}$$

where an exothermic reaction that transforms reagents a and b into solution c, produces heat following an exponential model. The rate at which both reagents change follows this model:

$$\begin{bmatrix}\dot{A}\\ \dot{B}\end{bmatrix} = \begin{bmatrix}a_0 k_1 g(L_1) - k_2 A g(L_2)\\ -k_2 B g(L_2) + b_0 u_2\end{bmatrix} - \begin{bmatrix}r_a\\ r_b\end{bmatrix} k_r A B \exp\left(-\frac{E_a}{RT_2}\right). \tag{5}$$

Fig. 4 shows the response of the plant when a step is applied in each input, u_1 and u_2, at $t = 1$. We observe that the step response of the plant becomes stable at about $t = 30$. Therefore, the time constant of the plant's response to a step is roughly 6 seconds [8].

Name	Physical Meaning	Value
L_1, L_2	fluid levels of Tanks 1 and 2	—, — m
T_2	temperature in Tank 2	— $°C$
A, B	concentrations of reagents a and b in Tank 2	—, — mol/L
u_1, u_2	input flow rate of Pump 1 Pump 2	—, — L/s
H_r	reaction energy generation rate	— J/s
A_1, A_2	cross sectional areas of Tank 1 and Tank 2	2, 2 m^2
k_1, k_2	flow rates of the pipes in Tank 1 and Tank 2	1, 1 $L/s\sqrt{m}$
t_1, t_2	temperatures of reagents a and b	20, 20 $°C$
H_c	heat generated per mol of reaction	0.5 $J\,°C$/mol
h_2	power of heater	40 J/s
a_0, b_0	concentrations of a, b in each Reagent Tank	1, 1 mol/L
r_a, r_b, r_c	reaction rates of reagents a, b, and c	1, 1, 1
E_a	activation energy of chemical reaction	1 $J\,°C$/mol
R	molar gas constant	8.134 J/Kmol
α, β	pipe flow model parameters	2, 0.5
k_r	reaction rate constant	0.2 mol/s

Table 1: Plant Parameters

4.2 Observer and Controller Implementation

As explained above, we use an Extended Kalman Filter for intermittent observations, described in [29], as our observer, and a Model Predictive Control scheme with signal buffers as our controller. Using the same notation as in Figure 1, the output of the system is $y(k) = \big[L_2(k),\, T_2(k)\big]$, which is sent to the EKF via the wireless network. It is worth noting that the EKF also receives the input $u(k)$ computed by the MPC scheme, as is common with dynamical state observers, which require the inputs and outputs of the plant to estimate its states. Yet, the input actually applied to the plant is $\hat{u}(k)$, which is equal to $u(k)$ whenever the wireless network successfully delivers a packet, but it differs from $u(k)$ whenever the network fails to deliver the latest input update (in that case $\hat{u}(k)$ equal the next available packet in the buffer, as explained in Section 4.3). We added this simplification in our closed-loop implementation because the buffer is located remotely from the EKF. Hence our state observer has no way of knowing (without delay) whether $\hat{u}(k)$ equals $u(k)$ or not. Assuming that all actuation packets are successfully delivered is a reasonable compromise in this situation.

Our MPC scheme solves a discrete-time finite-horizon constrained Linear-Quadratic Regulator (LQR) optimal control problem at each time step. Given $u(k-1)$, we linearize the system around this control signal and use those matrices in our controller, satisfying safety box constraints for states and inputs. An integrator was added to the controller model to eliminate the steady-state bias in our regulation objective; hence the controller considers a model with 7 states: the original 5

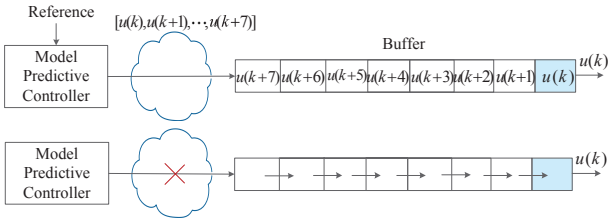


Figure 5: Buffered Actuation Mechanism

states of the plant plus 2 integrators (one for each input). The prediction horizon was chosen as $N = 21$; yet due to constraints in the network packet size, we only transmit to the buffer 8 of the 21 input samples in each iteration. The resulting quadratic programming optimization problem was solved in Matlab/Simulink using Gurobi optimizer [2].

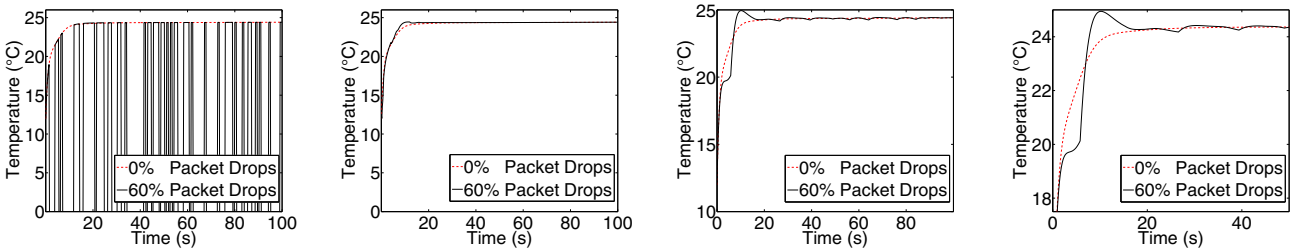
4.3 Buffered Actuation

As mentioned above, we place a buffer on each actuator. The size of the buffer is primarily decided by the capacity of an IEEE 802.15.4 packet, as we assume that actuation commands are carried by one packet per control period. The packet size defined by IEEE 802.15.4 is 128 bytes, comprised of an 11 byte header, a 7 byte metadata, and a 110 byte payload. We allocate 60% of the payload area (64 bytes) for actuation values while saving the rest for other uses, e.g., periodic actuator compensation or calibration. In our study, we use *float* data types for actuation values, which take 4 bytes per value. The 64 bytes buffer-size also defines a bound for the allowable size of transmitted actuation values through wireless, i.e., 8 pairs of u_1 and u_2.

Fig. 5 shows how the buffered actuation works in our design. With a buffer size of 8 elements, in time step 1, if there is no packet drop, the first value $u(k)$ on the rightmost will be used for actuation, replacing all the information in the buffer with the newly received data. In step 2, if the actuation packet is lost, the remaining values in the buffer, headed by $u(k+1)$, will be shifted right, and $u(k+1)$ will be used for actuation. This shifting and reuse mechanism goes on whenever packets are dropped until all values are used. In the worst case, after 8 consecutive packets are lost, the actuator will retain the value $u(k+7)$ until a new actuation packet arrives.

4.4 Implementation of WSAN

Our WSAN uses the WirelessHART network protocol stack in WCPS 2.0. The WirelessHART stack includes a multichannel TDMA MAC layer, a real-time transmission scheduler, and a routing layer supporting source routing and graph routing [19]. We have implemented the asymmetric routing framework that allows any com-



(a) Before EKF. (b) After EKF. (c) Impact of Actuation Loss. (d) Zoom in of (c).

Figure 6: Results of EKF and Buffered Actuation: (a) Temperature measurement before EKF and with 60% sensing packet drops; (b) Temperature after EKF and with 60% sensing packet drops; (c) Temperature measurement with 60% actuation packet drops; (d) Zoom in view of (c).

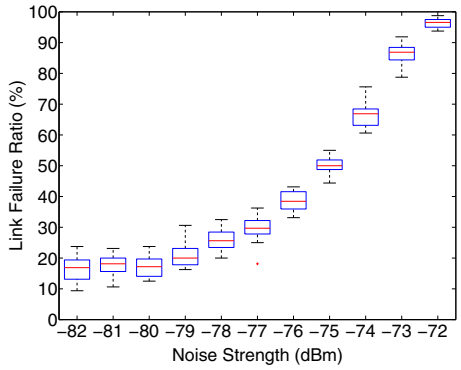


Figure 7: Link Failure Ratio.

binations of source and graph routing strategies to be used for sensing and actuation.

The simulated WSAN employs the topology of 16 nodes from the WSAN testbed at Washington University [1]. The WSAN includes 16 nodes and has an average sensor-controller-actuator distance of 4 hops. We collect the traces of received signal strengths and noise from the testbed as inputs to the wireless simulations in TOSSIM, the standard simulator of TinyOS with a realistic link model. To evaluate the wireless control systems under a wide range of wireless conditions, we introduce offsets to the noise traces to simulate varying levels of noise. By varying the offset in different experiments, we can simulate network conditions ranging from normal conditions to stress tests with excessive packet drops. Fig. 7 shows link failure ratios under different average noise strengths. The statistical distributions are based on all bidirectional links in the topology with 500 transmissions per link. The lowest noise level (-82 dBm) results in a 15% link failure ratio, while the highest noise level (-72 dBm) causes around 98% link failures, which represents extreme conditions such as adversarial jamming attacks or extreme weather conditions.

5. CASE STUDY RESULTS

In the following case study, we use the exothermic chemical reaction system introduced earlier. The control goal is to reach a target temperature in Tank 2 (see Fig. 3). Pumps on Reagent Tank 1 and Reagent Tank 2 are used as two actuators.

We will first test and compare the resilience of the state observer and actuation buffer to packet drops. We will then explore the trade-off between source and graph routing in terms of network performance and cost. Finally we will evaluate the performance of the integrated wireless control system through holistic simulations.

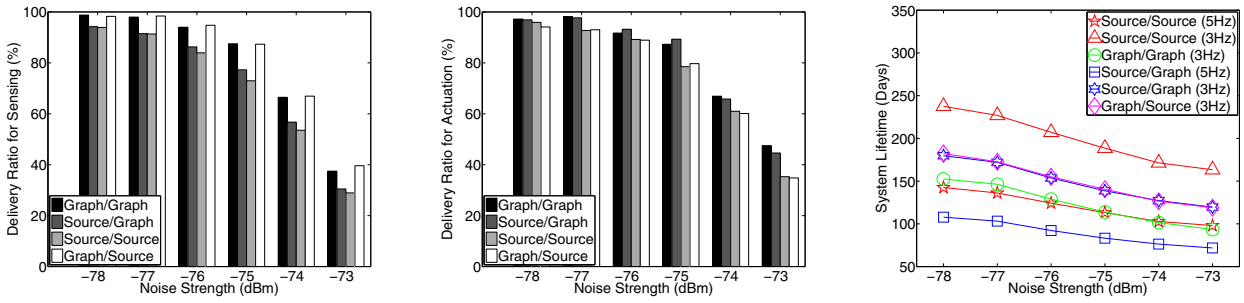
5.1 EKF and Buffered Actuation Results

We tested our network under high stress conditions for the resilience and robustness of our closed-loop implementation. Figs. 6(a) and 6(b) show the performance of the EKF when 60% of the sensing packets are dropped. We observe that the EKF does a very good job at filtering out jitters caused by sensing packet drops. Fig. 6(c) shows closed-loop temperature control results when 60% of the actuation packets are dropped, hence relying on the buffers to cover for the lost data. In this case, we see an obvious overshoot caused by actuation packet drops. Zooming in Fig. 6(d) reveals an overshoot of over 2 °C. These results support our conjecture that actuation is more vulnerable to packet drops than sensing.

5.2 Network Results

To simplify our exposition, and follow the implementation described in Section 3.3.3, we will denote abbreviate source routing as S and graph routing as G. Moreover, we will consistently denote the sensing routing approach first and the actuation routing second. For example, S/G corresponds to source routing for sensing and to graph routing for actuation.

Fig. 8(a) shows sensing delivery ratio as a function of the noise strength. We observe that delivery ratios of all routing approaches degrade as the strength of noise



(a) Sensing Delivery. (b) Actuation Delivery. (c) System Lifetime.

Figure 8: Wireless Network Results.

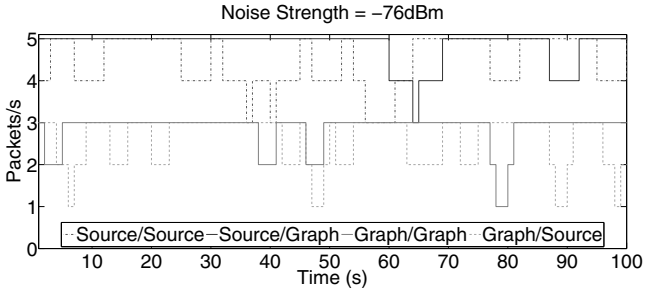


Figure 9: Data Rates for the Actuator

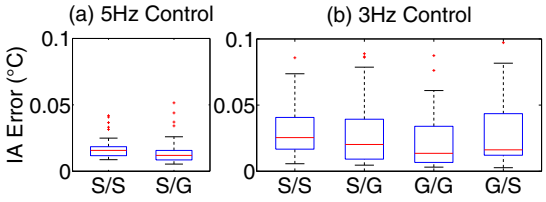


Figure 10: Integral Absolute Error for Temperature Control with Noise Strength -74 dBm.

increases. G/G and G/S consistently outperform other approaches because of the redundancy offered by graph routing. Note that -74 dBm noise has over 60% link failures in Fig. 7, yet we still see that G/G reaches nearly 60% multi-hop delivery ratios, which shows the strength of graph routing in improving network reliabilities.

Fig. 8(b) shows actuation delivery ratios. We observe similar trend as above, where S/G and G/G achieve better delivery ratios than the rest.

Recall that graph routing needs more time slots because it allocates 3 slots for each packet, whereas source routing only allocates 2 slots. As such, we get an asymmetry in the maximum frequency supported by each routing approach, where S/S and S/G support up to 5 Hz flows, while G/S and G/G support up to 3 Hz.

Fig. 8(c) shows the system lifetime performance of the network. In this simulation, we include two simulations for S/S and S/G, the first with flows at 5 Hz and the second with flows at 3 Hz, so we can compare the results across all routing approaches. We assume a general battery capacity of 8640 J, which is the typical capacity of two AA batteries. We define the system lifetime as the time to deplete the first node in the network. We observe that the system lifetime of all routing approaches degrades as the noise level increases, due to increasing retransmissions. S/S (3 Hz) has best system lifetime because it has the fewest transmissions. We observe that the S/G (5 Hz), however, has the worst system lifetime, because we use graph routing for actuation and also because it runs at a higher frequency. On the other hand, S/G (3 Hz) is at the second best, which confirms the impact of control frequency on system lifetime.

Fig. 9 shows actuation data rates at the noise strength of -76 dBm, where nearly 40% of the packets are dropped, as shown in Fig. 7. S/S and S/G, which support up to 5 Hz flows, clearly show higher data rates. Moreover, we observe S/G and G/G have less degradation due to the better reliability of graph routing in actuation.

5.3 Wireless Control Results

To evaluate our control results, we adopt two metrics: Integral Absolute Error (IAE) and Maximum Absolute Error (MAE). IAE is the normalized time-average of the absolute error between the closed-loop responses using wired control (i.e., no packet drops) and wireless control. MAE is the maximum value of the same absolute error.

Fig. 10 shows the IAE for the control, i.e., regulating T_2, with -74 dBm noise in the network. Here -74 dBm noise features around 60% end-to-end delivery ratio (i.e., 40% loss) for both sensing and actuation, which also implies that the control system operates normally under relatively large wireless loss ratios. The left plot shows control results running at 5 Hz using S/S and S/G, where S/G has smaller error than S/S, as expected. The right plot shows 3 Hz control with all four routing approaches, where G/G outperforms the

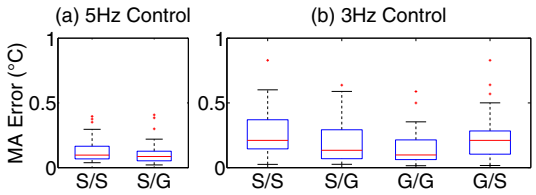


Figure 11: Maximum Absolute Error for Temperature Control with Noise Strength = -74 dBm.

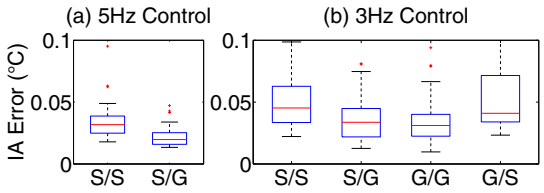


Figure 12: Integral Absolute Error for Temperature Control with Noise Strength = -73 dBm.

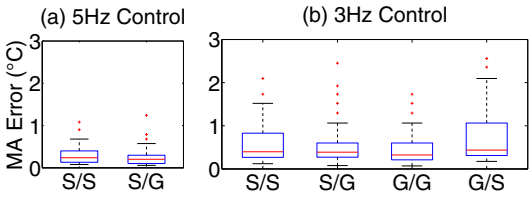


Figure 13: Maximum Absolute Error for Temperature Control with Noise Strength = -73 dBm.

other approaches, also as expected. A cross comparison between both plots reveals that S/G achieves the smallest IA error, since S/G runs at a higher frequency than G/G and G/S, while it has better actuation reliability than S/S. This is interesting as it clearly shows tradeoffs between WSAN reliability and control frequency.

Fig. 11 shows the MAE for the control goal. As in Fig. 10, we have 5 Hz results on the left and 3 Hz on the right. We observe a similar trend as before, with S/G outperforming the other approaches. This again shows the importance of taking advantage of appropriate tradeoffs between wireless routing and control frequency.

With increasing attention on cyber physical attacks, it would be interesting to see how our design reacts in extremely challenging conditions. We next introduce results under harsher noise conditions.

5.4 Results under Harsh Wireless Conditions

In the following study, we further level up the noise in our simulation to -73 dBm. We note -73 dBm features 85% median link failures, as shown in Fig. 7. We observe around 35% end-to-end delivery ratio for both sensing (Fig. 8(a)) and actuation (Fig. 8(b)).

Fig. 12 shows the IAE of both 5Hz and 3Hz control. We note G/G and G/S can only support up to 3Hz control, and Fig. 12 represents their best achievable performance.

In Fig. 12, we see that S/G (5Hz) behaves the best among both the 5Hz and the 3Hz methods. We note the IAE under -73dBm noise is 1.5-2 times larger than that in Fig. 10, which is because the former has 35% delivery ratio while the latter has 60%. Among 3Hz control in Fig. 12(b), we see G/G performs the best because it has better reliability for both sensing and actuation. G/S in Fig. 12(b) has a larger IAE than S/G, which again proves that actuation data is more vulnerable to packet drops than sensing.

Fig. 13 shows the MAE with -73 dBm noise and around 65% data loss ratio for both sensing and actuation. We see significant challenges on the control result, as the MAE reaches up to 3 °C in 3Hz control. Given such challenging wireless conditions, 5Hz control with S/G in Fig. 13(a) achieves the smallest MAE at a small variance. This again proves the value of asymmetric routing design in extremely challenging wireless control cases.

6. CONCLUSION

This paper explores the interactions of wireless routing and control through a cyber-physical case study on a wireless process control system. Our case study integrates a networked control design and a realistic wireless mesh network based on the WirelessHART standard. We observe the control system has different levels of resilience to packet loss for sensing and actuation. We then propose the asymmetric routing approach, where different routing strategies can be selected for sensing and actuation. We further present a cyber-physical co-design approach to tailor the routing strategies for sensing and actuation based on the resiliency of control to packet loss. Holistic simulations show that asymmetric routing designed based on the cyber-physical co-design approach can effectively enhance the resiliency of wireless control systems under a wide range of wireless conditions. Our results highlight the importance of co-joining the design of wireless network protocols with control in wireless control systems.

7. ACKNOWLEDGEMENT

This work was supported, in part, by NSF through grant 1320921 and the Fullgraf Foundation.

8. REFERENCES

[1] http://wsn.cse.wustl.edu/index.php/Testbed.
[2] Gurobi optimizer. http://www.gurobi.com/index.

[3] ISA100: Wireless Systems for Automation. http://www.isa100wci.org.

[4] Wireless Industrial Networking Alliance. http://www.wina.org.

[5] ZigBee alliance. http://www.zigbee.org.

[6] WirelessHART specification, 2007. http://www.hartcomm2.org.

[7] J. Araujo, A. Anta, M. Mazo, J. Faria, A. Hernandez, P. Tabuada, and K. Johansson. Self-triggered control over wireless sensor and actuator networks. In *International Conference on Distributed Computing in Sensor Systems and Workshops*, pages 1–9, 2011.

[8] K. J. Aström and R. M. Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[9] G. Baliga, S. Graham, L. Sha, and P. Kumar. Etherware: Domainware for wireless control networks. In *IEEE International Symposium on Object-oriented Real-time Distributed Computing*, 2004.

[10] A. Bemporad. Predictive control of teleoperated constrained systems with unbounded communication delays. In *IEEE Conference on Decision and Control*, volume 2, pages 2133–2138. IEEE, 1998.

[11] A. Bemporad, A. Casavola, and E. Mosca. Nonlinear control of constrained linear systems via predictive reference management. *IEEE Transactions on Automatic Control*, 42(3):340–349, 1997.

[12] B. Demirel, Z. Zou, P. Soldati, and M. Johansson. Modular co-design of controllers and transmission schedules in WirelessHART. In *IEEE Conference on Decision and Control and European Control Conference*, pages 5951–5958, 2011.

[13] C. E. Garcia, D. M. Prett, and M. Morari. Model predictive control: theory and practice—a survey. *Automatica*, 25(3):335–348, 1989.

[14] S. Gobriel, D. Mosse, and R. Cleric. Tdma-asap: Sensor network tdma scheduling with adaptive slot-stealing and parallelism. In *IEEE International Conference on Distributed Computing Systems*, pages 458–465. IEEE, 2009.

[15] S. Han, X. Zhu, A. Mok, D. Chen, and M. Nixon. Reliable and real-time communication in industrial wireless mesh networks. In *IEEE Real-Time and Embedded Technology and Applications Symposium*, 2011.

[16] J. Heo, J. Hong, and Y. Cho. Earq: Energy aware routing for real-time and reliable communication in wireless industrial sensor networks. *IEEE Transactions on Industrial Informatics*, 5(1):3–11, 2009.

[17] K.-D. Kim and P. Kumar. The importance, design and implementation of a middleware for networked control systems. In *Networked Control Systems*, pages 1–29. Springer, 2010.

[18] X. Koutsoukos, N. Kottenstette, J. Hall, P. Antsaklis, and J. Sztipanovits. Passivity-based control design for cyber-physical systems. In *International Workshop on Cyber-Physical Systems-Challenges and Applications*, 2008.

[19] B. Li, L. Nie, C. Wu, H. Gonzalez, and C. Lu. Incorporating emergency alarms in reliable wireless process control. In *ACM/IEEE International Conference on Cyber-Physical Systems*, 2015.

[20] B. Li, Z. Sun, K. Mechitov, C. Lu, D. Dyke, G. Agha, and B. Spencer. Realistic case studies of wireless structural control. In *ACM/IEEE International Conference on Cyber-Physical Systems*, April 2013.

[21] B. Li, D. Wang, F. Wang, and Y. Ni. High quality sensor placement for SHM systems: Refocusing on application demands. In *IEEE International Conference on Computer Communications*, 2010.

[22] X. Liu and A. Goldsmith. Kalman filtering with partial observation losses. In *IEEE Conference on Decision and Control*, volume 4, pages 4180–4186. IEEE, 2004.

[23] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen. Real-time wireless sensor-actuator networks for industrial cyber-physical systems. *Proceedings of the IEEE*, 2016.

[24] M. Pajic, S. Sundaram, J. Le Ny, G. J. Pappas, and R. Mangharam. Closing the loop: a simple distributed method for control over wireless networks. In *International Conference on Information Processing in Sensor Networks*, IPSN '12, pages 25–36, New York, NY, USA, 2012. ACM.

[25] K. Pister, P. Thubert, S. Dwars, and T. Phinney. Industrial routing requirements in low-power and lossy networks. Technical report, 2009.

[26] 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 *IEEE Real-Time Systems Symposium*, 2015.

[27] 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*, 2012.

[28] Y. Shi and H. Fang. Kalman filter-based identification for systems with randomly missing measurements in a network environment. *International Journal of Control*, 83(3):538–551, 2010.

[29] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004.

[30] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, and S. S. Sastry. Time varying optimal control with packet losses. In *IEEE Conference on Decision and Control*, volume 2, pages 1938–1943. IEEE, 2004.

[31] Z. Sun, B. Li, S. Dyke, and C. Lu. Evaluation of performances of structural control benchmark problem with time delays from wireless sensor network. In *EMI/PMC'12*, 2012.

[32] Z. Sun, B. Li, S. J. Dyke, C. Lu, and L. Linderman. Benchmark problem in active structural control with wireless sensor network. *Structural Control and Health Monitoring*, 23(1):20–34, 2016.

[33] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[34] C. Wu, D. Gunatilaka, A. Saifullah, M. Sha, P. B. Tiwari, C. Lu, and Y. Chen. Maximizing Network Lifetime of WirelessHART Networks under Graph Routing. In *IEEE International Conference on Internet of Things Design and Implementation*, April 2016.

[35] C. Wu, D. Gunatilaka, M. Sha, C. Lu, and Y. Chen. Conflict-aware real-time routing for industrial wireless sensor-actuator networks. In *Technical Report WUCSE-2015-005. All Computer Science and Engineering Research*, 2015. http://openscholarship.wustl.edu/cse_research/507/.

[36] Z. Zinonos, R. Silva, V. Vassiliou, and J. S. Silva. Mobility solutions for wireless sensor and actuator networks with performance guarantees. In *International Conference on Telecommunications*, pages 406–411. IEEE, 2011.