Winter 12-15-2015

Cyber-Physical Co-Design of Wireless Control Systems

Bo Li
Washington University in St. Louis

WASHINGTON UNIVERSITY IN ST. LOUIS

School of Engineering and Applied Science
Department of Computer Science and Engineering

Dissertation Examination Committee:
Chenyang Lu, Chair
Roger Chamberlain
Christopher Gill
Humberto Gonzalez
Roch Guerin
Abusayeed Saifullah

Cyber-Physical Co-Design of Wireless Control Systems

by
Bo Li

A dissertation presented to the
Graduate School of Arts and Sciences
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

December 2015
Saint Louis, Missouri

Table of Contents

List of Figures

List of Tables

Acknowledgments

First I want to deliver gratitudes to the dissertation committee for their great feedbacks.

Teamwork is the basis for this dissertation. Sincere appreciations are indebted to all colleagues who have worked with me and all Cyber-Physical System Laboratory (CPSL) alumni of Washington University.

Special thanks are given to Dr. Chenyang Lu for his superb guidance as an advisor and for his invaluable support as a friend.

I would like to delivery sincere thanks to my wife, my parents in law and their family for their support.

This dissertation is dedicated to the *Li* family for *always pursuing the goodness of human beings and the usefulness of science.*

Bo Li

Washington University in Saint Louis

December 2015

Dedicated to the *Li* family.

ABSTRACT OF THE DISSERTATION

Cyber-Physical Co-Design of Wireless Control Systems

by

Bo Li

Doctor of Philosophy in Computer Science

Washington University in St. Louis, 2015

Professor Chenyang Lu

Wireless sensor-actuator network (WSAN) technology is gaining rapid adoption in process industries because of its advantages in lowering deployment and maintenance cost in challenging environments. While early success of industrial WSANs has been recognized, significant potential remains in exploring WSANs as unified networks for industrial plants. This thesis research explores a cyber-physical co-design approach to design wireless control systems. To enable holistic studies of wireless control systems, we have developed the Wireless Cyber-Physical Simulator (WCPS), an integrated co-simulation environment that integrates Simulink and our implementation of WSANs based on the industrial WirelessHART standard. We further develop novel WSAN protocols tailored for advanced control designs for networked control systems. WCPS now works as the first simulator that features both linear and nonlinear physical plant models, state-of-art WirelessHART protocol stack, and realistic wireless network characteristics. A realistic wireless structural control study sheds light on the challenges of WSC and the limitations of a traditional structural control approach under realistic wireless conditions. Systematic emergency control results demonstrate that our real-time emergency communication approach enables timely emergency handling, while allowing regular feedback control loops to effectively share resources in WSANs during normal operations. A co-joint study of wireless routing and control highlights the importance of the co-design approach of wireless networks and control.

Chapter 1

Introduction

Wireless sensor-actuator network (WSAN) technology is gaining rapid adoption in process industries because of its advantages in lowering deployment and maintenance cost in challenging environments. While early success of industrial WSANs has been recognized, significant potential remains in exploring WSANs as unified networks for industrial plants. This thesis research explores a cyber-physical co-design approach to design wireless control systems.

To enable holistic studies of wireless control systems, we have developed the Wireless Cyber-Physical Simulator (WCPS), an integrated co-simulation environment that integrates Simulink and our implementation of WSANs based on the industrial WirelessHART standard. Under the WCPS framework, we have implemented a state-of-art WirelessHART protocol stack. We have further developed novel WSAN protocols and case studies for advanced wireless control systems.

Wireless Structural Control (WSC) systems can play a crucial role in protecting civil infrastructure in the event of earthquakes and other natural disasters. Such systems represent an exemplary class of cyber-physical systems that perform close-loop control using wireless sensor networks. Existing WSC research usually employs wireless sensors installed on small lab structures, which cannot capture realistic delays and data loss in wireless sensor networks deployed on large civil structures. The lack of realistic tools that capture both the cyber (wireless) and physical (structural) aspects of WSC systems has been a hurdle for cyber-physical systems research for civil infrastructure. This thesis research advances the state of the art through the following contributions. First, we developed the *Wireless Cyber-Physical Simulator (WCPS)*, an integrated environment that combines realistic simulations of both wireless sensor networks and structures. WCPS integrates Simulink and TOSSIM, a state-of-the-art sensor network simulator featuring a realistic wireless model seeded by signal traces. Second, we performed two realistic case studies each combining a structural model with wireless traces collected from real-world environments. The *building study* combines a benchmark building model and wireless traces collected from a multi-story building. The *bridge study* combines the structural model of the Cape Girardeau bridge over the Mississippi River and wireless traces collected from a similar bridge (the Jindo Bridge) in South Korea. Finally, we proposed a *cyber-physical co-design* approach to WSC that integrates a novel holistic scheduling scheme (for sensing, communication and control) and an Optimal Time Delay Controller (OTDC) that substantially improves structural control performance. Recently, we have further enhanced the wireless building control study, transforming it into a benchmark that can be used by the structural control community to explore and evaluate different wireless control approaches, allowing practitioners to easily generate and configure realistic nuisances such as network induced delay, data loss, measurement noise, and control constraints.

Recent years have witnessed adoption of wireless sensor-actuator networks (WSANs) in process control. Many real-world process control systems must handle various emergency alarms under stringent timing constraints in addition to regular control loops. However, despite considerable theoretical results on wireless control, the problem of incorporating emergency alarms in wireless control has received little attention. This thesis presents, to the best of our knowledge, the first systematic approach to incorporate emergency alarms into wireless process control. The challenge in emergency communication lies in the fact that emergencies occur occasionally, but must be delivered within their deadlines when they occur. To tackle this problem, we first propose efficient real-time emergency communication protocols based on slot stealing and event-based communication; We then conduct systematic studies on a coupled water tank system controlled over a 6-hop 21-node WSAN. Extensive results demonstrate the feasibility and efficacy of incorporating emergency alarms into wireless process control systems. WCPS and the case studies provide an enabling framework for exploring wireless process control design and hence represents a promising step toward practical wireless process control systems.

While there has been significant theoretical advancement in networked control design, there exist limited empirical results that combine control design with realistic WSAN standards. We further presents a cyber-physical case study that 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, a model predictive controller and an actuator buffer of recent control inputs. We observe 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.

To summarize, the main contributions of this thesis are four-fold: (1) WCPS is the first simulator that features both linear and nonlinear physical plant models, state-of-art WirelessHART stack, and realistic wireless network characteristics. (2) Our wireless structural control study sheds light on the challenges of WSC and the limitations of a traditional structural control approach under realistic wireless conditions. (3) Results in wireless emergency control demonstrates our real-time emergency communication approach enables timely emergency handling, while allowing regular feedback control loops to effectively share resources in WSANs during normal operations. (4) The wireless routing and control study further highlights the importance of the co-design approach of wireless networks and control.

Chapter 2

Wireless Cyber-Physical Simulator

This chapter describes the design and implementation of WCPS [1].

2.1 Introduction

Since deployments of large-scale wireless control systems are costly and not always feasible, wireless control systems have mostly been evaluated using wireless sensors installed on small lab structures. Unfortunately, such networks cannot capture the delays and data loss in wireless sensor networks deployed on real-world industrial environments. As such, there is a critical need for simulation tools and case studies that realistically model wireless characteristics and the plant dynamics.

There have been a number of wireless control simulators in the literature. However, most of them either suffer from the simplistic wireless radio model or the lack of a comprehensive protocol stack following real standards.

Truetime [20] is a well established control system simulator that enables holistic studies of CPU scheduling, communication and control algorithms. While Truetime supports wirelss networks, its wireless models are relatively simple and do not capture complex properties of wireless sensor networks such as probabilistic and bursty packet receptions and irregular radio properties [71]. In addition, Truetime implements wireless models within Simulink. While a native implementation may improve efficiency, it cannot leverage existing wireless simulators that implement sophisticated wireless models.

NCSWT [29] is a recent simulator for wireless cyber-physical systems. Instead of implementing wireless simulations natively, it integrates with the NS-2 simulator with support for wireless networks. While WCPS shares a similar federated approach to incorporate an existing wireless simulator, we choose to integrate WCPS with TOSSIM [43] which features a more realistic wireless sensor network model than NS-2. Despite its wide adoption as a network simulator, the wireless models in NS-2 suffers from being incapable of capturing the probabilistic and irregular packet receptions that are common in low-power wireless networks.

Gisoo in [12] is a recent simulator for wireless control systems based on Cooja, but the wireless link model in Cooja simulation remains to be improved.

To support holistic cyber-physical co-design and evaluation of wireless control systems we have developed Wireless Cyber-Physical Simulator (WCPS), an integrated simulator for wireless control systems. WCPS employs a federated architecture and integrated Simulink and TOSSIM. Simulink has been widely used for control system designs; TOSSIM is designed to simulate wireless sensor networks based on a realistic wireless link model validated in diverse real-world environments [42].

In WCPS, leveraging noise traces and statistical models, TOSSIM can capture complex temporal link dynamics that are crucial for realistic cyber-physical systems modeling. Moreover, the trace-driven simulation approach of TOSSIM enables us to study the impacts of different wireless environments. With WCPS, we have been able to conduct a series of realistic case studies based on real-world wireless traces, as well as novel wireless-control co-designs.

2.2 Architecture

WCPS follows a general wireless control system model. A wireless control system consists of a set of wireless sensors, a controller and a set of actuators. The sensors and actuators form a wireless mesh network connected with a base station hosting the controller. Following the centralized network management approach of the WirelessHART standard [21], WCPS employs a centralized network manager to compute routing and transmission schedules for the wireless sensor network.

WCPS employs a federated architecture that integrates (1) Simulink for simulating the physical system dynamics and the controller and (2) TOSSIM for simulating the wireless sensor network. Simulink has been widely used by control engineers to design and study control systems, while TOSSIM is specifically designed to simulate wireless sensor networks based on realistic wireless link models that have been validated in diverse real-world environments [42]. By combining Simulink and TOSSIM, WCPS provides an integrated environment to simulate wireless control systems in a holistic and realistic fashion.

As shown in Fig. 2.1, WCPS simulates the feedback control loop of the control system as follows. Sensor data is generated from physical models. Through a cross-platform function call from Simulink, sensor data is injected to the corresponding wireless sensors in TOSSIM. Following the routes and transmission schedule calculated by the network manager module, TOSSIM simulates the end-to-end wireless communication of the sensor data packets from

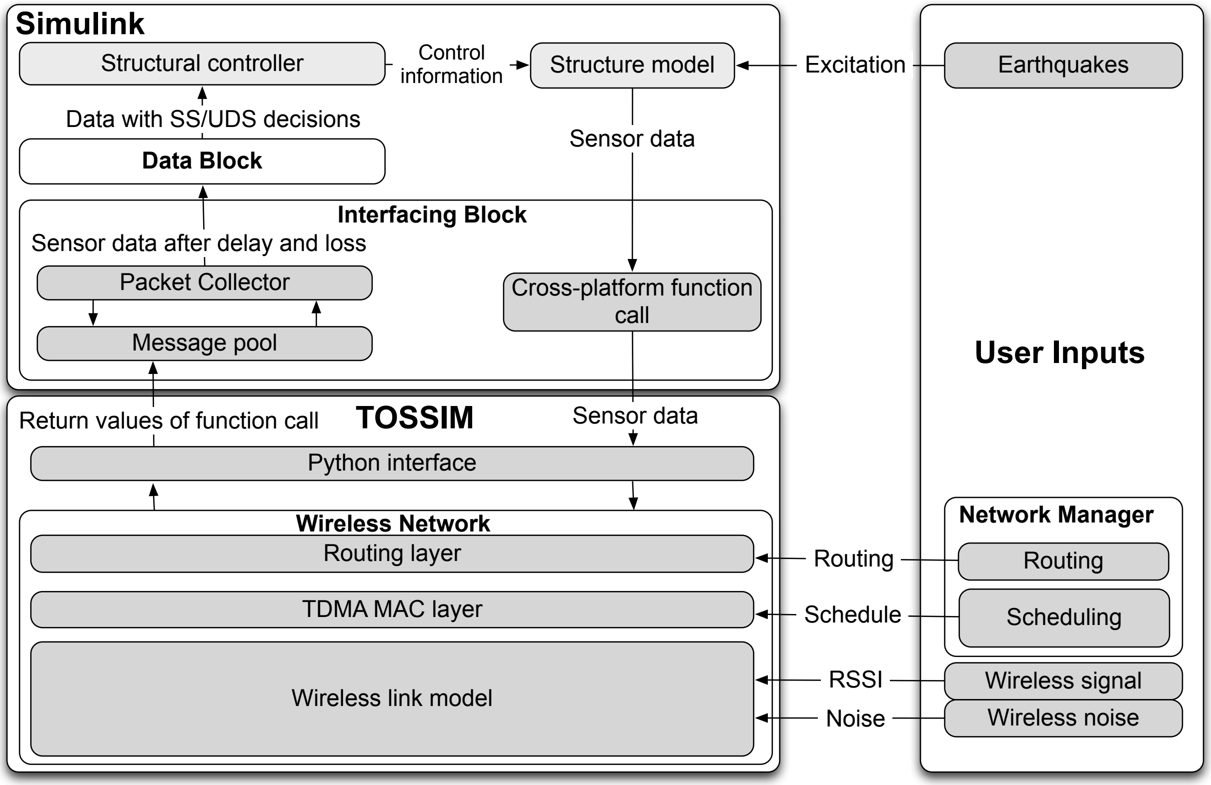


Figure 2.1: Component architecture of WCPS.

the sensors to the base station, and then return the packet delay and loss to the Interfacing Block in Simulink through the Python interface. The Packet Collector module then to extracts packet delivery information(the delay and loss) from the message pool of returned values in Simulink. Sensor data and their loss and delay are then provided to the Data Block, which then feed the sensor data to the controller at the right time based on the packet delay (if the packet is not lost). WCPS utilizes basic API (e.g., the dos, UNIX command) of MATLAB to do cross-platform function calls.

Fig. 2.1 also shows an example structural model, whose input includes excitation signals to the structure (e.g., acceleration caused by earthquakes) and wireless traces used as input to TOSSIM. The scheduler module calculates transmission schedules. Networking schedule is then deployed into the MAC layer code of wireless nodes and becomes effective after a TinyOS compilation. The TDMA MAC layer in WCPS is developed based on the MAC Layer Architecture (MLA) library [40] and further adapted for TOSSIM under TinyOS 2.x. Received Signal Strength Indication(RSSI) and wireless noises traces can be collected from real-world environments and provided to the wireless model in TOSSIM [42] for realistic wireless network simulations.

In Fig. 2.1, the interfaces between the Simulink model and TOSSIM are encapsulated as two MATLAB embedded functions in Simulink: the Interfacing Block and the Data Block. The Interfacing Block extracts delay and loss information from TOSSIM messages, and the Data Block decides what data will be used for discrete control during each sampling period. The federated architecture of WCPS provides great flexibilities to incorporate different physical models and implement alternative scheduling-control approaches.

2.3 Implementation

This sections first gives an overview of the WCPS implementation and further presents key WirelessHART components. We focus on architectural concepts here, and we refer interested users to `http://wcps.cse.wustl.edu` for more technical details.

We have updated WCPS to version 2.0 by implementing a WirelessHART protocol stack comprised of multi-channel communication, reliable Graph Routing and Dedicated/Shared time slotting supported by a robust Finite Sate Machine, and a centralized TDMA scheduler. Moreover, to accurately simulate aperiodic events, we have reorganized the simulator architecture to support event-driven co-simulation between TOSSIM and Simulink.

Under the new co-simulation architecture, TOSSIM is configured as a TCP/IP server that simulates the WSAN. Control models in Simulink connects to the TOSSIM server as a socket client. Since TOSSIM is a discrete event simulator, ideally 10^7 event ticks(time steps) corresponds to 1ms in our co-simulation, each client call from Simulink will advance TOSSIM by 10ms[1]. Configuring TOSSIM as a background server process allows effective data exchange between TOSSIM and Simulink while preserving all system states across client calls. To our knowledge, WCPS is the first simulator that can simulate high-fidelity interaction between TOSSIM and Simulink.

As the WirelessHART standard is gaining widespread adoption in process industries, it is important to study wireless control systems based on WirelessHART networks. As an integral part of WCPS 2.0, we have implemented a WirelessHART protocol stack in the TOSSIM simulator. Our WirelessHART stack realizes WirelessHART protocols at the routing and MAC layers, extends the TOSSIM link model to support multiple channels, and also implements a centralized network manager with a routing algorithm and transmission scheduler.

2.3.1 Centralized Scheduler

We have implemented a centralized network manager including a graph routing algorithm and a transmission scheduler. The transmission scheduler generates a superframe consisting of a sequence of time slots, each assigned a set of transmissions to occur on different channels. At run time the schedule is used in a cyclic fashion, repeating the superframe after reaching the end of the schedule.

[1] In practice, 10ms corresponds to 97656250 TOSSIM ticks, because 1K stands for 1024 in the adopted 32KHz alarm clock and hence 320 alarms (i.e., 10ms, 320/32KHz) lasts shorter than 10^8 ticks.

2.3.2 Multi-channel Communication

The original TOSSIM wireless model only supports a single channel. A key feature of WirelessHART is exploiting spectrum diversity by utilizing multiple channels supported by IEEE 802.15.4 radios. To support WirelessHART networks, we extend the TOSSIM simulator to support communication over multiple channels. The extended TOSSIM in WCPS 2.0 now can accept wireless signal and noise traces of multiple channels simultaneously and use them as inputs for simulations of wireless communication over multiple channels within a same time slot.

2.3.3 Graph Routing

WirelessHART supports two types of routing, Source Routing and Graph Routing, the latter of which is desirable for reliable communications. Source Routing provides a single route for each sensor/actuator; Graph Routing improves reliability through redundant routes, where each node in a graph route has two alternative receivers. In Graph Routing, two *dedicated slots* are first allocated for transmissions to the primary receiver, followed by a *shared slot* for the retransmission to the alternative receiver.

While WCPS 1.0 only supported Source Routing, we have implemented Graph Routing in TOSSIM for WCPS 2.0. This new routing approach enables us to explore reliable communication for wireless control systems. Further, we have implemented a robust Finite State Machine(FSM, see Fig. 2.2) that runs in the MAC layer of wireless sensors, which supports execution of Graph Routing, dynamic channel hopping, and TDMA schedules.

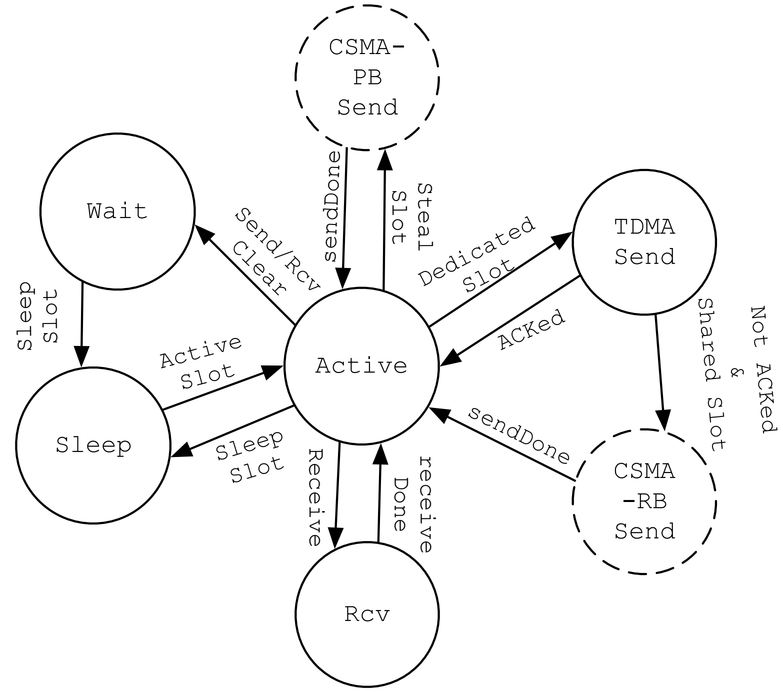


Figure 2.2: Finite State Machine for Wireless Sensors

2.3.4 Dedicated/Shared Slots

We implement both dedicated and shared slots at the MAC layer. In a *dedicated* slot, only one transmission is allowed on a same channel; in contrast, multiple transmissions can be scheduled in a same channel to contend in a *shared* slot. As in Fig. 2.2, in a dedicated slot, the owner does TDMA Send without channel assessment. On the other hand, CSMA with Random Backoff (CSMA-RB) is used in *shared* time slots, when different sensors compete for the transmission opportunity. TDMA Send and CSMA-RB can provide basic supports for WirelessHART communications.

We further devise CSMA Permanent Backoff (CSMA-PB) to support Slot Stealing. Sensors that send packets with CSMA-PB will *permanently* cease any transmission attempt *within a slot* when energy of others has been detected. That is, during Slot Stealing, *owner* will send a packet at the very beginning of the slot while the *stealer* will do channel assessment with an offset, followed by a backoff if channel is not clean, or a transmission otherwise.

2.4 Summary

In sum, WCPS for the first time enables both realistic wireless simulations and state-of-art WirelessHART protocol stack. Moreover, in-depth realistic Cyber-Physical co-designs and case studies are made possible by WCPS, which we show in the following chapters. More details about WCPS (including user manual, documentation and the source code) are available at `http://wcps.cse.wustl.edu`.

Chapter 3

Realistic Case Studies for Wireless Structural Control

3.1 Introduction

Wireless Structural Control (WSC) is a promising cyber-physical system technology for protecting our civil infrastructure in the event of earthquakes and other natural disasters. A WSC system employs a feedback control loop to control the dynamic response of a civil structure based on sensor data collected through wireless sensor networks [48]. As a representative example of cyber-physical systems, a WSC system requires holistic system designs that crosscut cyber (wireless and control) and physical (structural dynamics) components.

Since hardware deployments on large civil structures are costly and labor intensive [44, 59], to date WSC systems have mostly been evaluated using wireless sensors installed on small lab structures. Unfortunately, such networks cannot capture the delays and data loss in wireless sensor networks deployed on large civil structures in real-world environments. There is a critical need for simulation tools and case studies that realistically model wireless characteristics and the structural dynamics of WSC systems.

To meet this challenge in WSC research, we have developed a simulator specifically designed to support realistic simulations of wireless cyber-physical systems. Specifically, the contributions of this chapter are three-fold:

- First, we describe the *Wireless Cyber-Physical Simulator(WCPS)*, an integrated environment that combines realistic simulations of both wireless sensor networks and structures. WCPS integrates Simulink and TOSSIM, a state-of-the-art sensor network simulator featuring a realistic wireless model seeded by real-world wireless traces.
- Second, we present two realistic case studies each matching a structural model with wireless traces collected from real-world environments. The *building study* combines a benchmark building model and wireless traces collected from a multi-story building. The *bridge study* combines the structural model of the Cape Girardeau Bridge over the Mississippi River and wireless traces collected from a similar bridge (the Jindo Bridge) in South Korea. These case studies shed lights on the challenges of WSC and the limitations of traditional structural control approaches.
- Finally, we propose a *cyber-physical co-design* approach to WSC that integrates a holistic scheduling scheme (including sensing, communication and control) and an Optimal Time Delay Controller (OTDC), which substantially improves structural control performance in the presence of wireless communication delay and packet loss.

While this chapter focuses on WSC as case studies, the WCPS tool can be used to simulate other wireless control systems. Furthermore, our cyber-physical co-design approach and insights from the case studies can be generalized to other cyber-physical systems, especially large-scale wireless control systems. WCPS has been released as open-source software at `http://wcps.cse.wustl.edu`.

The rest of the chapter is organized as follows. Section 3.2 reviews related works. Section 3.3 describes explicit designs of the case studies. Section 3.4 details the cyber-physical co-design approach to wireless control. Section 3.5 presents the results of the case studies. Section 3.6 summarizes the chapter.

3.2 Related Work

Wireless Structural Health Monitoring(WSHM) research has been active in the past decade [32, 39]. Recent efforts for WSHM include: a distributed wireless sensing system for WSHM [53], the first wireless system deployed on a tower of over 600 meters tall [56] , a networked computing approach in WSHM [36], a high quality sensor placement study for WSHM [49, 50], a cyber-physical co-design of wireless distributed structural health monitoring [31], the largest wireless bridge monitoring system in the world [35] and Torre Aquila deployment for heritage building monitoring [19]. To name a few.

However, close-loop wireless control for civil structures is still in its infancy. While early efforts developed control algorithms and prototype wireless control systems [55, 76, 77], all the previous experiments were performed on small-scale lab structures. In the lab settings, wireless sensors within a single hop and experience no data loss due to physical proximity of the devices.

Wireless control has been studied with promising results in other domains [13, 15, 16, 57]. The challenge in realistic experimentation with WSC systems motivates the development of our WCPS and case studies based on real-world wireless traces and realistic structural models. Our work thus expands the field of wireless control to the civil infrastructure domain.

Moreover, WCPS can also be used to simulate other large-scale wireless control systems, and our scheduling-control co-design approach may be generalized to other wireless control systems. Research works using WCPS yet with more focus on civil structural analysis are introduced in [72, 74]

Truetime [20] is a well established control system simulator that enables holistic studies of CPU scheduling, communication and control algorithms. While Truetime supports wirelss networks, its wireless models are relatively simple and do not capture complex properties of wireless sensor networks such as probabilistic and bursty packet receptions and irregular radio properties [71]. In addition, Truetime implements wireless models within Simulink. While a native implementation may improve efficiency, it cannot leverage existing wireless simulators that implement sophisticated wireless models.

NCSWT [29] is a recent simulator for wireless cyber-physical systems. Instead of implementing wireless simulations natively, it integrates with the NS-2 simulator with support for wireless networks. While WCPS shares a similar federated approach to incorporate an existing wireless simulator, we choose to integrate WCPS with TOSSIM [43] which features a more realistic wireless sensor network model than NS-2. Despite its wide adoption as a network simulator, the wireless models in NS-2 suffers from being incapable of capturing the probabilistic and irregular packet receptions that are common in low-power wireless networks. Leveraging noise traces and statistical models, TOSSIM can capture complex temporal link dynamics that are crucial for realistic cyber-physical systems modeling. As the standard TinyOS simulator, TOSSIM has been widely used for wireless sensor network research and has been validated in diverse real-world environments [42]. Moreover, the trace-driven simulation approach of TOSSIM enables us to study the impacts of different wireless environments. We also provide the first set of realistic case studies based on real-world wireless traces, as well as a novel scheduling-control co-design approach to WSC.

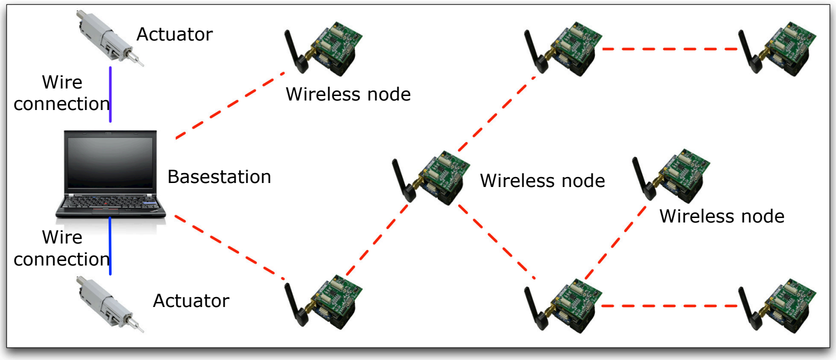


Figure 3.1: WSC System Model.

3.3 Case Study Design

This section presents the design of the case studies on wireless control of a three-story building and a bridge, respectively.

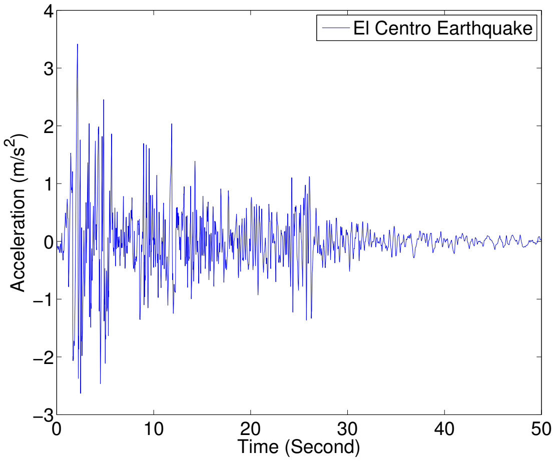


Figure 3.2: The El Centro earthquake as excitation signal of structural control [28] [70].

3.3.1 Excitation Signal

To study structural response to an earthquake, we use measurements from a real earthquake as the excitation signal in both case studies. As shown in Fig. 3.2 the excitation signal was recorded at the Imperial Valley Irrigation District substation in El Centro, California, during the Imperial Valley, California earthquake of May 18, 1940 [28]. The EI Centro earthquake last 50 seconds with a maximum acceleration of $3m/s^2$ at the beginning.

3.3.2 Design the Building Study

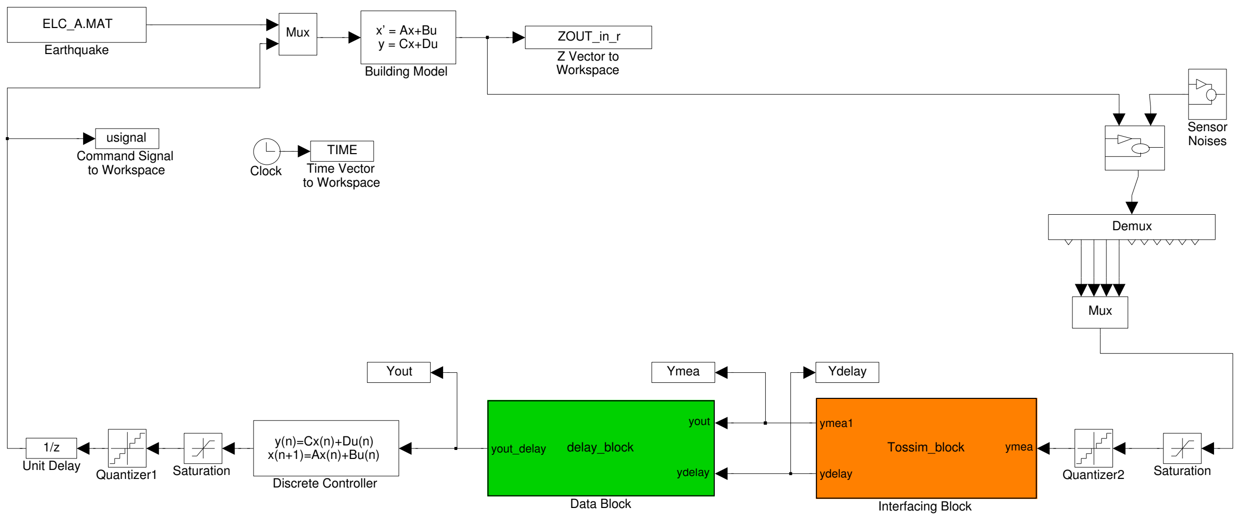


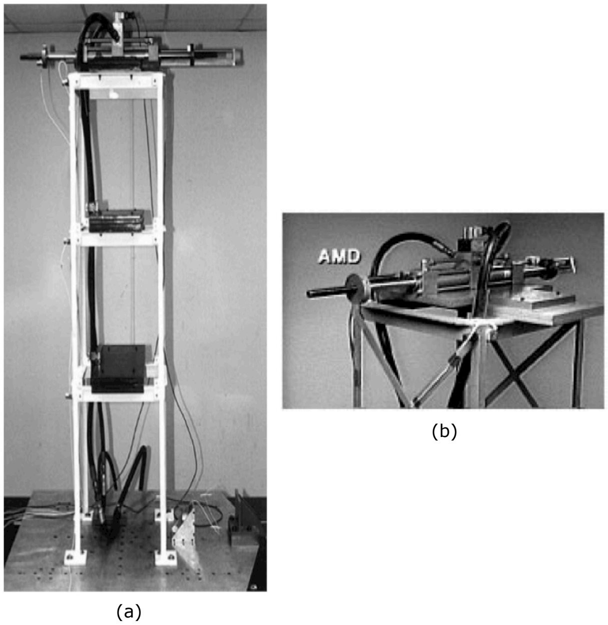
Figure 3.3: Simulink diagram for wireless building control.

Wireless trace collection

The wireless sensor network in this study comprises a base station and four distributed sensors. Wireless traces were collected from Bryan Hall of Washington University. The base station is located on floor 3, and the wireless sensors (TelosB motes [61]) are placed on floor

0, 1, 2, and 3, respectively. The sensors record RSSI and noise traces on channel 26 of the IEEE 802.15.4 radio. Each TelosB mote is equipped with a Chipcon CC2420 radio with its transmission power set to 0 dBm. Our measurements show that the wireless signal of the TelosB motes can go through at most two floors. As a result the sensor on floor 0 needs a multi-hop route to send data to the base station on floor 3.

Building model



(b)

(a)

Figure 3.4: Building control system [70]: (a) The 3-story test structure; (b) Active Mass Driver actuation system.

Our building model is based on a three-story test structure shown in Fig. 3.4(a) [70]. The test structure is subject to one-dimensional ground motion. The frame is constructed of steel, with a height of 158 cm. For control purposes, a simple implementation of an Active Mass Driver (AMD) is placed on the 3rd floor of the test structure (see Fig. 3.4(b)). The AMD actuation system has a single hydraulic actuator with steel masses attached to the ends of the piston rod. Since hydraulic actuators are inherently open-loop unstable, position feedback is employed to stabilize the actuator. The position of the actuator is obtained from an LVDT (linear variable differential transformer), rigidly mounted between the end of the piston rod and the third floor. The first three modes of the test structure are 5.81 Hz, 17.68 Hz and 28.53 Hz, with associated damping ratios of 0.33%, 0.23%, and 0.30%, respectively [70].

We developed a Simulink model (shown in Fig. 3.3) with reference to the steel test structure at a 1:1 ratio. The Simulink model is designed to simulate a real-world three-story building, with mapping ratios of: force = 1:60, mass = 1:206, displacement = 4:29 and acceleration = 7:2, and time =1:5. Since the time scales of the Simulink model and a real-world building have a 1:5 ratio, the natural frequencies of the model are approximately five times as large as those of a real-world building. Previous Simulink implementations of the building model was modeled as a continuous system and a time step of 0.0001 second was used to reduce integration errors. In WCPS we further discretize the Simulink model and perform step-by-step simulations with a step length of 1 ms, which corresponded to 5 ms in a real-world building. As the network used 10 ms slots for TDMA, a slot in the simulated wireless network therefore correspond to two run steps of the Simulink model.

As shown in Fig. 3.3, the structural response signal is first generated by the building model, then converted by the Analog to Digital Converter (ADC) to digital values, and fed to TOSSIM. TOSSIM delivers the sensor data along with its status (loss and delay) to the discrete controller. The output of the controller is then converted from digital values to analog signals by the Digital to Analog Converter (DAC). Eventually, sensor data with control information are fed back to the building model, which closes the control loop.

3.3.3 Design the Bridge Study

The bridge study simulates wireless control of the Cape Girardeau bridge in Missouri, USA. The cable-stayed bridge (see Fig. 3.8(b)) is the Missouri 74 Illinois 146 bridge spanning the Mississippi River near Cape Girardeau, Missouri, designed by the HNTB Corporation. Since no wireless sensors have been deployed on the bridge, we opt to use wireless traces collected from a wireless sensor network deployed on the Jindo bridge [35], South Korea, which shares similar dimensions (e.g., tower height and span range) and designs with the Cape Girardeau bridge. The sensor placement of the Jindo deployment is then mirror mapped onto the Cape Girardeau bridge. This approach takes advantage of the flexibility of WCPS to combine structural models and wireless traces from different (but similar) structures for integrated WSC simulations.

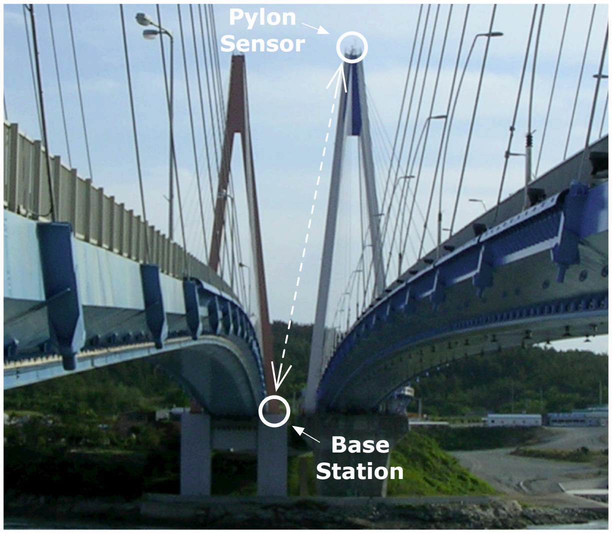


Figure 3.5: Wireless pylon sensor and base station placement on the Jindo bridge.

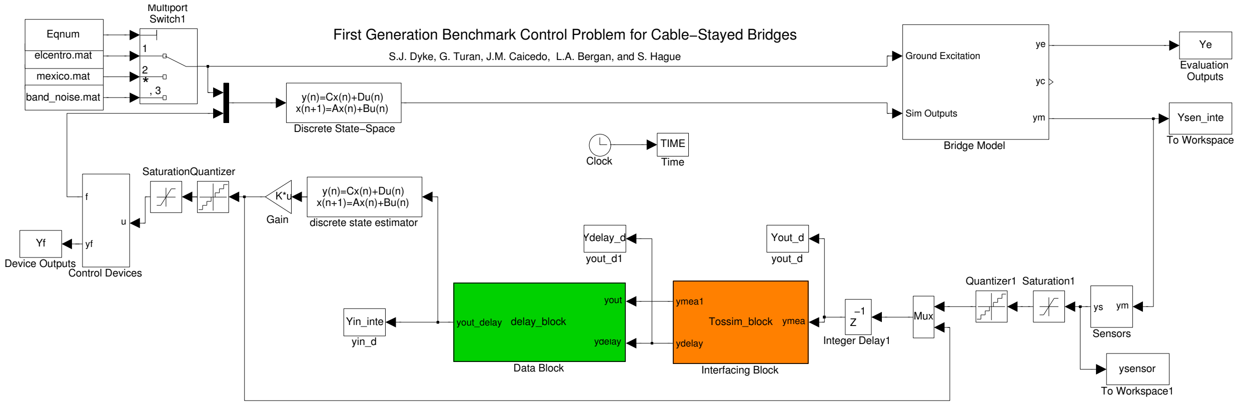


Figure 3.6: Simulink diagram for wireless bridge control.

Wireless trace collection

The Jindo deployment utilizes the MEMSIC Imote2 platform and a total of 113 Imote2 sensor nodes with 659 distinct sensor channels. Each node integrates the Imote2, the ISM400 sensor board, and a rechargeable battery supplied by a solar panel. Combined with the Illinois SHM Services Toolsuite [2], these powerful nodes allow for synchronized data collection, aggregation, synthesis and decision-making in real time. The system has successfully captured ambient traffic loading with peak acceleration ranging from less than 5 mg to over 30 mg. Further analysis of the data resulted in the successful identification of the first twelve modes of vibration on the deck, as well as tension forces of 10 cables with large tensile stresses [35]. To serve as input to the TOSSIM simulation, a subset of Imote2 nodes located along deck of the bridge and sensors on the top of the pylons are selected for wireless trace collection. With wireless traces collected from the Jindo bridge, we are able to build a 58-node routing network in TOSSIM for the Cape Girardeau bridge.

During our wireless trace collection on the Jindo bridge, sensors located on the top of the pylons pose a special case for trace collection. Whereas the sensors on the bridge deck form a connected graph, the pylon nodes are isolated. Due to the height of the pylons, these nodes are outside the maximum radio range of the deck nodes. In our Jindo deployment, pylon sensors are fitted with directional antennas, which are pointed away from the bridge deck towards a base station, located on the nearby Jindo Bridge(see Fig. 3.5). For the purpose of modeling a connected network, the real link quality measurements between the pylon sensor and base station node are mapped onto a virtual link in TOSSIM. During the network mapping, as the distances involved in Jindo bridge and Cape Girardeau bridge are similar and both bridges are in open areas, we assume this network mapping would correspond closely to a real wireless network setup.

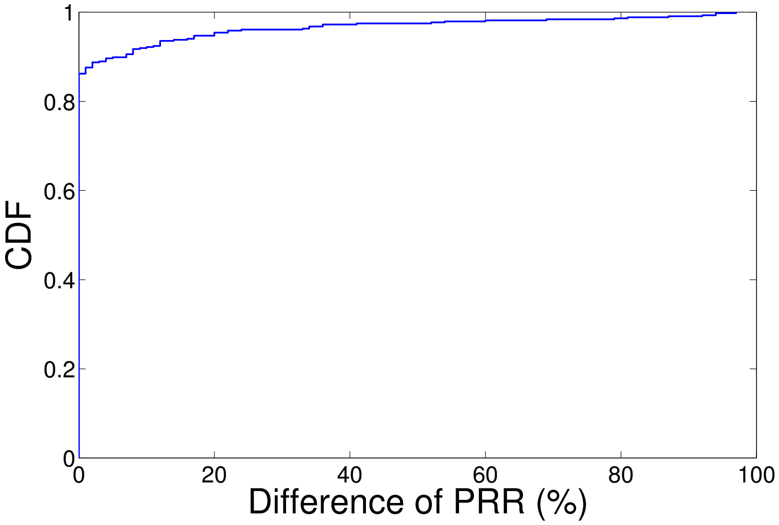


Figure 3.7: PRR Difference between field measurement and TOSSIM simulation.

Based on the structural model [28] we select sensor 240 and 353 located on the tow towers of the Cape Girardeau bridge, sensors 151 and 185 at the foots of towers and sensor 34 in the mid-span for structural control. Acceleration and displacement readings from the five selected sensors are sent to the base station located near sensor 185 using routes with the minimum ETX in the network. To test the accuracy of the TOSSIM simulation, we implement a test application in TOSSIM and compare the Packet Reception Ratio (PRR) of the simulation with that from the field test in Jindo. Fig. 3.7 plots the Cumulative Distribution Function (CDF) of the PRR difference between the field measurements and the TOSSIM simulations for all 467 wireless links. Of all the wireless links, over 85% of them have the same PRR in the field measurements and the simulation, indicating TOSSIM can deliver high fidelity link simulations based on real-world traces.

Bridge model

A high-fidelity Cape Girardeau bridge model (see Fig. 3.8(b)) was incorporated in WCPS for bridge control. A linear evaluation model was used for evaluation of the benchmark bridge model. However, the stiffness matrices used in this linear model are those of the structure determined through a nonlinear static analysis corresponding to the deformed state of the bridge with dead loads. Experimental study indicates that the longitudinal direction of the bridge is most destructive [28].

For control purposes the joints between the tower and the deck are disconnected and replaced by the control devices. As expected, the frequencies of this model are much lower than those of the nominal bridge model after incorporating the control device. The first ten frequencies of this second model are 0.1618, 0.2666, 0.3723, 04545, 0.5015, 0.5650, 0.6187, 0.6486, 0.6965, and 0.7094 Hz [28].

Fig. 3.6 shows the block diagram of the wireless bridge control system. Similar to the building control, the structural response of the bridge will go through ADC, a wireless network simulated in TOSSIM, and a discrete state estimator. The control inputs are converted by DAC to analog signals sent to the actuator.

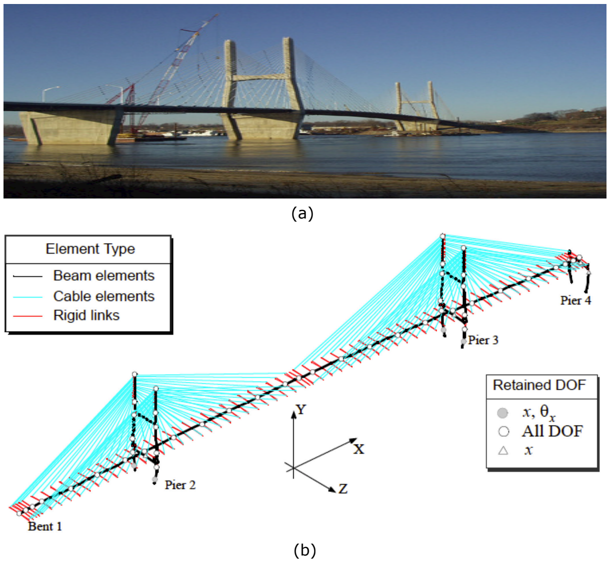


Figure 3.8: Cape Girardeau model in WCPS: (a) the Cape Girardeau bridge; (b) Simulink model of the Cape Girardeau bridge [28].

3.4 Wireless Control Approaches

We implement and compare two alternative control approaches to WSC. Instead of isolating the designs of the control algorithm and wireless sensor networks, we study holistic cyber-physical co-designs that integrate control algorithms and scheduling strategies for data collection, communication and utilization. As a baseline design the first approach integrates a traditional structural control algorithm called the Sample Controller (SC) [70] and a scheduling strategy that minimizes sensing delays. The second approach integrates the Optimal Time Delay Controller (OTDC) [26] and a novel scheduling strategy that lead to uniform sensing delays. Note that both SC and OTDC controllers were originally designed for wired structural control. Our work provides the first case studies of these control algorithms when applied to wireless structural control.

3.4.1 Sample Controller

SC employs the Linear Quadratic Gaussian (LQG) optimal control algorithm [70]. LQG is a combination of linear quadratic estimator (LQE) and linear quadratic regulator (LQR). The cost function to be minimized in SC is defined in Equation 3.1, where x^r is the reduced states vector, u is the control force, C_r^Z and D_r^Z are system matrices for the regulated output vector, and Q and R are weighting matrices. More details of SCc can be found in [70] and [28] .

$$\begin{aligned} J = \lim_{\tau\to\infty} \frac{1}{\tau} E \left[\int_0^\tau \left\{ \left(C_r^Z x^r + D_r^Z u \right)^T Q \left(C_r^Z x^r + D_r^Z u \right) \right.\right. \\ \left. +u^T R u \right\} (dt)] \end{aligned} \tag{3.1}$$

Specifically for SC, we implement a Sequential Scheduler (SS) which schedule one packet each TDMA time slot. Key data utilization mechanism for SS is to transmit the latest available data. For example, given vector $[y_1^x, y_2^x, y_3^x, y_4^x]$ as the data collected by sensor 1, 2, 3, 4 at the beginning of slot x, sensor 3 at the beginning of slot 2 (see Fig. 3.9) chooses to transmit y_3^2 instead of y_3^1 because y_3^2 is the latest reading. Similarly, sensor 2 chooses to transmit y_2^3 at the starting point of slot 3 because y_2^3 is the most up-to-date reading. SS makes sure that only latest sensor data is used for control, but it also sacrifices sensing synchronizations.

Fig. 3.9 illustrates working mechanism of SC and SS with a four-sensor network example. Sensor 1, 2, 3, 4 are located on floor 1, 2, 3, 4 of a building while the base station is located on the 4th floor. Since sensor 4, 3, and 2 have 1-hop distance to the base station, each needs one time slot to its data to the base station, while sensor 1 needs two because it is

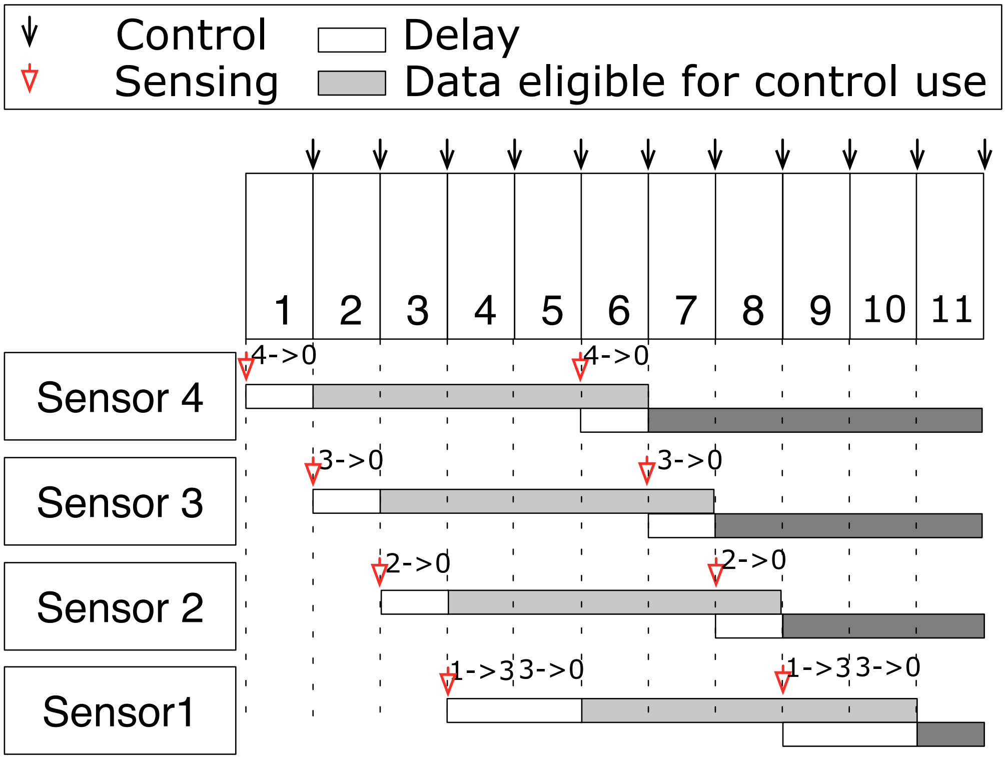


Figure 3.9: Example of baseline SC controller with Sequential Scheduler.

two hop away from the base station. SC control (denoted by dark arrows in Fig. 3.9) starts at the end of slot 1 with data vector $[0, 0, 0, y_4^1]$ as only the first reading (collected in slot 1) of sensor 4 has arrived. By the end of the slot 2, SC computes its control input with $[0, 0, y_3^2, y_4^1]$ because the second reading (collected in slot 2) of sensor 3 has arrived. By the end of slot 3, SC uses $[0, y_2^3, y_3^2, y_4^1]$ to compute its control input. The same data vector is used again at the end of slot 4 because no reading from sensor 1 has arrived yet. By the end of slot 5, SC uses $[y_1^4, y_2^3, y_3^2, y_4^1]$ for control, which completes a communication cycle from all sensors. Starting from slot 6, another cycle of data collection and control occur using the same schedule. Intuitively, the combination of SC and SS aims to reduce the delay of the sensor data used for control. Henceforth, we refer to the control scheme combining SC and SS as SC for simplicity.

3.4.2 Optimal Time Delay Controller

OTDC [26] was originally designed for constant-delay system as shown in Equation 3.2, where l is the time delay. OTDC is designed to minimize the cost function J by selecting an optimal control force p_d in Equation 3.3. However, in a wireless sensor network data from different sensors will be delivered to the controller at different delays. To use OTDC to WSC effectively we design a novel scheduling strategy called the Uniform Delay Scheduler (UDS) that pushes sensor data to the controller at uniform delays.

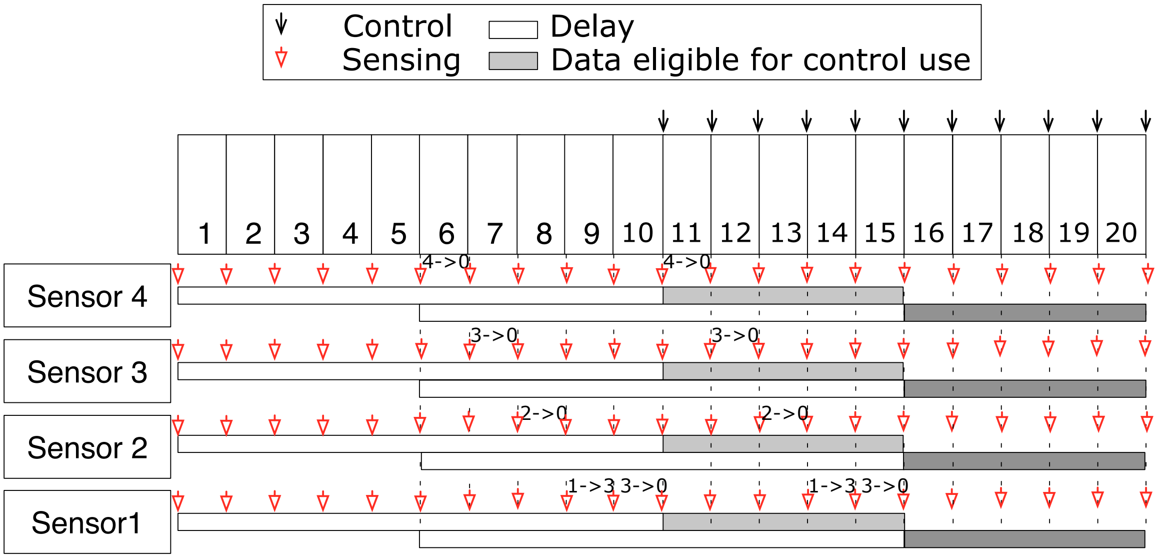


Figure 3.10: Example of OTDC-1 with UDS scheduler.

$$z\left[k+1\right] = Az\left[k\right] + Bp_d\left[k-l\right] \tag{3.2}$$

Fig. 3.10 illustrates the schedule produced by UDS for the same four-sensor example used in the last subsection. UDS first buffers one batch of data (five readings for each sensor). Afterwards, a cycle of five time slots is used to deliver the batched data to the base station. By the end of slot 10, OTDC starts with data vector $[y_1^1, y_2^1, y_3^1, y_4^1]$, followed by $[y_1^2, y_2^2, y_3^2, y_4^2]$ in the next time slot, and $[y_1^3, y_2^3, y_3^3, y_4^3]$ in the time slot after. This pattern continues till the end of slot 14. In time slot 15 OTDC starts a new cycle and uses $[y_1^6, y_2^6, y_3^6, y_4^6]$ by the end of time slot 15. Under UDS data from different sensors shares a uniform network delay (10 time slots, or 100 ms in Fig. 3.10). UDS therefore trades one cycle of delay for uniform delays among sensors. This feature makes UDS particularly suitable for OTDC specifically designed for systems with constant delays. As shown in our case studies this scheduling-control co-design approach leads to an effective WSC system.

$$J|_{p_d} = \sum_{k=l}^{\infty} \left(z_d^T\left[k\right] Q z_d\left(k\right) + {p_d}^T\left[k-l\right] R p_d\left[k-l\right] \right) \tag{3.3}$$

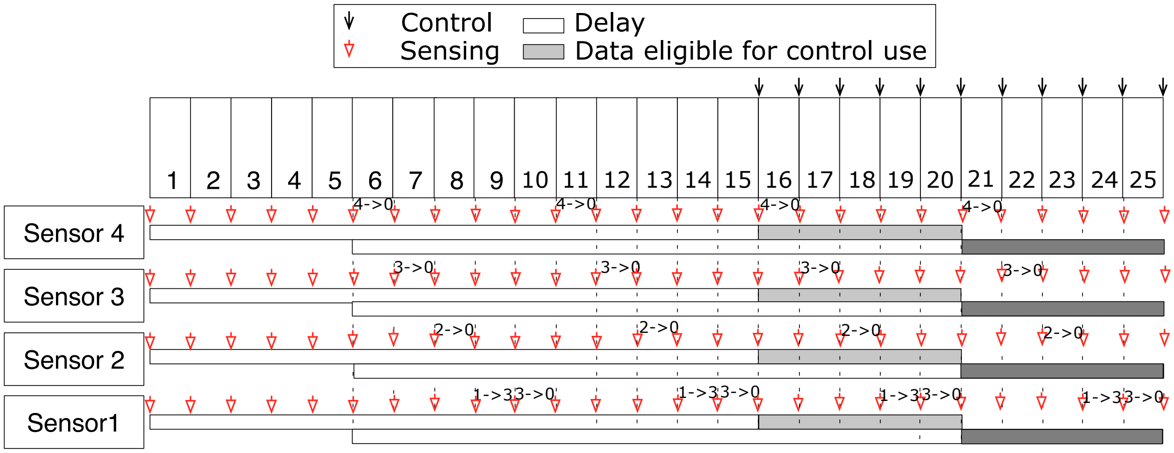


Figure 3.11: Example of OTDC-2 with UDS scheduler.

Another challenge introduced by wireless networks is packet loss. Since the basic version of UDS described above schedules only one transmission attempt for each sensor reading, a packet drop means losing one batch of readings (e.g., 5 readings in Fig. 3.10). To deal with packet loss we extend UDS to support multiple transmissions per sensor reading. Henceforth we use OTDC-k to denote a design that integrates OTDC and UDS that transmit each sensor reading k times. Due to the limited bandwidth of wireless sensor networks, OTDC-k retransmit sensor data from earlier cycles by merging them into packets of later cycles. The simple packet-merging mechanism in OTDC-2 avoids costly retransmissions of entire packets (e.g., as in WirlessHART). The number of batches that can be merged into a packet merging is limited by the packet payload size, e.g., over 100 bytes for IEEE 802.15.4 packets.

For example, in Fig. 3.11, though the batch of data $[y_1^1, y_1^2, y_1^3, y_1^4, y_1^5]$ from sensor 1 may be available by the end of slot 10, OTDC-2 waits for one more cycle (five time slots) before pushing the sensor data to the controller. At the same time $[y_1^1, y_1^2, y_1^3, y_1^4, y_1^5]$ is merged with $[y_1^6, y_1^7, y_1^8, y_1^9, y_1^{10}]$ and goes through another cycle of network communication. In this way, $[y_1^1, y_1^2, y_1^3, y_1^4, y_1^5]$ are transmitted twice and thus has better chance to be successfully delivered. OTDC-2 therefore trades additional network delay for higher reliability, while maintaining uniform delays across sensors. Increasing k in OTDC-k increases network delays while achieving higher reliability.

3.5 Results of Case Studies

This section presents the results of the case studies under realistic structural and wireless models in WCPS. In both case studies we compare the performance of alternative wireless control approaches, SC and OTDC-1. We also study the tradeoff between delay and data loss by comparing OTDC with different numbers of retransmissions (OTDC-1, OTDC-2 and OTDC-3).

3.5.1 Wireless Building Control

The building remains stable under all control approaches throughout this case study. To evaluate the control performance we use three categories of metrics: resource requirement, structural response, and constraints of the control system. We refer interested readers to [70] for detailed definitions of these metrics. We perform simulations using four different wireless control approaches (SC, OTDC-1, OTDC-2, and OTDC-3). Experimental results presented below are from 25 simulations for each control approach and each simulation lasts 10,000 control steps.

Fig. 3.12 shows the end-to-end packet delivery ratio of the wireless network. The end-to-end delivery ratio means the fraction of packets from the sensors that are successfully delivered to the controller. As shown in Fig. 3.12 Sensor 1 has the lowest delivery ratio because it has a 2-hop route to the controller. Recall that OTDC-1 does not perform any retransmission, while OTDC-2 and OTDC-3 performs retransmit each packet once and twice, respectively. Under OTDC-1 Sensors 1 and 4 have delivery ratios of 70% and over 95%, respectively. As expected more retransmissions improve the deliver ratios of all sensors at the cost of longer delays as described earlier.

Fig. 3.13 shows the resource requirement of different control approaches. OTDC-k approaches (see Fig. 3.13(a)) consistently require less control power than SC. As k increases, OTDC-k requires slightly less control power. Similarly, as shown in Fig. 3.13(b), OTDC-1 reduces control force by 80% when compared to SC. The differences in control force among

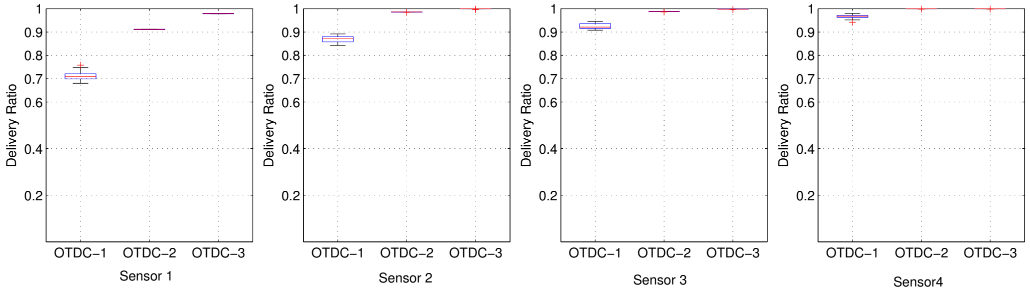


Figure 3.12: End-to-End Packet Delivery Ratio of Sensors in Building Study

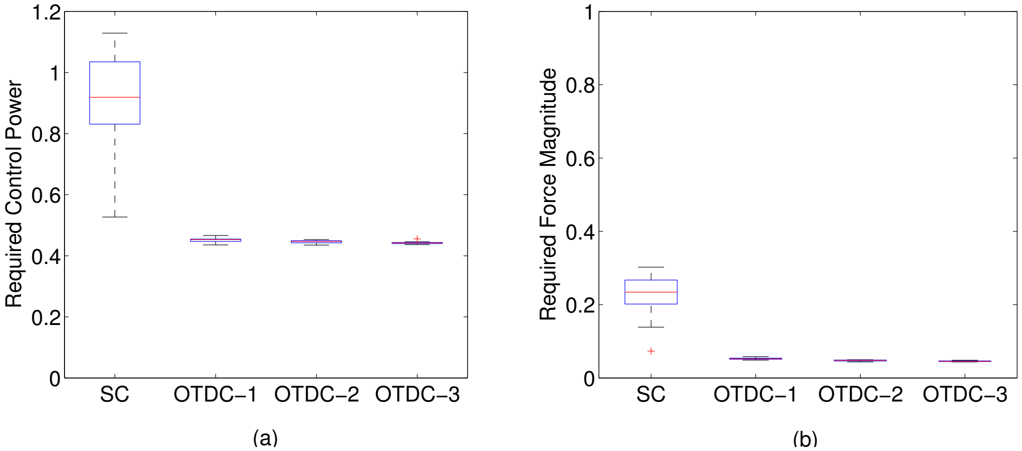
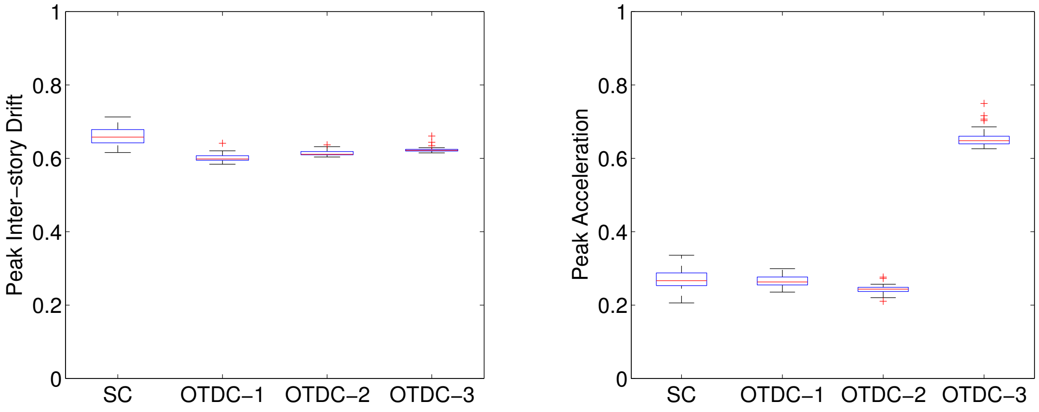


Figure 3.13: Required Resource for Wireless Building Control: (a) Required Control Power; (b) Required Force Magnitude.

different OTDC-k approaches are negligible. The results that OTDC-1 outperforms SC in both metrics indicate resource requirements are more sensitive to data synchronization than to sensing delays in this building control system. OTDC-k with larger k results in negligible reduction of control power and force, indicating resource requirements are not sensitive to network reliability in this case study.

The control performance regarding structural response is shown in Fig. 3.14. In term of peak inter-story drift in Fig. 3.14(a), OTDC-k achieves more reduction in inter-story drift than SC. Interestingly, higher k in OTDC-k *increases* peak inter-story drift. Recall a higher k



(a)

(b)

Figure 3.14: Structural Response under Wireless Building Control: (a) Peak Inter-story Drift; (b) Peak Acceleration.

leads to higher communication reliability but longer sensing delay. Inter-story drift is thus more sensitive to sensing delays than to data loss in this case study. Similarly, as shown in Fig. 3.14(b), OTDC-3 causes worse peak acceleration than all the other approaches. Hence, building structural responses are more sensitive to sensing delays than to data loss. In addition, OTDC-1 only slightly outperforms SC, which indicates limited impact of data synchronization on structural response.

The control performance regarding control system constraints is shown in Fig. 3.15. Fig. 3.15(a) and (b) plot the actuator peak acceleration and Root Mean Square(RMS) acceleration, respectively. On both metrics OTDC-k approaches result in smaller actuator accelerations than SC. As k increases, we can see gradual decreases in peak and RMS accelerations, indicating that these metrics are more sensitive to improvement of communication reliability than to longer sensing delays. In addition, the comparison between OTDC-1 and SC shows that the better data synchronization under OTDC-1 has a larger impact than sensing delays.

In summary, we observe complex tradeoffs among data synchronization, sensing delay and communication reliability in wireless building control. Overall the OTDC approach combining a constant-delay control design and a scheduling scheme achieving data synchronization outperforms the SC approach that minimizes sensing delay without data synchronization. This result highlights the efficacy of our control-scheduling co-design approach to wireless control. Moreover, the design of the wireless communication protocol involves tradeoff between communication delay and data loss, with each having stronger influence on different performance metrics. For our specific building study OTDC-1 and OTDC-2 outperforms OTDC-3. The complex tradeoff among multiple design aspects confirms the importance of a realistic simulation tool in designing wireless control systems.

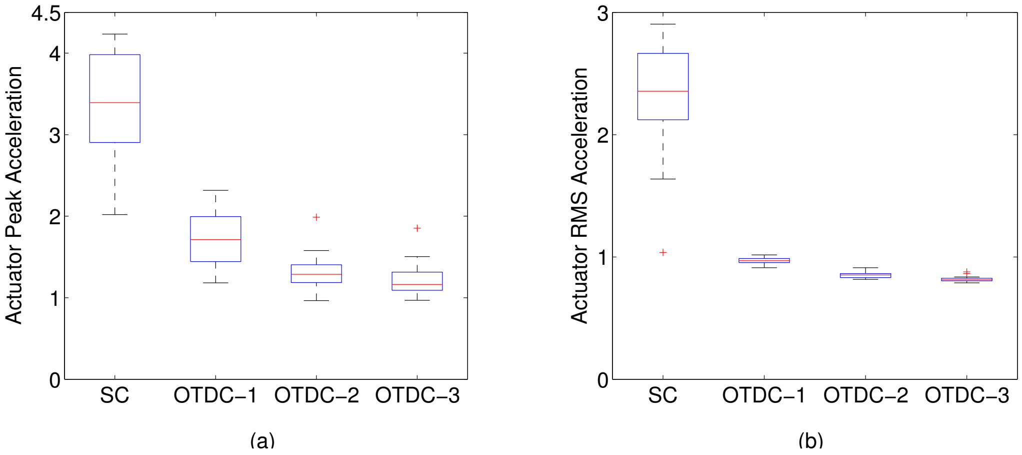


Figure 3.15: System Constraint under Wireless Building Control: (a) Actuator Peak Acceleration; (b) Actuator RMS Acceleration.

3.5.2 Wireless Bridge Control

Given the similarities in both the structural and wireless characteristics shared by the Jindo bridge and the Cape Girardeau bridge, wireless traces collected from the Jindo bridge were used to simulate the wireless sensor network used to control the Cape Girardeau bridge.

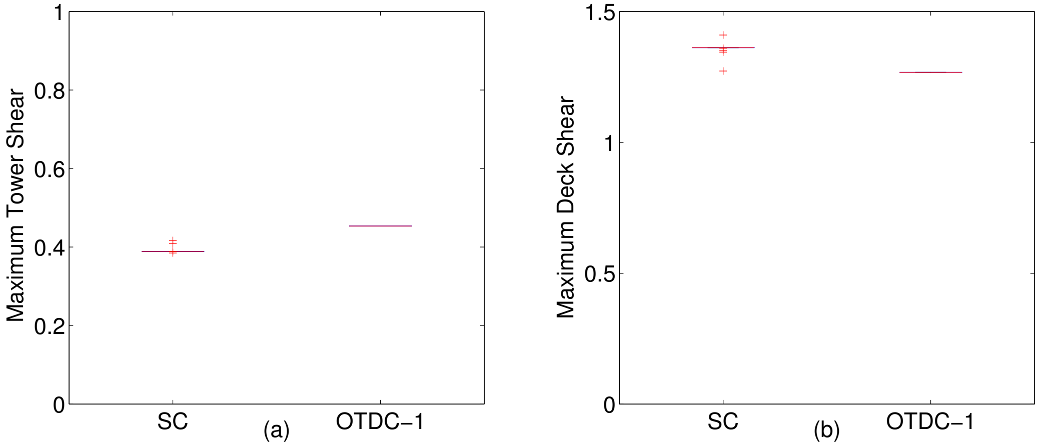


Figure 3.16: Maximum Shear Force in Wireless Bridge Control: (a) Maximum Tower Shear; (b) Maximum Deck Shear.

The longest routing path is 3-hop. The results presented below are from 25 simulations for each control case and each simulation lasts 10,000 control steps. To mitigate large delays caused by large amount of packet deliveries for multiple sensors, network scheduling in bridge control adopts an in-network aggregation approach [23] through packet merging.

The bridge network is highly reliable (99% PRR for almost all links with the Jindo trace) due to the relatively clean wireless environment on the Jindo bridge as well as the fact that the Jindo deployment has line-of-sight sensor placement and strong radio antennas. As such, retransmission is not needed to achieve reliable communication. Therefore we only present the results of SC and OTDC-1 in this case study.

Since buildings and bridges have distinct structural properties, we adopt three different sets of metrics for performance evaluation. The metrics include maximum shear force, normalized shear force and required control power. We refer interested readers to [28] for the mathematical details of the metrics.

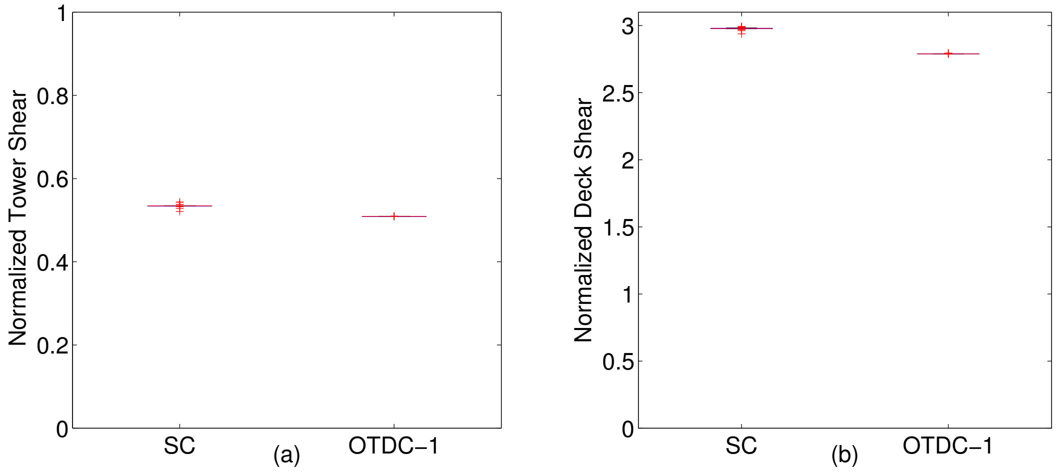


Figure 3.17: Normalized Shear Force in Wireless Bridge Control: (a) Normalized Tower Shear; (b) Normalized Deck Shear.

Fig. 3.16 plots the maximum shear force at the tower and the deck of the bridge. A smaller shear force is desirable in structural control. SC performs slightly better in reducing the maximum tower shear while OTDC-1 performs slightly better for reducing the maximum deck shear, respectively. Fig. 3.17 plots the normalized shear force at the tower and the deck. OTDC-1 slighly outperforms SC for reducing normalized shear force.

While OTDC-1 did not show significant advantage over SC in term of shear force, it reduces both the required maximum control power and the total power requirement by nearly 50% compared to SC (see Fig. 3.18). This result again demonstrated the effectiveness of the control-scheduling co-design approach adopted by the OTDC design.

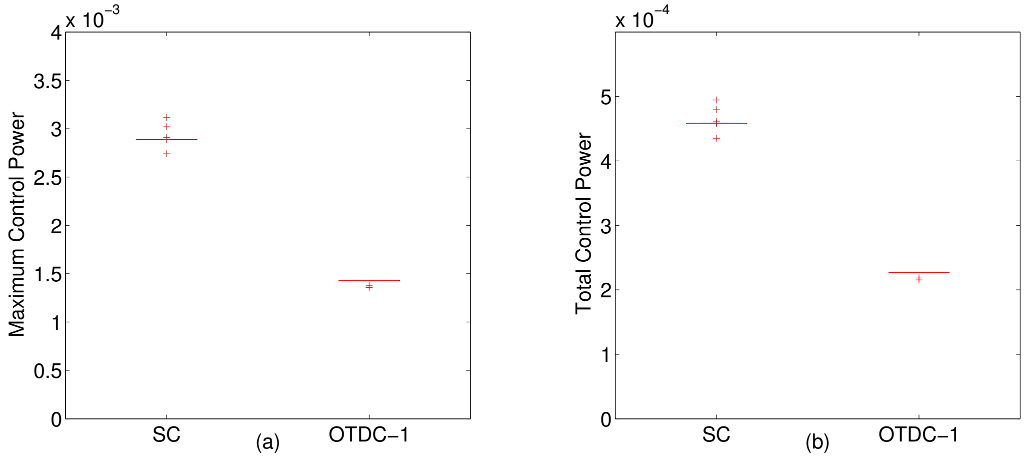


Figure 3.18: Control Power Requirement Performances for Wireless Bridge Control: (a) Maximum Control Power; (b) Total Control Power.

3.6 Summary

Wireless Structural Control (WSC) systems are a representative class of cyber-physical systems that have the promise to protect our civil infrastructure in the event of earthquake and other natural disasters. To develop WSC systems it is critical to capture both the cyber aspects (wireless communication and control) and the physical aspects (structural dynamics) through realistic and holistic simulations. We have developed the Wireless Cyber-Physical Simulator (WCPS) that integrates a high-fidelity wireless simulator (TOSSIM) and a standard control system simulator (Simulink). With WCPS, we performed two case studies on structural control systems. Each case study combines a realistic structural model and wireless simulations driven by traces collected from real-world deployments. Our case studies leads to three important insights. First, there exist complex tradeoffs among data synchronization, sensing delay, and network reliability under realistic wireless structural control settings. Second, a realistic, integrated wireless control simulator like WCPS is critical in exploring the design tradeoffs in wireless control design. Finally, a control-scheduling co-design approach is effective in wireless control design. In both case studies the integration of a contant-delay control design and a scheduling scheme achieving data synchronization lead to substantial improvement in control performance when compared to a traditional control design. Our cyber-physical simulation methodology and scheduling-control co-design approaches presented in this work not only represent a promising step toward smart civil infrastructure, but also provide useful insights and tools that can be generalized to other cyber-physical systems employing wireless control. The WCPS tool and the case studies have been released as open source software at `http://wcps.cse.wustl.edu`.

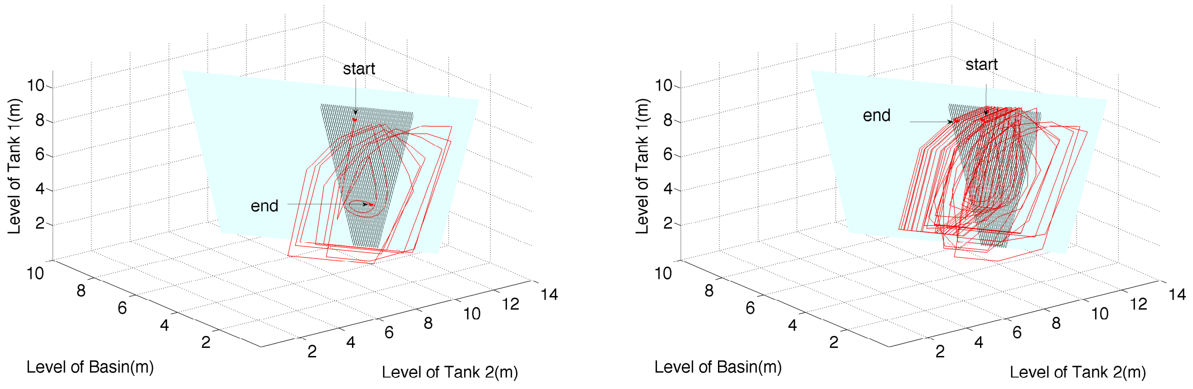
Chapter 4

Incorporating Emergency Alarms in Wireless Process Control

4.1 Introduction

Wireless sensor-actuator network (WSAN) technology is gaining adoptions in process industries due to their advantage in lowering deployment effort in challenging environments. Industrial standard organizations such as ISA, HART, WINA and ZigBee, have been actively pushing the application of wireless technologies in industrial automation [33]. While early success of industrial WSANs focused on monitoring applications, there is significant value in exploring WSANs for process control applications to take full advantage of wireless technology in industrial plants.

A wireless process control system employs feedback control loops to control the dynamic response of industrial processes through communications in a shared WSAN. Since communication delays and packet drops may lead to severe degradation of control or even instability of the system, it is critical to support real-time and reliable communication.



(a) Ideal Control

(b) Wireless Control

Figure 4.1: Control State Trajectory of A Coupled Water Tank System: Ideal vs. Wireless

Fig. 4.1 shows system state trajectories of wireless control versus ideal control for a water tank system. Here ideal control means the case where communications occur with no delay and no loss. Fig. 4.1(a) shows the ideal control system goes back to the shaded feasible region, reaches the set point and succeeds in control in a couple of rounds. In contrast, wireless control in Fig. 4.1(b) clearly takes more rounds and eventually fails to stabilize the system within the time limit, due to control packet drops and the communication delay. Hence, wireless control faces many challenges due to link failures and time varying delays (e.g., delay caused by retransmissions). In the face of emergencies, the control problem become even harder.

This work systematically investigates how to incorporate emergency alarms in wireless control systems, a problem that is critical in many real-world process plants, but yet received little attention in the literature. Simple controllers commonly used in industrial process control applications, such as PID or ON/OFF, can sometimes produce undesired responses, since they do not explicitly handle safety constraints. For this reason, it is also common to add safety measures, usually in the form of digital binary signals, to handle special situations that lead to physical damage of the plant, or even danger to the human operators. These signals take the form of tripwires around dangerous zones, emergency triggers for human operators, or contact switches in water tanks, among many others. In a wireless control scenario, as the one described in this chapter, these emergency signals must be transmitted using the same infrastructure as the regular control signals.

Despite significant body of theoretical results on real-time communication protocols and scheduling for WSANs, earlier research has largely focused on regular feedback control loops that employ communications in a periodic or event-driven fashion. Emergency alarms presents a challenging communication and control design problem. While emergencies occur sporadically, it is critical to communicate and handle emergency alarms in a timely fashion when they happen. Moreover, the lack of realistic simulation tools in compliant with state-of-art WSAN standards (e.g., WirelessHART [3]) has largely prevented in-depth wireless process control research. In this chapter we present the following contributions to address these challenges:

- We implement an WirelessHART protocol stack in the TOSSIM wireless simulator, on top of its realistic link model for IEEE 802.15.4 radios.
- We build the Wireless Cyber-Physical Simulator 2.0 that integrates Simulink and TOSSIM for holistic wireless control study while supporting both periodic and event-based simulations.
- We propose periodic and event-based real-time emergency communication protocols for WSAN.
- We construct a systematic case study on a coupled water tank system controlled over a 6-hop WSAN.

The rest of the chapter is organized as follows. Section 4.2 discusses related works. Section 4.3 presents the system model of a wireless control system. Section 4.4 introduces the wireless design. Section 4.5 details the control design. Section 4.6 presents systematic evaluation results and Seciton 4.7 summarizes the chapter.

4.2 Related Work

Promising results have been reported in the wireless control literature. Case studies on wireless structural monitoring and control systems were reported in [50,53,72,73]. Real-time transmission scheduling and co-designs for WSANs has been investigated in [24,63,65,66,81]. Reliable routing algorithms for WSANs have been presented [33]. Unfortunately none of these works considered emergency alarms.

Networked control systems have received tremendous attentions [16]. Discrete-time Kalman filters have been proposed for state estimation based on intermittent observation [68]. Co-design of transmission scheduling and controllers was explored in [27]. Passivity-based control architecture was proposed for cyber-physical systems [41]. Fault-tolerant control under uncertainties and time delays was studied in [25]. These work did not consider emergency alarms either.

Progress on WSAN protocols have been reported. Self-triggered control approaches have been developed for wireless networks [13, 75]. A distributed control approach has been proposed for WSANs [58]. These works however focused only on regular feedback control loops. Our work complements them by investigating emergency alarms alongside regular feedback loops.

Because large-scale real-world wireless control systems are not always available, a number of simulation tools have been developed. Truetime [20] is a well established control system simulator that enables holistic studies of CPU scheduling, communication and control algorithms. NCSWT [29] is a useful simulator for wireless cyber-physical systems. None of these simulators implemented WirelessHART, which is widely used in the industry. Gisoo in [12] is a recent simulator for wireless control systems based on Cooja, but the wireless link model in Cooja simulation remains to be improved. WCPS [47] connects Simulink and TOSSIM. WCPS 2.0 as a further development in this chapter has incorporated substantial changes including a new WirelessHART protocol stack. Finally, WCPS 2.0 can effectively simulate aperiodic emergency events.

Despite the fact that fault detections have been heavily studied in wireless sensor network [22, 82] and process control [10, 78], efforts in this study are orthogonal to existing fault tolerant literatures because those efforts mostly detect and isolate faults caused by sensor or controller failures rather than wireless link failures. Challenges arising from wireless link failures remain a problem even after detection and isolation of sensor or control failures. As such, reliable network protocols in this study is a natural complement for existing fault tolerant literature.

4.3 System Model

We consider a wireless control system consisting of a physical plant, a centralized controller and a WSAN. Sensors and actuators communicate through a multi-hop WSAN forming a multi-hop wireless mesh network. In the *sensing phase*, sensors send their measurements to the controller. Control commands issued by the controller will be sent to actuators in the *actuation phase* through the same WSAN.

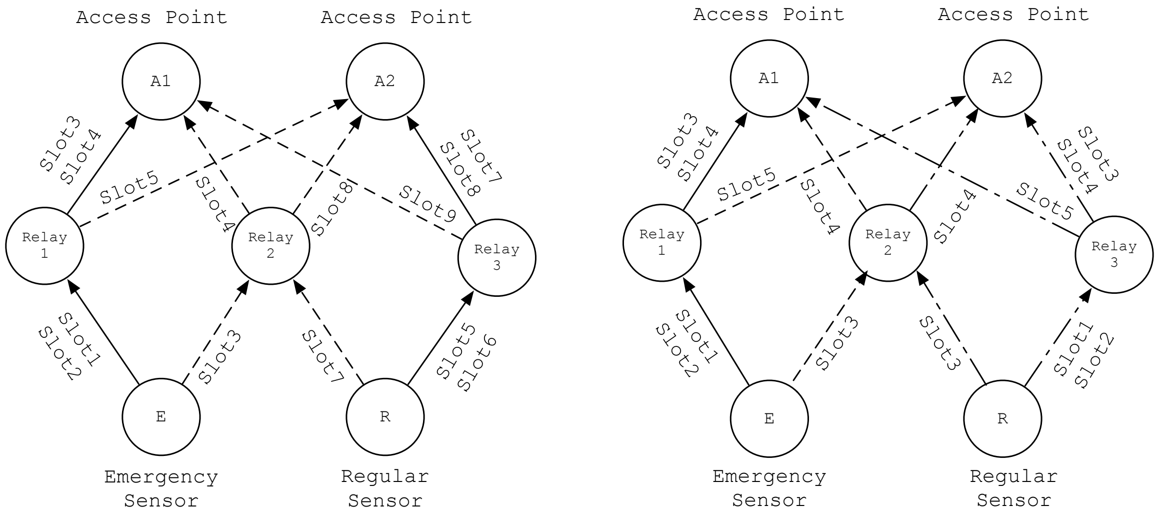
There are two types of flows in our system: periodic regular flows and aperiodic emergency alarms. A regular flow generates packet periodically in both *sensing phase* and *actuation phase*. Emergency alarms are triggered sporadically. Packets of a regular flow or an emergency alarm must be delivered within its deadline. An emergency alarm is more critical than a regular flow.

Based on the state-of-art WirelessHART standard [3], the WSAN adopts a centralized architecture in our design. The Network Manager and Access Points are usually connected by reliable wired links while the rest of the WSAN communicate using the wireless mesh network. The transmission schedule is organized in terms of time slots (10 ms per slot). The network protocol stack comprises (1) a routing layer that supports both source routing and reliable graph routing. (2) a MAC layer running a multi-channel Time Division Multiple Access (TDMA) protocol and (3) the IEEE 802.15.4 physical layer for low-power radios.

4.4 Wireless Design

In this section, we firstly introduce our Wireless Cyber-Physical Simulator [1] [47]; we then describe our WirelessHART stack implementation; we finally we present our real-time emergency communication protocols, and other major changes in WCPS 2.0.

Given a WirelessHART network, we consider the real-time communication problem of $k + l$ flows $F = \{E_1, .., E_k, R_1, ..., R_l\}$. Each regular flow $R_i \in R$ is periodically generated with a period P_i and a deadline D_i, where $D_i \leq P_i$. As specified in Graph Routing [33], from the source to the destination, there exists at least *two* outgoing links(one primary, one backup) for every non-destination node. An emergency flow $E_j \in E$ is triggered aperiodically with a



(a) Periodic Scheduling (PS)

(b) Slot Stealing (SS)

Figure 4.2: Wireless Communication Protocols

deadline D_j. The communication latency L_n of a packet for a flow generated at slot n and delivered at slot m is defined as $m - n + 1$.

We observe the real-time communication problem for wireless control involve mixed criticalities, where the emergency flows have higher criticality than regular flows. The objectives of real-time communication are two fold: (1) In the regular mode, i.e., when there is no emergency, all regular flows should meet their deadlines; (2) In the emergency mode when emergency occurs, emergency flows should meet their deadlines, while no guarantee is provided to regular flows.

The challenge in supporting emergency communication lies in the fact that emergencies occur only occasionally and the system operates in the regular mode most of the time. However, when emergency does occur, it is critical to meet the deadlines of emergency flows.

A simple approach to schedule an emergency flow is to reserve time slots for a virtual periodic flow (also called a *periodic server*) that is scheduled alongside the regular flows. Emergency alarms are transmitted within the time slots designated to the periodic server. A drawback of this periodic scheduling (PS) approach is that it wastes network bandwidth when there is no emergency.

To avoid wasting resources during the regular mode, we introduce a slot stealing (SS) mechanism that allows regular flows to *steal* slots from emergency schedule when emergency does not exist, and thus would enhance slot utilization during regular operations. Furthermore, we propose event-based emergency communication to further improve network efficiency during the emergency mode.

4.4.1 Periodic Scheduling (PS)

PS creates a virtual periodic flow for each emergency alarm and schedule them alongside the regular flows. Emergency alarms are transmitted in the slots allocated for the corresponding virtual periodic flow.

We adopt a fixed priority scheduling policy and a two-level priority assignment approach. Virtual periodic flows always have higher priorities than regular flows. Among the virtual periodic flows, we assign their priorities based on the rate monotonic policy. Similarly, regular flows are also prioritized based on the rate monotonic policy.

For example, Fig. 4.2(a) illustrates a transmission schedule of PS. For simplicity purposes, this example uses a single channel, but we consider multi-channel communication in our case studies. Links in Fig. 4.2(a) are categorized as primary paths (solid lines) and backup paths (dashed lines). Communication on primary paths happens in *dedicated* time slots while communication on backup paths are scheduled in *shared* time slots, when different senders may contend for transmission opportunities. Emergency sensor E is scheduled to transmit to Relay 1 in Slot 1 and Slot 2 . If either of the transmissions in Slot 1 or 2 succeeds, following transmissions from Relay 1 to A1 will be scheduled in Slot 3 and Slot 4. However, if both transmissions in Slot 1 and Slot 2 fail, a backup link will be used by E to transmit to Relay 2 and then from Relay 2 to A1, in shared Slot 3 and Slot 4, respectively. Data from the regular sensor R will take similar scheduling and routing strategy. PS takes 9 slots in total to schedule both flows. Algorithm 1 shows a detailed algorithm of PS.

Algorithm 1: Periodic Scheduling(PS)

```
input  : $E, R, routes, connectivity$
output: $S[1 \cdots T][0 \cdots m-1]$
$F \leftarrow \{E, R\}$; $ch \leftarrow 0$; $m \leftarrow$ total channel; $T \leftarrow$hyper period;
while ($F \neq \emptyset$) do
    $flo \leftarrow$ Highest priority flow in $F$; $rout \leftarrow \{route$ of $flo\} \subset routes$;
    while ($rout \neq \emptyset$) do
        $send \leftarrow$ first transmission on $rout$.
        if ($s \leq T$) then
            if ($\{conflicts\ in\ connectivity\} = \emptyset$) then $S[s][ch] \leftarrow send$; $ch \leftarrow ch + 1$;
        else
            return $unschedulable$.
        $rout \leftarrow rout - \{send\}$; $s \leftarrow s + 1$;
    $F \leftarrow F - \{flo\}$;
return $S[1 \cdots T][0 \cdots m-1]$;
```

4.4.2 Periodic Scheduling with Slot Stealing (SS)

In PS, time slots allocated to emergency flows are left unused when there is no emergency, which is a waste of precious network resource. To overcome this limit, SS allows emergency alarms and regular flows to be scheduled in the *same* dedicated slots. When emergency does not exist, emergency slots will be used(stealed) by regular flows instead. Whenever an emergency exists, the emergency transmission would take the slot while the regular transmission would back off.

Slot Stealing is technically inspired by hybrid MAC protocols such as Z-MAC [62]. An emergency packet is transmitted immediately at the beginning of a slot shared with the regular packet. In contrast, a regular packet first performs a Clear Channel Assessment(CCA) after waiting for a constant backoff time. If there is any other transmission going on(likely from an emergency sender), the regular sender would cease its transmission. Otherwise, it goes ahead and transmit the packet.

Fig. 4.2(b) shows an example of SS. Following the same retransmission and Graph routes as PS, we see SS takes 5 time slots (4 slots fewer) to accommodate both flows. Algorithm 2 depicts the detailed algorithm of SS.

4.4.3 Event-based Slot Stealing (SS-Event)

There are two alternative approaches to send emergency alarms during an emergency. For systems that need to periodically monitor and control the emergency state, an emergency control flow is activated whenever emergencies exist. The emergency flow then periodically generates sensor data and control command until the emergency is over.

For systems that do not need to periodically monitor and control the emergency state, the system can adopt an event-based approach to communicate the emergency alarms, i.e., an emergency sensor only sends an alarm-start and an alarm-end packets in the beginning and

Algorithm 2: Scheduling with Slot Stealing

```
input  : $E, R, routes, connectivity$
output: $S[1 \cdots T][0 \cdots m-1]$
$ch \leftarrow 0$; $m \leftarrow$ total channel; $T \leftarrow$hyper period;
Schedule $E$ with the $PS$ algorithm.
while ($R \neq \emptyset$) do
    $flo \leftarrow$ Highest priority flow in $R$; $rout \leftarrow \{route$ of $flo\} \subset routes$;
    while ($route \neq \emptyset$) do
        $send \leftarrow$ first transmission on $rout$.
        if ($s \leq T$) then
            if ($\{conflicts\ in\ connectivity\} = \emptyset$) then
                if ($\{free\ channel\} \neq \emptyset$) then $S[s][ch] \leftarrow send$; $ch \leftarrow ch + 1$;
                else $ch \leftarrow$ sharable ch of emergencies; $S[s][ch] \leftarrow send$;// Steal
            else
                if ($\{shareable\ channel\} \neq \emptyset$) then $ch \leftarrow$ sharable ch of emergencies;
                $S[s][ch] \leftarrow send$;// Steal
                else return $unschedulable$;
        else
            return $unschedulable$.
        $rout \leftarrow rout - \{send\}$; $s \leftarrow s + 1$;
    $R \leftarrow$ R$-\{flo\}$;
return $S[1 \cdots T][0 \cdots m-1]$;
```

end of the emergency. While this event-based communication results in the same transmission schedule as in Algorithm 2, event-based SS communication can significantly reduce the number of regular transmissions that are affected by emergency transmissions, potentially leading to better control performance. Hence forth, we denote the combination of event-based communication and SS as SS-Event.

We note SS and SS-Event have clear tradeoffs between data loads and communication reliability. Periodic flows in SS on one hand would reduce chances of missing emergency alarms while on the other it would override regular flows with excessive periodic traffics(in stealed slot). In contrast, SS-Event has less impact on regular flows but runs at the danger of completely missing critical alarm packets.

4.5 Control Design

We apply our emergency handling protocol in a coupled water tank system as a case study. In this section we describe the dynamical model of the water tank system and our controller design.

4.5.1 Coupled Water Tank

A diagram of the coupled water tank is shown in Figure 4.3. This system shares similar dynamics with many other process control systems, e.g., irrigation networks [13]. Our choice to use this system as a case study is based on its simple yet representative dynamics, its hybrid dynamical nature (as the evolution of the system changes when the water tanks are

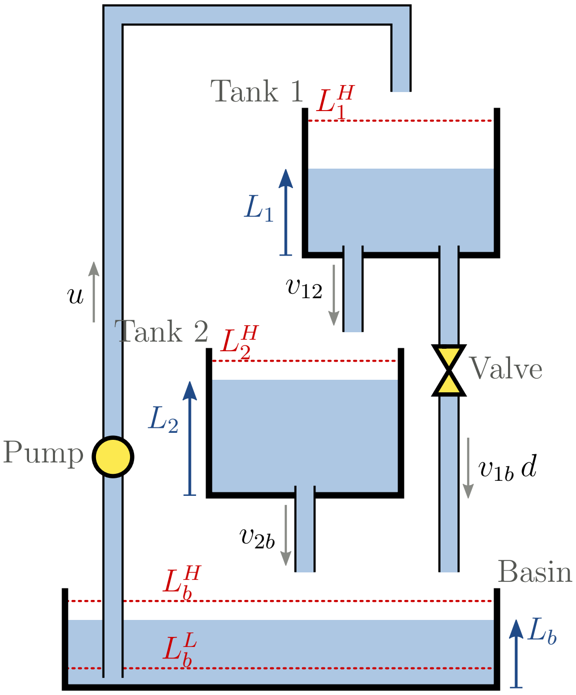


Figure 4.3: Diagram of the coupled water tank system. The water levels of Tank 1, Tank 2, and Basin are denoted L_1, L_2, and L_b, respectively. The emergency water levels are denoted L_1^H, L_2^H, L_b^L, and L_b^H. The natural pipe flows are denoted u, v_{12}, v_{1b}, v_{2b}. The state of the valve is denoted $d \in \{0,1\}$, where $d = 1$ if the valve is ON.

either full or empty), and more importantly, its similarity to systems commonly used in industrial applications.

The coupled water tank system is comprised of one pump, one ON/OFF valve, two water tanks, and one basin. The pump is responsible for pushing water from the basin to Tank 1. The flow through the pump is a controlled variable, denoted u. Tank 1 is placed higher than Tank 2, and water flows due to gravity via a pipe at the bottom of Tank 1 placed above Tank 2.

The flow through this pipe is denoted v_{12}, and satisfies the following equation:

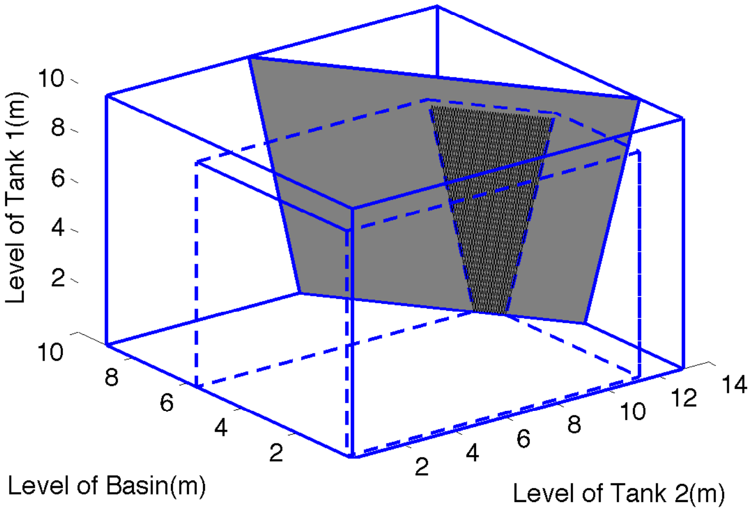


Figure 4.4: Diagram of the state space of the coupled water tank system. The gray plane corresponds to the feasible states since no water enters or exits the system. The dark region of the plane corresponds to the region where no emergencies occur. The dotted lines show emergency limits, and the solid lines show the physical limits of the tanks.

$$v_{12} = \frac{1}{\rho\, R_{12}} \sqrt{\rho\, g\, L_1}, \tag{4.1}$$

where ρ is the density of water, g is the gravity constant, R_{12} is the resistance parameter of the pipe, and L_1 is the level in Tank 1.

Similarly, Tank 2 is placed higher than the basin, and water flows via a pipe at the bottom of Tank 2 placed above the basin. The flow through this pipe is denoted v_{2b}, and satisfies the following equation:

$$v_{2b} = \frac{1}{\rho\, R_{2b}} \sqrt{\rho\, g\, L_2}, \tag{4.2}$$

where, besides the parameters defined in Equation (4.1), R_{2b} is the resistance parameter of the pipe, and L_2 is the level in Tank 2.

A pipe at the bottom of Tank 1, above the basin, is interrupted by the *ON*/*OFF* valve, hence water flows only when the valve is *ON*. The flow through this pipe, denoted v_{1b} is zero when the valve is *OFF*, and satisfies the following equation when the valve is *ON*:

$$v_{1b} = \frac{1}{\rho\, R_{1b}} \sqrt{\rho\, g\, L_1}, \tag{4.3}$$

where R_{1b} is the resistance parameter of the pipe.

Using conservation of mass and equations (4.1) to (4.3) we get the following dynamic equations for the coupled water tank system:

$$\frac{\mathrm{d}L_1}{\mathrm{d}t} = \begin{cases} \frac{1}{\rho\, A_1}\big(-v_{12} - v_{1b}\, d + u\big) & \text{if } L_1 \in [0, L_1^{\max}], \\ 0 & \text{otherwise}, \end{cases} \tag{4.4}$$

$$\frac{\mathrm{d}L_2}{\mathrm{d}t} = \begin{cases} \frac{1}{\rho\, A_2}\big(v_{12} - v_{2b}\big) & \text{if } L_2 \in [0, L_2^{\max}], \\ 0 & \text{otherwise}, \end{cases} \tag{4.5}$$

$$\frac{\mathrm{d}L_b}{\mathrm{d}t} = \begin{cases} \frac{1}{\rho\, A_b}\big(v_{1b}\, d + v_{2b} - u\big) & \text{if } L_b \in [0, L_b^{\max}], \\ 0 & \text{otherwise}, \end{cases} \tag{4.6}$$

where $L_1^{\max}$, $L_2^{\max}$, and $L_b^{\max}$ are the physical heights of the tanks, A_1, A_2, and A_b are the areas of the tanks, and $d \in \{0, 1\}$ is a controlled variable such that $d = 1$ when the valve is *ON*, and $d = 0$ when the valve is *OFF*.

4.5.2 System Emergencies

In our case study, the objective is to achieve set-point tracking of the water level in the Tank 2 by adjusting the flow u. We assume that Tank 2 has a level sensor (measuring L_2), and that the pump allows us to fully control the flow $u \in [0, u^M]$.

We also define four emergency situations, three of them corresponding to each water tank having too much water, which may produce spillage, and one corresponding to the basin having too little water, which may lead to the pump sucking air instead of water. Using the notation in Figure 4.3, the emergencies occur in the following situations: $L_1 > L_1^H$, $L_2 > L_2^H$, $L_b > L_b^H$, and $L_b < L_b^L$.

Note that the coupled water tank system is closed, i.e., it only recirculates water, and no water enter or leaves the system. This condition can be observed from equations (4.4) to (4.6), since $A_1 \frac{\mathrm{d}L_1}{\mathrm{d}t} + A_2 \frac{\mathrm{d}L_2}{\mathrm{d}t} + A_b \frac{\mathrm{d}L_b}{\mathrm{d}t} = 0$ (when the water levels are within the normal limits). In other words, $A_1 \, L_1(t) + A_2 \, L_2(t) + A_b \, L_b(t) = A_1 \, L_1(0) + A_2 \, L_2(0) + A_b \, L_b(0)$ for each $t \geq 0$. Using this extra constraint, even though the system has three states, we can plot its trajectories in a two-dimensional plane, as shown in Figure 4.4. The dark region in that figure corresponds to the subset of the two-dimensional plane where no emergencies occur. The rest of the two-dimensional plane corresponds to the state space where at least one emergency is active. The two-dimensional plane is bounded by the physical constraints of the system, i.e., the fact that all levels must remain above zero and below the maximum height.

4.5.3 Actuator and Controller Design

For water level control in Tank 2 we use a PID controller sensing L_2 and acting on u. Also, to efficiently correct emergencies, the valve, which is normally OFF, is sometimes switched to ON. Thus, the control strategy for this system is hybrid, since whenever an emergency is activated the controller behavior is changed. PID parameters are decided empirically in this study, we refer interested users to our code for more design details.

The PID controller, used when no emergencies are active, follows the following equations:

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(s)\, ds + T_d \frac{\mathrm{d}e}{\mathrm{d}t}(t) \right), \tag{4.7}$$

where $e(t) = L_2^{\mathrm{sp}} - L_2(t)$, L_2^{sp} is the desired set-point, and K, T_i, and T_d are the controller parameters. Also, whenever the right-hand side of equation (4.7) is above u^M then we set $u(t) = u^M$, and when it is below zero then we set $u(t) = 0$. The interested readers can go to [14] for more details regarding PID controllers.

We apply the following rules when emergencies are activated, in priority order:

1. If $L_1 > L_1^H$, $L_2 > L_2^H$, or $L_b < L_b^L$, we set $u = 0$ and $d = 1$ (i.e., shut off the pump and open the valve).
2. If $L_b > L_b^H$, and either $L_1 > L_1^H$ or $L_2 > L_2^H$, then we set $u = 0$ and $d = 1$.
3. If $L_b > L_b^H$, we set $u = u^M$ and $d = 0$ (i.e., pump as much water as possible from the basin and close the valve).

The rules above are heuristics designed under the assumption that maintaining the level in Tanks 1 and 2 is more important than maintaining the level in the basin. Hence, if either

Tank 1 or Tank 2 have too much water, or if the basin has too little water, we transport water from the Tanks to the basin as quickly as possible (Rule 1). Since removing water from either of the Tanks conflicts with removing water from the basin, the Tanks take precedence (Rule 2). Finally, if the water level in the basin is too high then we pump water from the basin as quickly as possible (Rule 3).

To avoid high frequency switching between different emergency modes (or even Zeno executions [37]), the controller is forced to stay at least $\kappa > 0$ seconds in each mode after it is activated.

4.6 Case Study

This section presents performance evaluations with a case study. The objective here is not simply to compare networking protocols. Rather, we would like to systematically evaluate our holistic framework, including the WCPS 2.0 simulator, the WirelessHART stack, as well as the real-time communication protocols.

To stress the WSAN, we have implemented two sets of Coupled Water Tank systems sharing the same WSAN. Each coupled water tank set is comprised of two water tanks and one basin. In total we have 4 tanks and 2 basins in the plant. We have attached 8 emergencies sensors and 2 regular sensors for monitoring purposes. In our system, plant data is first generated by water tanks and then fed into the WSAN in TOSSIM. Having been transmitted through the WSAN, sensor data with delay and loss information will be updated to the controller. Control commands from the controller would later be transmitted through the downlink WSAN and eventually applied for closed-loop control. We note since WirelessHART adopts deterministic

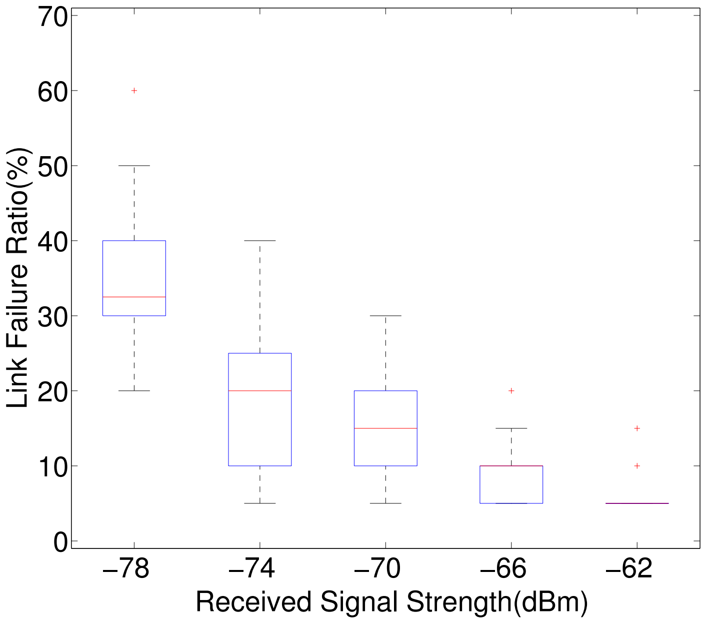
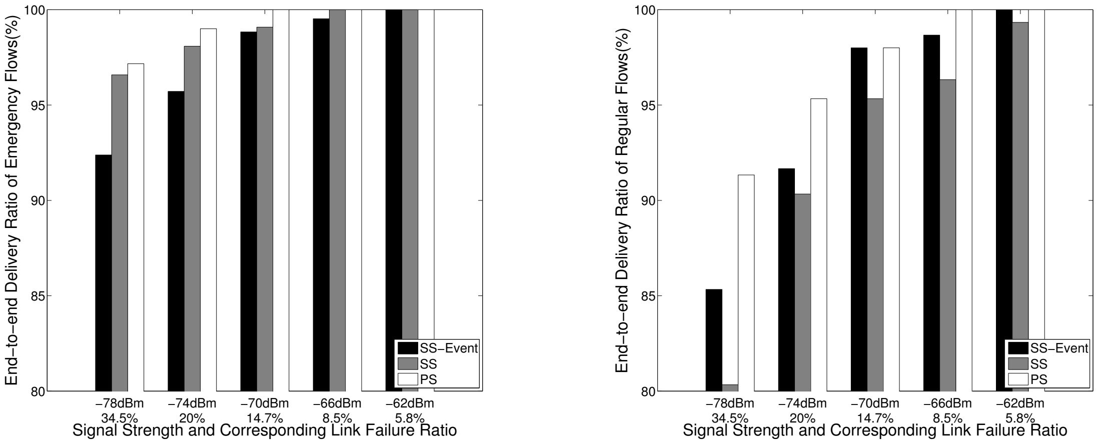


Figure 4.5: Link Failure Ratio

TDMA schedules, in this evaluation we focus more on impact of network reliabilities instead of network delays; for sensor data that have not arrived before the deadline (e.g., dropped or missed the deadline), control decisions are made upon most recent available packet from the same sensor.

4.6.1 Network Performance

In our study, wireless traces from a 21-node subset of the Wustl Testbed [4] have been used to form a WSAN in TOSSIM, with a maximum path length of 6 hops . We directly use the connectivity and the multi-channel wireless noises from the Wustl Testbed; on the other hand, we use controlled Received Signal Strength with uniform gaps to simulate various wireless signal strength. Fig. 4.5 shows link quality statistics (3000 packets), where -74 dBm



(a) Delivery Ratio for Emergencies

(b) Delivery Ratio for Regular Flows

Figure 4.6: Average End-to-end Delivery Ratio

and -62 dBm features 20% and 5.8% averaged link failure under the Wustl Testbed noise, respectively.

Fig. 4.6 shows end-to-end delivery ratios of the three communication protocols implemented in WCPS 2.0. Since all experiments are done with the Wustl Testbed noise, we show both Received Signal Strength and corresponding Link Failure Ratio on the x-axis. In other words, multi-hop end-to-end delivery ratios in Fig. 4.6 can be reproduced in WCPS 2.0 as long as Link Failure Ratio is the same, whereas signal strength and noise traces needn't. As in Fig. 4.6(a) and Fig. 4.6(b), both types of flows have better end-to-end delivery ratios as link failures improve(Received Signal Strength increases).

It is interesting to see PS, SS and SS-Event all achieve over 95% emergency delivery and 90% regular flow delivery ratios at 20% link failure ratio (-74dBm), implying that Graph Routing is working properly. Note for all 3 protocols, emergencies in Fig. 4.6(a) outperforms regular flows in Fig. 4.6(b), which is because emergency flows always have higher transmission priorities over regular ones. If we compare SS against SS-Event in Fig. 4.6(a) and Fig. 4.6(b),

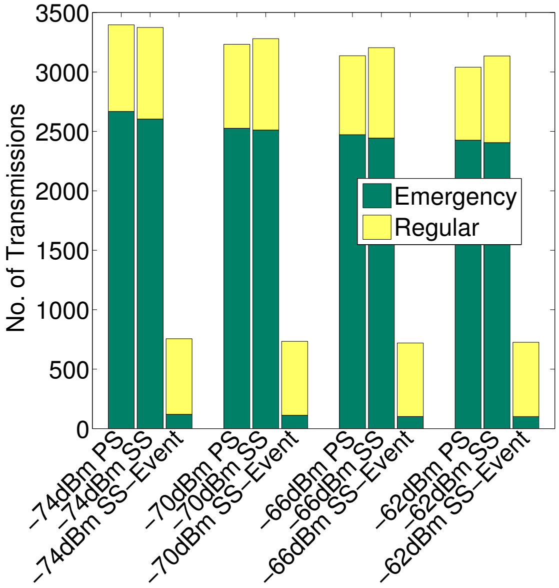


Figure 4.7: Number of Transmissions under Various Wireless Conditions

respectively, we can see SS has better emergency alarm delivery ratios (because of redundant periodic flows) and worse regular flow deliveries (because of conflicts from emergencies).

Fig. 4.7 further shows number of transmissions (50 second simulation) under various wireless conditions. Dark part of a bar means transmission counts for emergencies while light parts represent traffics for regular flows. As expected, PS and SS have more emergency transmissions due to the periodic nature. For regular traffics, it is interesting to see that SS has more regular traffics than the other two, which is because of retransmissions caused by backoffs during Slot Stealing. This also validates correctness of our WirelessHART stack and the Slot Stealing mechanism.

4.6.2 Control Performance

In our control evaluations, we choose to study cases with challenging initial conditions(e.g., small trapezoid area in Fig. 4.4). System failure is defined in twofold: first, the system can not stabilize inside feasible region given a time bound, e.g., 100 seconds; second, the system violates physical constraints, e.g., water spilling.

We further adopt three general evaluation metrics: *system failure ratio*, *time percentage outside the feasible region*, and *maximum distance to the feasible region. Time percentage outside the feasible region* is defined as the time percentage when the system state is out of the feasible region but within the control time limit. *Maximum distance to the feasible region* is defined as the maximum distance of system state to the feasible region(see Fig. 4.4). Simulations were executed under the following conditions:

- No emergency exists at the beginning;
- Total amount of water (i.e. $A_1\,L_1(0) + A_2\,L_2(0) + A_b\,L_b(0)$) is equivalent to maximum allowed capacity of Tank 2 and Basin ($A_2\,L_2^H + A_b\,L_b^H$);
- Emergency control runs at maximally supported frequency by the WSAN.
- Each simulation lasts 200 seconds, decided by the system time constant.

In our case study, we set control period consistently as 1Hz for regular loops while setting emergency control on the *maximally supported frequency*, bounded by the operation period of the WSAN. For example, TDMA superframe for SS-Event has 48 time slots and hence the maximally supported operation frequency for SS-Event is 2Hz(i.e., 500ms period). Similarly,

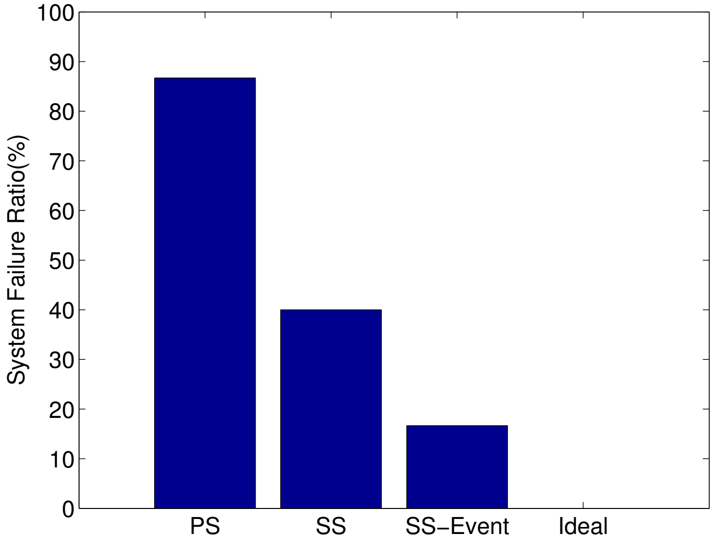


Figure 4.8: System Failure Ratio under 20% Link Failures (-74dBm Wireless Signal Strength)

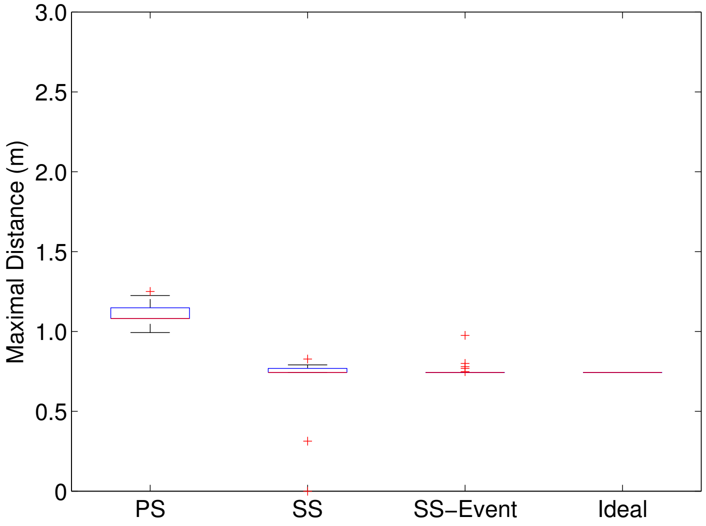


Figure 4.9: Tank 2 Maximal Distance to Feasible Region under 20% Link Failures (-74dBm Wireless Signal Strength)

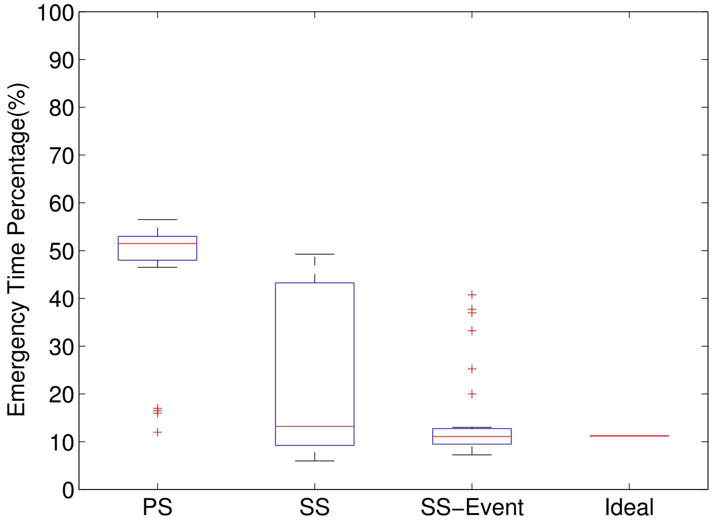


Figure 4.10: Emergency Time Percentage under 20% Link Failures (-74dBm Wireless Signal Strength)

PS who has a longer superframe can only operate at a slower control frequency, i.e., 1Hz in our case study.

All statistics in Fig. 4.8 and Fig. 4.10 are done under 20% link failures (-74dBm wireless signal strength), with results averaged from 15 simulations (3000 control steps). System failure ratios in Fig. 4.8 shows SS-Event has achieved the closest performance compared to *Ideal(Wired) Control*, which is because SS-Event has higher emergency control frequency and hence shorter delays. A close look shows that SS is less successful than SS-Event is because SS has consistently dropped too many regular packets due to conflicts.

Maximum distance to the feasible region of Tank 2 in Fig. 4.9 and time percentage outside the feasible region in Fig. 4.10 shows the same trend that SS-Event is the better than the other two. Fig. 4.11 further depicts control performance of SS-Event under various wireless conditions. For cases featuring 5.8% link failures, SS-Event have 0% system failure, i.e.,

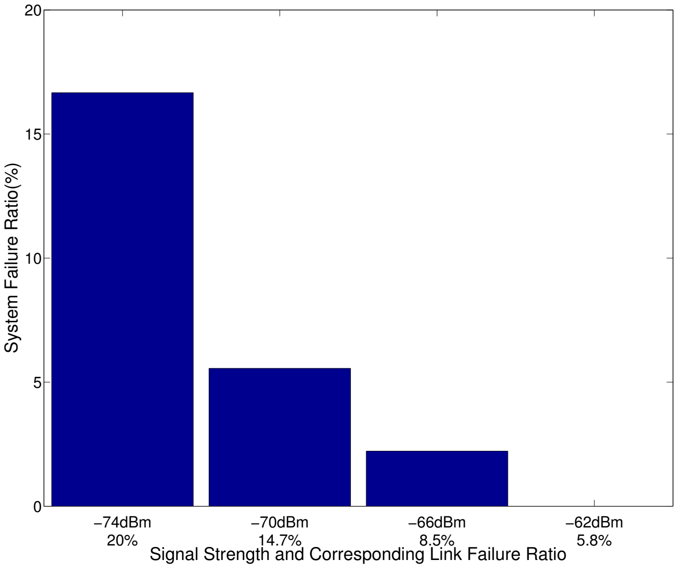


Figure 4.11: System Failure under Various Wireless Conditions

100% control success. This is indeed encouraging as it demonstrates even for a 6-hop lossy wireless network, successful system control can still be achieved with careful wireless designs.

In sum, sitting on top of the state-of-art WirelessHART protocol stack in WCPS 2.0, we for the first time have been able to do scalable case studies for wireless emergency handling, and encouraging experiment results have been achieved. The periodic and event-based communication framework can be easily tailored according to application needs or optimization formulations, which is beyond the scope of this work and left for future studies.

4.7 Summary

Recent years have witnessed significant interests in adopting wireless sensor-actuator networks in process control. However, the problem of incorporating emergency alarms in wireless process control remains to be explored. This chapter presents the first systematic approach to integrate emergency alarms into wireless process control systems. The challenge in emergency communication lies in the fact that emergencies occur occasionally, but must be delivered within their deadlines when they occur. The contributions of this work are three-fold: (1) we propose efficient real-time emergency communication protocols based on slot stealing and event-based communication; (2) we build an open-source WirelessHART protocol stack in the Wireless Cyber-Physical Simulator (WCPS) for holistic simulations of wireless control systems; (3) we conduct systematic studies on a coupled water tank system controlled over a 6-hop 21-node WSAN. Our results demonstrate our real-time emergency communication approach enables timely emergency handling, while allowing regular feedback control loops to effectively share resources in WSANs during normal operations. Our work demonstrates the feasibility and efficacy of incorporating emergency alarms into wireless process control systems. Moreover, WCPS 2.0 with the WirelessHART protocol stack and case studies provide an enabling framework for exploring wireless process control design and hence represents a promising step toward practical wireless process control systems.

Chapter 5

Wireless Routing and Control: A Cyber-Physical Case Study

5.1 Introduction

Wireless sensor-actuator network (WSAN) technology is gaining rapid adoption in process industries to lower deployment and maintenance costs in challenging industrial environments [54]. Industrial standard organizations such as ISA [5], HART [9], WINA [6] and ZigBee [8] have been actively pushing WSAN for industrial automation [64]. 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 [69], while previous control inputs can be buffered at actuators for use in case of packet loss from the controller [38]. Despite considerable success in theoretical advancements in wireless control design, however, there has been limited empirical studies on wireless control systems combining state-of-the-art control design and standard-based industrial WSANs under realistic wireless conditions.

This chapter 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, 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) [46, 47] that integrates Simulink and a WirelessHART protocol stack based on a realistic wireless model and traces collected from a real-world wireless testbed. Our experiments demonstrate 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 [45]. 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 where 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 improve 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-art control techniques such as observers, MPC, 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 chapter is organized as follows. Section 5.2 discusses related works. Section 5.3 introduces the wireless control design. Sections 5.4 and 5.5 present the design and results of the case study. Section 5.6 concludes the chapter.

5.2 Related Work

There have been extensive studies on WSAN routing. A thorough review on routing schemes in wireless sensor networks was presented in [11]. Detailed routing requirements in low-power and lossy networks for industrial applications were introduced in [60]. A graph routing algorithm for WirelessHART was presented in [33]. A conflict-ware real-time routing algorithm

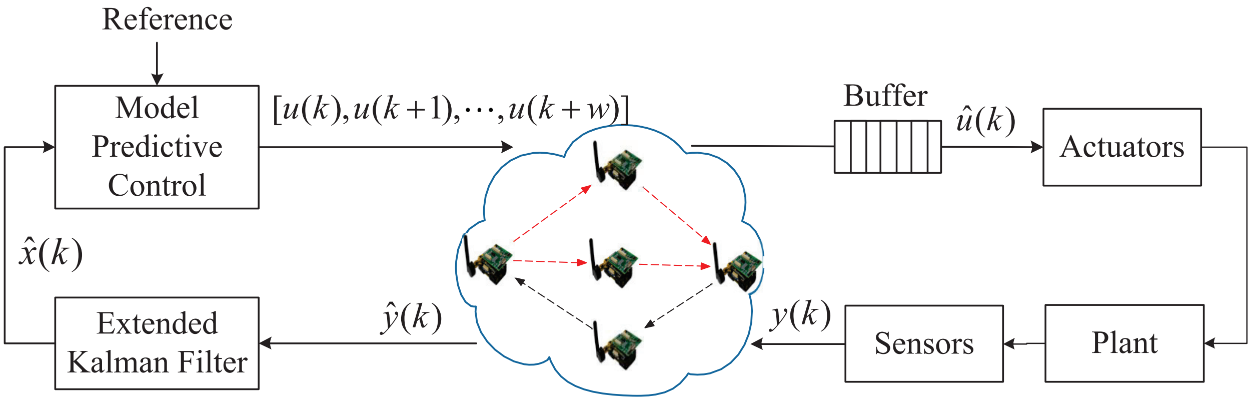


Figure 5.1: System Architecture: Sensor measurements from sensors are transmitted wirelessly to the observer, which generates estimated system states. The controller takes estimated system states and computes actuation commands, which again are transmitted to actuators wirelessly. We introduce a buffer for actuators such that we can reuse previously buffered actuation inputs when packet drops happen.

for industrial WSAN was presented in [80]. An energy-efficient routing approach was introduced in [79]. Energy-aware routing for real-time and reliable wireless industrial sensor networks was introduced in [34]. These works focus solely on the network without considering the control aspect of a wireless control system.

Networked control has received considerable attentions. Passivity-based control architecture was proposed for cyber-physical systems in [41]. Self-triggered control has been developed for wireless networks [13, 75]. A distributed control approach has been proposed for WSANs [58]. State observers such as Kalman filters have been proven to be resilient against uncertainties [52, 67]. An effective Kalman filter was introduced based on intermittent observations [68]. While the results are encouraging, these works do not consider the routing of a wireless mesh network based on industrial standards.

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 [27]. Sampling rate selection for wireless control systems is studied in [65]. Etherware in [16,38] 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.

5.3 Wireless Control Design

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

5.3.1 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. 5.1.

On the control side, we use state-of-art control schemes. We first introduce an observer in sensing to mitigate uncertainties caused by the wireless network. We next adopt an Model Predictive Control (MPC) as the controller. On actuators, we introduce a buffer which stores a sequence of control commands computed by the MPC.

5.3.2 Control Design

Sensing with Extended Kalman Filter

State observers such as Kalman filters have been shown to be resilient against uncertainties [52, 67, 68]. 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 to the previous step. Then, the *update* step compares the current outputs estimation with newly arrived sensing data y_k, 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 [68]. In their chapter, they propose a modified *update* step which disregards the sensing data when a packet is dropped. For this reason we use a modified EKF with their idea in our WSAN closed-loop system.

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) [30]. 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 are computed by the MPC, corresponding to the whole time horizon of length T, only the first sample is commonly used for actuation while the rest are discarded. To improve the resilience, instead of transmitting the first sample, we transmit a certain sequence of values in the prediction horizon, which 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 every time sample with a new sequence of control inputs. But if the wireless network drops a packet, then the buffer simply applies the next available control input, which was received in the last packet that arrived successfully. We refer to this approach as a *buffered control*, which even though it is not a new term [17,18], to our knowledge this is the first chapter studying it in conjunction with a wireless mesh network based on the WirelessHART standard. In particular, we present what to our knowledge the first systematic study of the interaction between wireless routing protocols and control.

In Fig. 5.1, the state observer takes measurements $y(k)$ from the plant to produce estimated system states $\hat{x}(k)$; a MPC takes these estimates $\hat{x}(k)$, compares against the reference signal and generate a sequence of predicted control commands $\big[u(k), u(k+1), \ldots, u(k+w)\big]$ with a length of w, which we explain later with more details. The sequence is later stored in a buffer on the actuator and $u(k)$, the control command corresponding to the current time, is applied to the actuator.

5.3.3 Network Design

WirelessHART Architecture We adopt a WirelessHART [9] architecture for our WSAN design. WirelessHART uses multiple channels defined in IEEE 802.15.4 physical layer specification, and adopts *channel hopping* for 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 the transmission schedules, 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 senders compete for one transmission opportunity.

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 backup path from each intermediate node to the destination. In *source* routing, only one sender/receiver pair will be considered for the first transmission and one more 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 3 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 costs. 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 period, but worse network reliabilities. We show results with more details in Section 5.5.

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. For example, in our specific case study, the state observer is highly effective in mitigating the effects of packet loss from sensors to the controller. In comparison, the system performance is more vulnerable to packet loss from the controller to actuators despite the help of the 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 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 5.5), 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 [79]. Those original algorithms were designed to improve the network lifetime of a WirelessHART network, e.g., the operation time till the first field device runs out of its battery. 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 [79] the source routing and graph routing algorithms were designed as separate algorithms each of which is 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.

5.4 Case Study Design

In this section, we first introduce an exothermic chemical reaction system as our control plant. We then introduce 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 [47], 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 [47], wireless process control systems [46], and was recently used to build a benchmark problem for wireless structural control released to the civil engineering community [73]

5.4.1 Physical 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 5.2. 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 [13] or oil refineries [83]. Our choice of this plant is also motivated on 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 $T_2(t)$. 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.

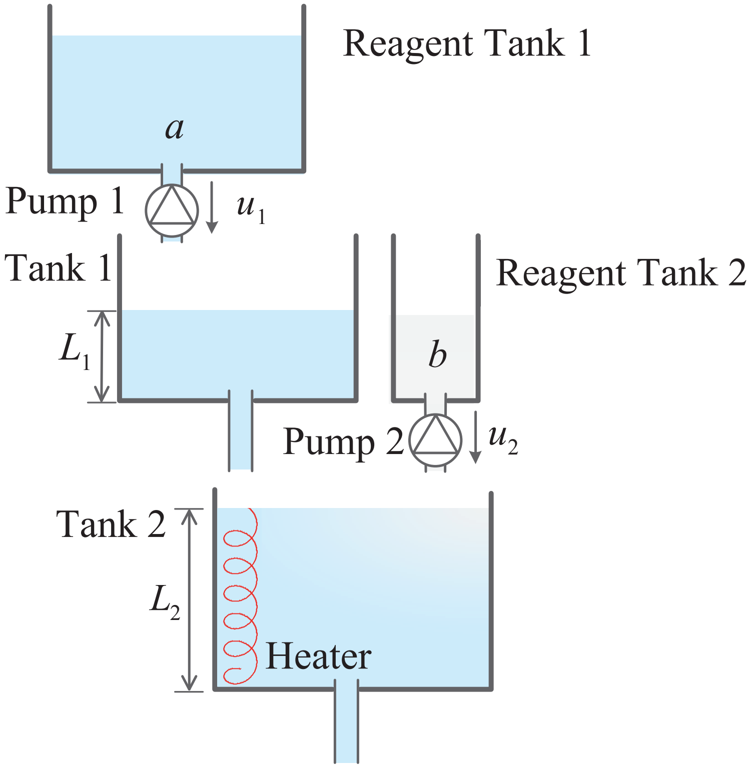


Figure 5.2: Exothermic Chemical Reaction Plant

Table 5.1 shows the list of parameters modeling the dynamic behavior of our plant. To simplify our notation we define:

$$g(L) = \alpha \arctan(\beta L). \tag{5.1}$$

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:

$$A_1\,\dot{L}_1 = -k_1\,g(L_1) + u_1. \tag{5.2}$$

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

$$A_2\,\dot{L}_2 = k_1\,g(L_1) - k_2\,g(L_2) + u_2. \tag{5.3}$$

The dynamical model for the temperature in Tank 2 is:

$$A_2\,L_2\,\dot{T}_2 \;=\; H_c\,\big(k_1\,t_1\,g(L_1) \;-\; A_2\,T_2\,g(L_2) + t_2\,u_2\big) + h_2\,L_2 + H_r\,r_c\,k_r\,A\,B\,\exp\left(-\frac{E_a}{R\,T_2}\right). \tag{5.4}$$

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

$$\begin{bmatrix}\dot{A}\\ \dot{B}\end{bmatrix} \;=\; \Big[a_0\,k_1\,g(L_1) - k_2\,A\,g(L_2) - k_2\,B\,g(L_2) + b_0\,u_2\Big] \;-\; \begin{bmatrix}r_a\\ r_b\end{bmatrix} k_r\,A\,B\,\exp\left(-\frac{E_a}{RT_2}\right). \tag{5.5}$$

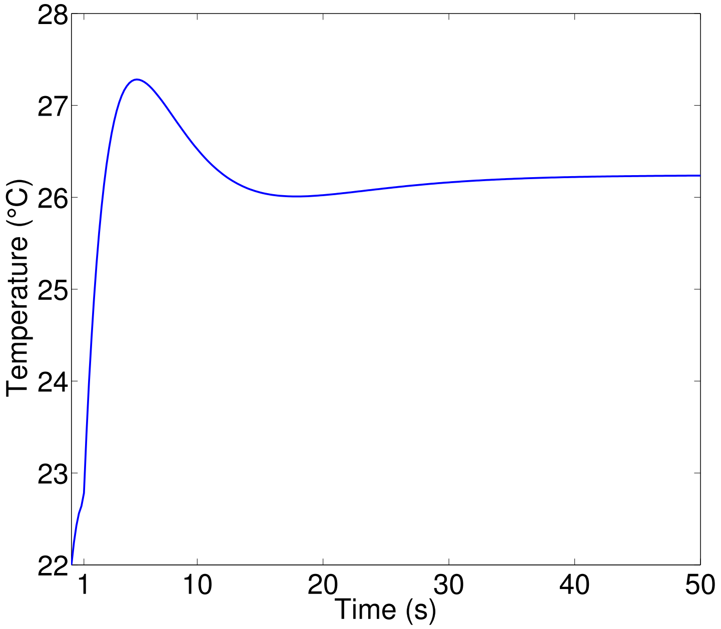


Figure 5.3: Open-loop Step Response

Fig. 5.3 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 4 seconds.

5.4.2 Observer and Controller Implementation

As explained above, we use an Extended Kalman Filter for intermittent observations, as described in [68], as our observer, and a Model Predictive Control scheme with signal buffers as our controller. Using the same notation as in Figure 5.1, the output of the system is $y(k) = \left[L_2(k), T_2(k)\right]$ 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

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 5.1: Plant Parameters

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 $u(k)$ when 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 5.4.3). 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.

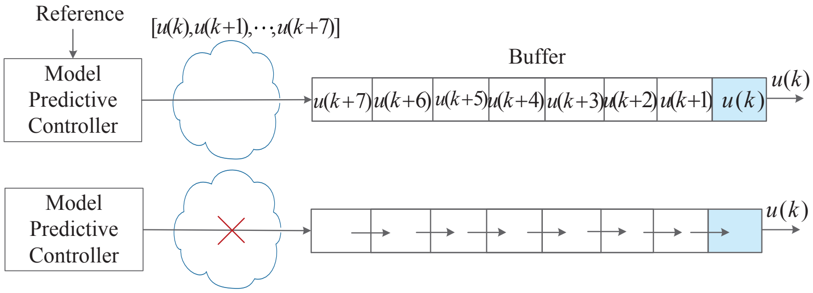


Figure 5.4: Buffered Actuation Mechanism

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 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 the solver *quadprog*.

5.4.3 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

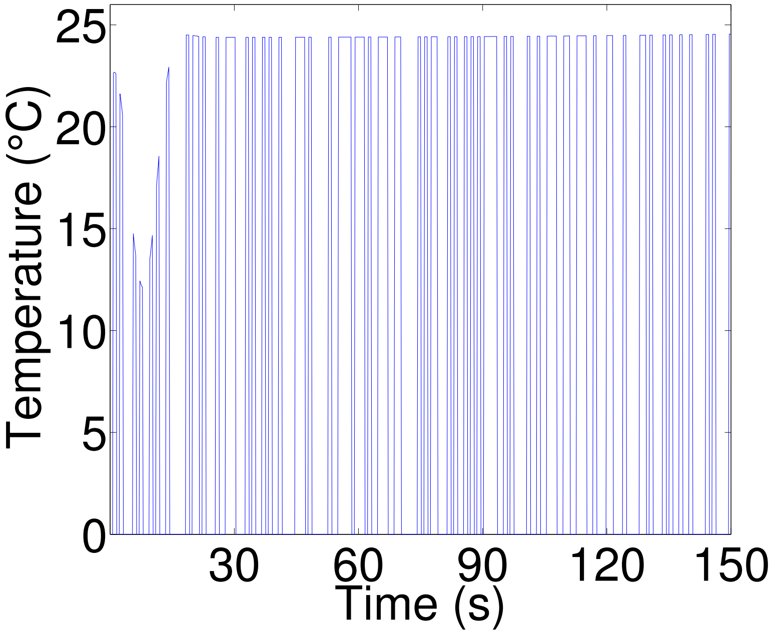


Figure 5.5: Temperature measurement before EKF and with 60% sensing packet drops.

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. 5.4 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, 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 worse case, after 8 consecutive packets are lost, the actuator will retain the value $u(k+7)$ until a new actuation packet arrives.

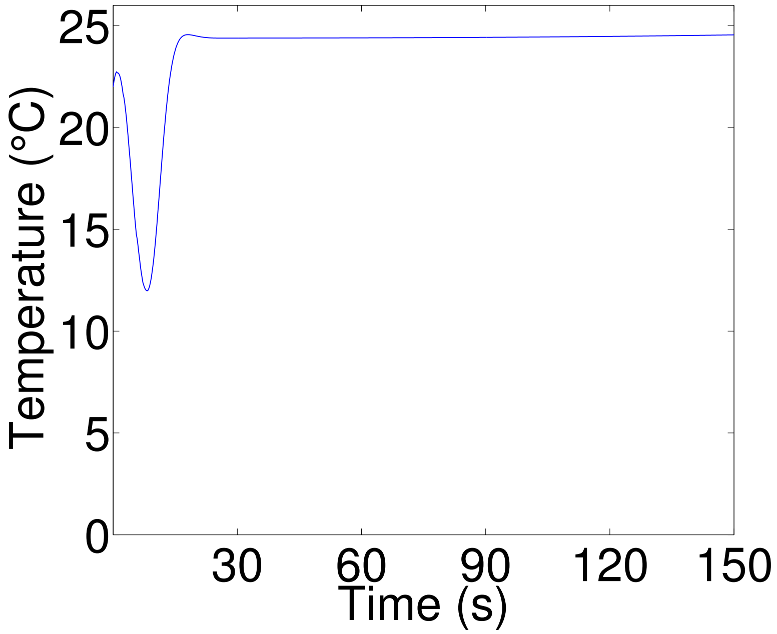


Figure 5.6: Temperature after EKF and with 60% sensing packet drops.

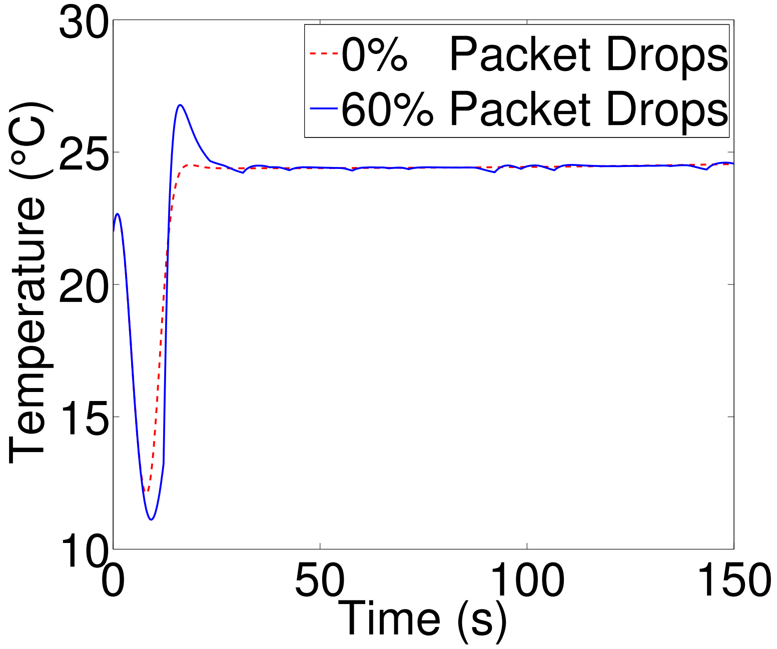


Figure 5.7: Temperature measurement with 60% actuation packet drops.

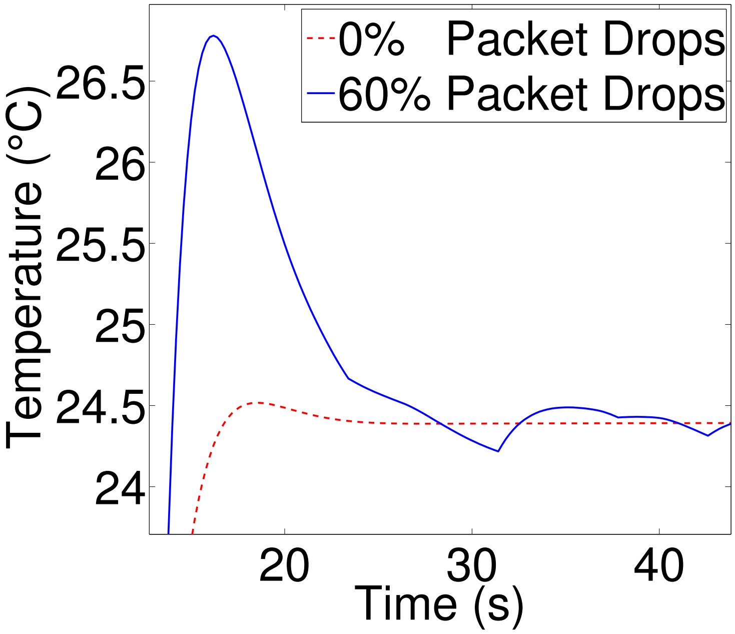


Figure 5.8: Zoomin View for Temperature measurement with 60% actuation packet drops.

5.4.4 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 [46]. We have implemented the asymmetric routing framework that allows any combinations 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 [4]. 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

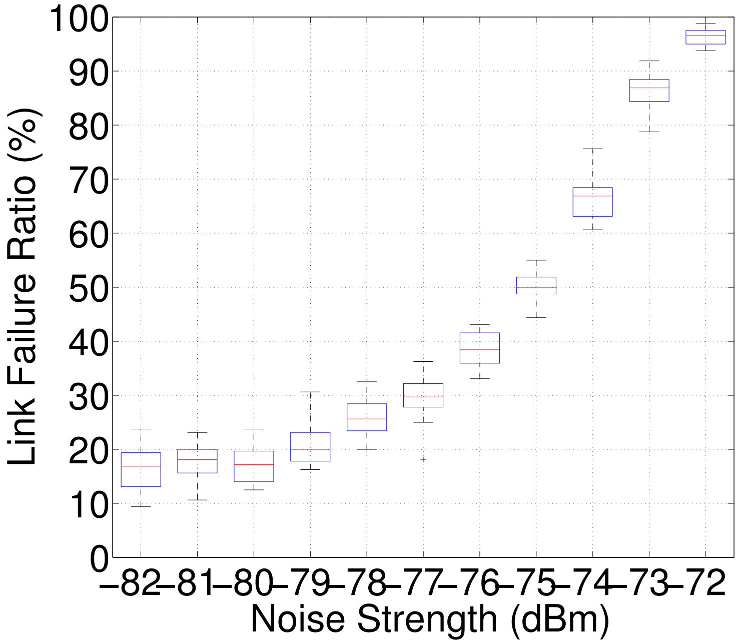


Figure 5.9: Link Failure Ratio.

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. 5.9 shows link failure ratios under different average noise strength. 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 represent extreme conditions such as adversarial jamming attacks or extreme weather conditions.

5.5 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. 5.2). 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 cyber-physical simulations.

5.5.1 EKF and Buffered Actuation Results

We tested our network under high stress conditions for the resilience and robustness of our closed-loop implementation. Figs. 5.5 and 5.6 show the performance of the EKF when 60% of the sensing packets are dropped. We observe the EKF does a very good job at filtering out jitters caused by sensing packet drops. Fig. 5.7 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. 5.8 reveals an overshoot of over 2 °C, which is 10 times larger than the overshoot with no packet drop. These results support our conjecture that actuation is more vulnerable to packet drops than sensing.

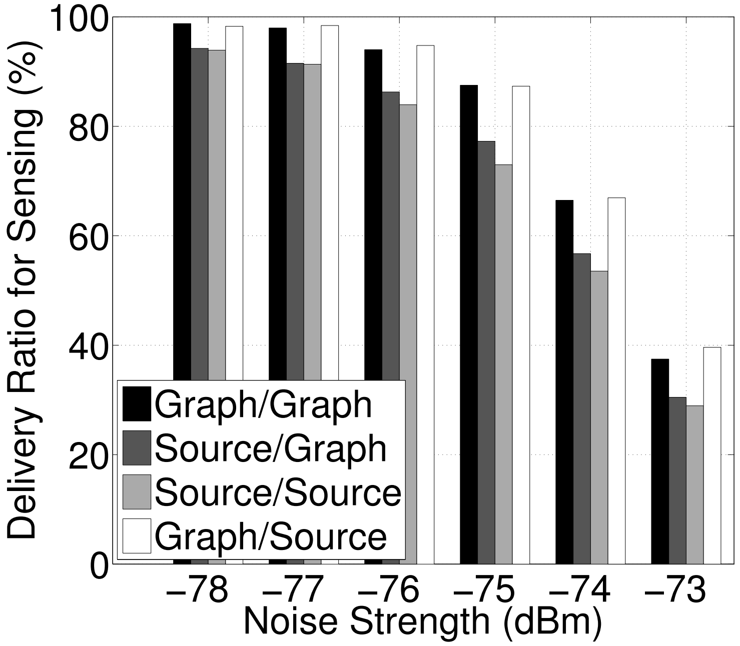


Figure 5.10: Sensing Delivery Ratio.

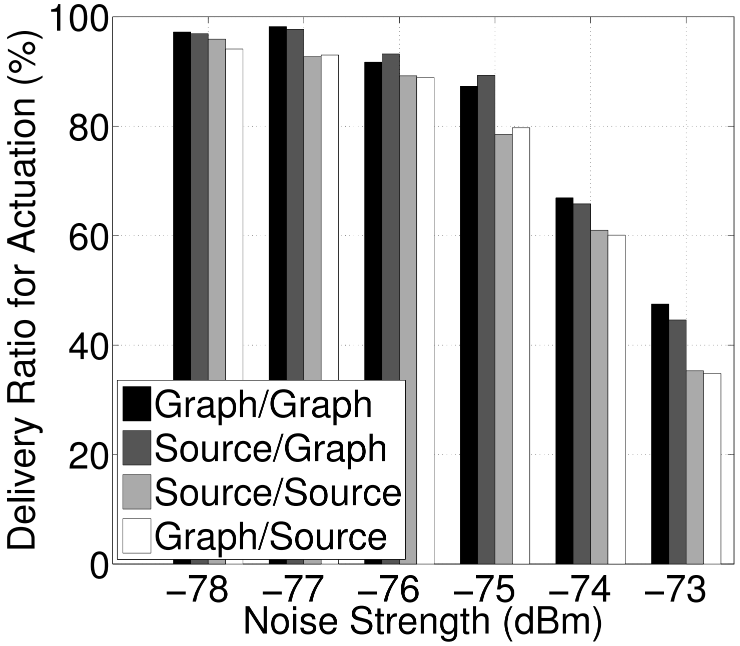


Figure 5.11: Actuation Delivery Ratio.

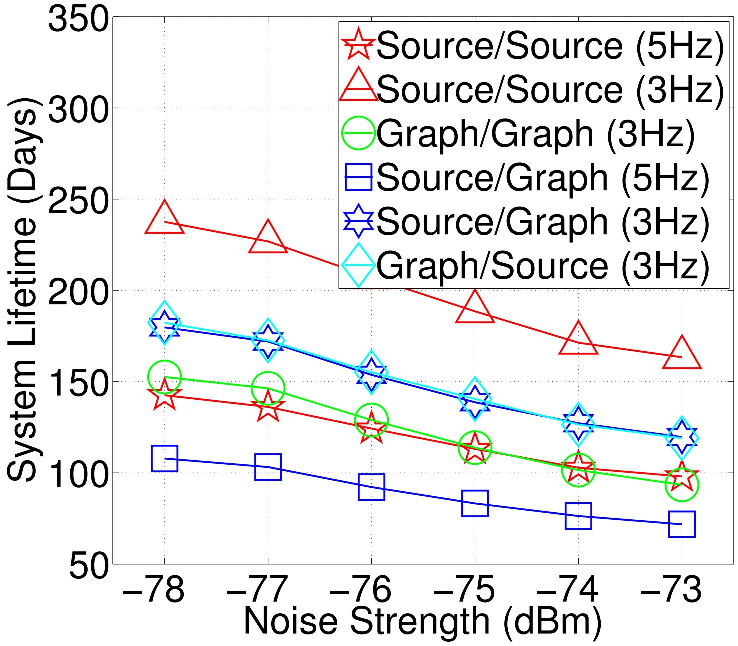


Figure 5.12: System Lifetime.

5.5.2 Network Results

To simplify our exposition, and follow the implementation described in Section **??**, 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 graph routing for actuation.

Fig. 5.10 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 increases. G/G and G/S consistently outperform the other approaches because of the redundancy offered by graph routing. Note that -74 dBm noise has over 60% link failures in Fig. 5.9, yet we still see G/G reaches nearly 60% multi-hop delivery ratios, which shows the strength of graph routing in improving network reliabilities.

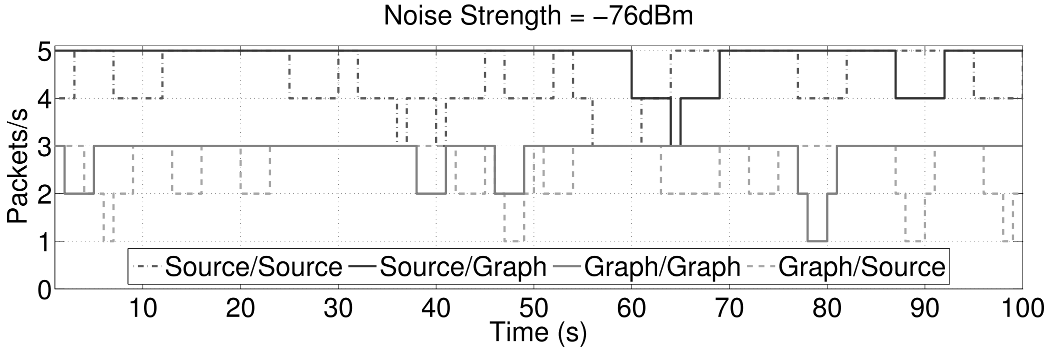


Figure 5.13: Data Rates for the Actuator

Fig. 5.11 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. For this reason 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 flows.

Fig. 5.13 shows actuation data rates at the signal strength of -76 dBm, where nearly 40% of the packets are dropped as shown in Fig. 5.9. S/S and S/G, which support up to 5 Hz flows, clearly show higher data rates. Moreover, we observe that S/G and G/G have less degradation due to the better reliability of graph routing in the actuation phase.

Fig. 5.12 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 increase, due to

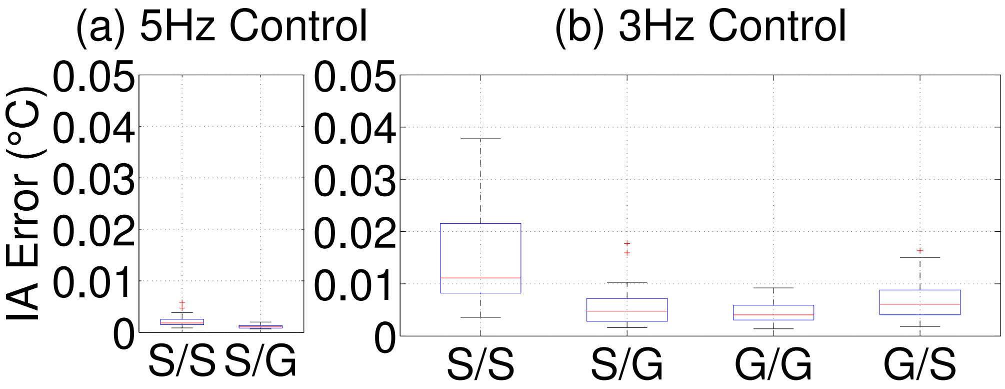


Figure 5.14: Integral Absolute Error for Temperature Control with Noise Strength -74 dBm.

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 the among the second best, which confirms the impact of control frequency on system lifetime.

5.5.3 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 as above.

Fig. 5.14 shows the IAE for the control, i.e., regulating T_2, with -74 dBm noise in the network. 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 show 3 Hz control with all four routing approaches, where G/G outperforms the other approaches, also as expected. A cross

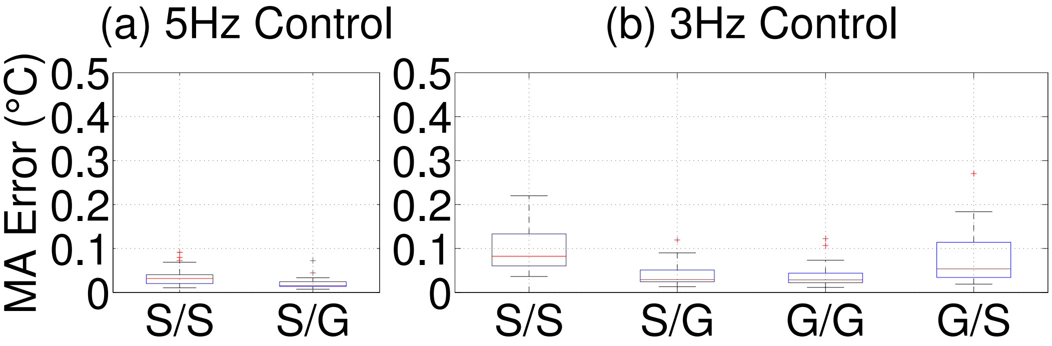


Figure 5.15: Maximum Absolute Error for Temperature Control with Noise Strength = -74 dBm.

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 trade-offs between WSAN reliability and control frequency.

Fig. 5.15 shows the MAE for the control goal. As in Fig. 5.14, we have 5 Hz results on the left and 3 Hz on the right. We observe a similar trends 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 attentions 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.

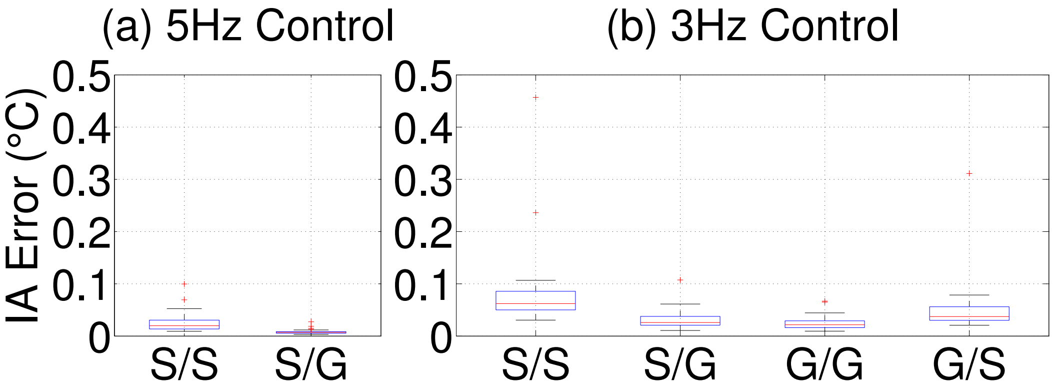


Figure 5.16: Integral Absolute Error for Temperature Control with Noise Strength = -73 dBm.

5.5.4 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. 5.9. We observe around 35% end-to-end delivery ratio for both sensing (Fig. 5.10) and actuation (Fig. 5.11).

Fig. 5.16 shows IAE errors of both 5Hz and 3Hz control. We note G/G and G/S can only support up to 3Hz control and Fig. 5.16 represent their best achievable performance.

In Fig. 5.16, we see S/G (5Hz) behaves the best among both 5Hz and 3Hz methods. We note the IAE error under -73dBm noise is 10 times larger than that in Fig. 5.14, which is because the former has 35% delivery ratio while the latter has 60%. Among 3Hz control in Fig. 5.16(b), we see G/G performs the best because it has better reliability for both sensing and actuation. G/S in Fig. 5.16(b) has larger IA error than S/G, which again proves that actuation data is more vulnerable to packet drops than sensing.

Fig. 5.17 shows the maximum absolute error with -73 dBm noise and around 65% data loss ratio for both sensing and actuation. We see significantly challenges on the control result, as

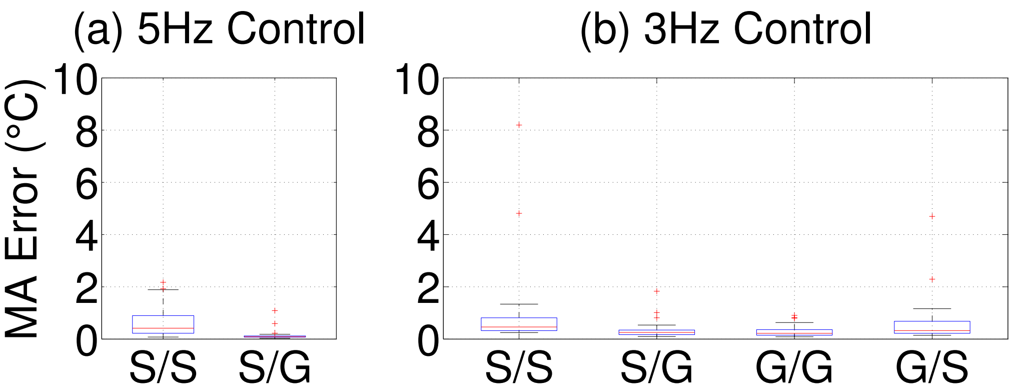


Figure 5.17: Maximum Absolute Error for Temperature Control with Noise Strength = -73 dBm.

the error reaches up to 2 °C in 5Hz control and 8 °C in 3Hz control. Given such challenging wireless condition, 5Hz control with S/G in Fig. 5.17(a) achieves the smallest MA error at a small variance. This again proves the value of asymmetric routing design in extremely challenging wireless control cases.

5.6 Summary

This chapter 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 network 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 cyber-physical simulations show 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 and control in wireless control systems.

Chapter 6

Conclusion

Wireless sensor-actuator network (WSAN) technology is gaining rapid adoption in process industries because of its advantages in lowering deployment and maintenance cost in challenging environments. While early success of industrial WSANs has been recognized, significant potential remains in exploring WSANs as unified networks for full-scale industrial plants. This thesis research explores a cyber-physical co-design approach for wireless control systems and presents four-fold contributions: (1) WCPS has been developed as the first simulator that features both linear and nonlinear physical plant models, state-of-art WirelessHART protocol stack, and realistic wireless network characteristics. (2) A realistic wireless structural control study sheds light on the challenges of wireless structural control and the limitations of a traditional structural control approach under realistic wireless conditions. (3) Systematic case studies demonstrate that our emergency communication approach enables effective emergency handling, while allowing regular control loops to share resources in WSANs during normal operations. (4) An in-depth study of wireless routing and control further highlights the importance of the co-design approach of wireless networks and control.

References

[1] http://wcps.cse.wustl.edu.

[2] http://shm.cs.uiuc.edu.

[3] http://www.hartcomm.org.

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

[5] ISA100: Wireless Systems for Automation. http://www.isa.org/MSTemplate.cfm?MicrositeID=1134&CommitteeID=6891.

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

[7] WiSA: Wireless sensor and actuator networks for measurement and control. http://www.control.hut.fi/Research/wisa.

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

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

[10] PAFNA Afonso, JML Ferreira, and JAAM Castro. Sensor fault detection and identification in a pilot plant under process control. *Chemical Engineering Research and Design*, 76(4):490–498, 1998.

[11] Jamal N Al-Karaki and Ahmed E Kamal. Routing techniques in wireless sensor networks: a survey. *Wireless communications, IEEE*, 11(6):6–28, 2004.

[12] Behdad Aminian, Jean Araujo, Mikael Johansson, and Karl H Johansson. Gisoo: a virtual testbed for wireless cyber-physical systems. In *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*, pages 5588–5593. IEEE, 2013.

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

[14] Karl Johan Aström and Richard M Murray. *Feedback systems: an introduction for scientists and engineers.* Princeton university press, 2010.

[15] Jia Bai, Emeka P. Eyisi, Fan Qiu, Yuan Xue, and Xenofon D. Koutsoukos. Optimal cross-layer design of sampling rate adaptation and network scheduling for wireless networked control systems. In *ICCPS*, 2012.

[16] Girish Baliga, Scott Graham, Lui Sha, and P.R. Kumar. Etherware: Domainware for wireless control networks. In *Proceedings of The 7th IEEE International Symposium on Object-oriented Real-time Distributed Computing*, 2004.

[17] Alberto Bemporad. Predictive control of teleoperated constrained systems with unbounded communication delays. In *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, volume 2, pages 2133–2138. IEEE, 1998.

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

[19] Matteo Ceriotti, Luca Mottola, Gian Pietro Picco, Amy L Murphy, Stefan Guna, Michele Corra, Matteo Pozzi, Daniele Zonta, and Paolo Zanon. Monitoring heritage buildings with wireless sensor networks: The torre aquila deployment. In *Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*, pages 277–288. IEEE Computer Society, 2009.

[20] Anton Cervin, Dan Henriksson, Bo Lincoln, Johan Eker, and Karl-Erik rzn. How does control timing affect performance? analysis and simulation of timing using jitterbug and truetime. *Proceedings of PWC 2003: Personal Wireless Communication, Lecture Notes in Computer Science*, 23(3):16 – 30, June 2003.

[21] Deji Chen, Mark Nixon, and Aloysius Mok. *WirelessHART^{TM} Real-Time Mesh Network for Industrial Automation.* Springer, 2010.

[22] Jinran Chen, Shubha Kher, and Arun Somani. Distributed fault detection of wireless sensor networks. In *Proceedings of the 2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks*, pages 65–72. ACM, 2006.

[23] Octav Chipara, Chenyang Lu, J.A. Stankovic, and G.-C. Roman. Dynamic conflict-free transmission scheduling for sensor network queries. *IEEE Transactions on Mobile Computing*, 10(5):734–748, May 2011.

[24] Octav Chipara, Chengjie Wu, Chenyang Lu, and William Griswold. Interference-Aware Real-Time Flow Scheduling for Wireless Sensor Networks. In *ECRTS'11*, 2011.

[25] Panagiotis D Christofides and Nael H El-Farra. *Control of nonlinear and hybrid process systems: Designs for uncertainty, constraints and time-delays*, volume 324. Springer New York, 2005.

[26] L.L. Chung, C.C. Lin, and K.H. Lu. Time-delay control of structures. *Earthquake Engineering and Structural Dynamics*, 24(5):687–701, 1995.

[27] B. Demirel, Zhenhua Zou, P. Soldati, and M. Johansson. Modular co-design of controllers and transmission schedules in wirelesshart. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 5951–5958, 2011.

[28] S. Dyke, J. Caicedo, G. Turan, L. Bergman, and S. Hague. Phase i benchmark control problem for seismic response of cable-stayed bridges. *Journal of Structural Engineering*, 129(7):857–872, 2003.

[29] Emeka Eyisi, Jia Bai, Derek Riley, Jiannian Weng, Yan Wei, Yuan Xue, Xenofon D. Koutsoukos, and Janos Sztipanovits. Ncswt: An integrated modeling and simulation tool for networked control systems. In *The 15th International Conference on Hybrid Systems: Computation and Control (HSCC)*, 2012.

[30] Carlos E Garcia, David M Prett, and Manfred Morari. Model predictive control: theory and practice?a survey. *Automatica*, 25(3):335–348, 1989.

[31] Gregory Hackmann, Weijun Guo, Guirong Yan, Chenyang Lu, and Shirley Dyke. Cyber-physical codesign of distributed structural health monitoring with wireless sensor networks. In *ICCPS*, pages 119–128, April 2010.

[32] Gregory Hackmann, Fei Sun, Nestor Castaneda, Chenyang Lu, and Shirley Dyke. A holistic approach to decentralized structural damage localization using wireless sensor networks. In *RTSS*, pages 35–46, December 2008.

[33] Song Han, Xiuming Zhu, A.K. Mok, Deji Chen, and M. Nixon. Reliable and real-time communication in industrial wireless mesh networks. In *Real-Time and Embedded Technology and Applications Symposium (RTAS), 2011 17th IEEE*, 2011.

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

[35] S. Jang, H. Jo, S. Cho, K. Mechitov, J. Rice, S.-H. Sim, H.-J. Jung, C.-B. Yun, B.F. Spencer, and G. Agha. Structural health monitoring of a cable-stayed bridge using smart sensor technology: deployment and evaluation. *Smart Structures and Systems*, 6(5):439–460, 2010.

[36] A. Jindal and M. Liu. Networked computing in wireless sensor networks for structural health monitoring. In *SPIE Symposium on Smart Structures and Materials, Nondestructive Evaluations and Health Monitoring, San Diego, CA*, March 2011.

[37] Karl Henrik Johansson, Magnus B. Egerstedt, John Lygeros, and S. Shankar Sastry. On the Regularization of Zeno Hybrid Automata. *Systems & Control Letters*, 38(3):141–150, 1999.

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

[39] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steven Glaser, and Martin Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *IPSN*, 2007.

[40] Kevin Klues, Gregory Hackmann, Octav Chipara, and Chenyang Lu. A component-based architecture for power-efficient media access control in wireless sensor networks. In *Sensys*, 2007.

[41] Xenofon Koutsoukos, Nicholas Kottenstette, Joe Hall, Panos Antsaklis, and Janos Sztipanovits. Passivity-based control design for cyber-physical systems. In *International Workshop on Cyber-Physical Systems-Challenges and Applications*, 2008.

[42] HyungJune Lee, Alberto Cerpa, and Philip Levis. Improving wireless simulation through noise modeling. In *IPSN*, 2007.

[43] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim: Accurate and scalable simulation of entire tinyos applications. In *Sensys*, 2003.

[44] Bo Li, Datong Liu, and Yu Peng. A high speed dma transaction method for pci devices. *Journal of Electronic Measurement and Instrument*, 22:705–716, 2008.

[45] Bo Li, Yehan Ma, Tyler Westenbroek, Humberto Gonzalez, and Chenyang Lu. Wireless routing and control: a cyber-physical case study. 2016.

[46] Bo Li, Lanshun Nie, Chengjie Wu, Humberto Gonzalez, and Chenyang Lu. Incorporating emergency alarms in reliable wireless process control. In *ACM/IEEE International Conference on Cyber-Physical Systems*, 2015.

[47] Bo Li, Zhuoxiong Sun, Kirill Mechitov, Chenyang Lu, Shirley Dyke, Gul Agha, and Billie Spencer. Realistic case studies of wireless structural control. In *ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'13)*, April 2013.

[48] Bo Li, Dan Wang, and Yi Qing Ni. Demo: An imote2 compatible high fidelity sensing module for shm sensor networks. In *IEEE INFOCOM*, 2010.

[49] Bo Li, Dan Wang, and Yiqing Ni. Demo: On the high quality sensor placement for structural health monitoring. In *IEEE INFOCOM*, pages 1–2. Citeseer, 2009.

[50] Bo Li, Dan Wang, Feng Wang, and YiQing Ni. High quality sensor placement for SHM systems: Refocusing on application demands. In *Proc. IEEE INFOCOM'10, San Diego, CA, Mar.*, 2010.

[51] H. Li, Z. Sun, M. Chow, and B. Chen. State feedback controller design of networked control systems with time delay and packet dropout. In *Proceedings of 17th World Congress The International Federation of Automatic Control*, July 2008.

[52] Xiangheng Liu and Andrea Goldsmith. Kalman filtering with partial observation losses. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 4, pages 4180–4186. IEEE, 2004.

[53] Xuefeng Liu, Jiannong Cao, Wen-Zhan Song, and Shaojie Tang. Distributed sensing for high quality structural health monitoring using wireless sensor networks. In *The 33rd IEEE Real-Time Systems Symposium (RTSS'12)*, 2012.

[54] 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.

[55] JP Lynch, Y Wang, RA Swartz, KC Lu, and CH Loh. Implementation of a closed-loop structural control system using wireless sensor networks. *Structural Control and Health Monitoring*, 15(4):518–539, 2008.

[56] YiQing Ni, Bo Li, K.H. Lam, Dapeng Zhu, Yang Wang, Jeremy Lynch, and K.H. Law. In-construction vibration monitoring of a super-tall structure using a long-range wireless sensing system. *Smart Structures and Systems*, 7(2):83–102, March 2011.

[57] M. Pajic, S. Sundaram, J. Ny, G. Pappas, and R. Mangharam. Closing the loop: A simple distributed method for control over wireless networks. In *IPSN*, 2012.

[58] Miroslav Pajic, Shreyas Sundaram, Jerome Le Ny, George J. Pappas, and Rahul Mangharam. Closing the loop: a simple distributed method for control over wireless networks. In *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, IPSN '12, pages 25–36, New York, NY, USA, 2012. ACM.

[59] Yu Peng, Bo Li, Datong Liu, and Xiyuan Peng. A high speed dma transaction method for pci express devices. In *Testing and Diagnosis, 2009. ICTD 2009. IEEE Circuits and Systems International Conference on*, pages 1–4. IEEE, 2009.

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

[61] Joseph Polastre, Robert Szewczyk, and David Culler. Telos: enabling ultra-low power wireless research. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, IPSN '05, Piscataway, NJ, USA, 2005. IEEE Press.

[62] Injong Rhee, Ajit Warrier, Mahesh Aia, Jeongki Min, and Mihail L. Sichitiu. Z-mac: a hybrid mac for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 16(3):511–524, June 2008.

[63] A. Saifullah, P. Tiwari, B. Li, C. Lu, and Y. Chen. Accounting for failures in delay analysis for wirelesshart networks. In *Tech. Rep., WUCSE-2012-16, Washington University in St Louis*, 2012.

[64] Abusayeed Saifullah, Dolvara Gunatilaka, Paras Tiwari, Mo Sha, Chenyang Lu, Bo Li, Chengjie Wu, and Yixin Chen. Schedulability analysis under graph routing in wirelesshart networks. In *RTSS'15*, 2015.

[65] Abusayeed Saifullah, Chengjie Wu, Paras Tiwari, You Xu, Yong Fu, Chenyang Lu, and Yixin Chen. Near optimal rate selection for wireless control systems. In *RTAS,12*, 2012.

[66] Abusayeed Saifullah, You Xu, Chenyang Lu, and Yixin Chen. Real-time scheduling for WirelessHART networks. In *RTSS*, 2010.

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

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

[69] Bruno Sinopoli, Luca Schenato, Massimo Franceschetti, Kameshwar Poolla, and Shankar S Sastry. Time varying optimal control with packet losses. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 2, pages 1938–1943. IEEE, 2004.

[70] B.F. Spencer, S. Dyke, and H. Deoskar. Benchmark problems in structural control: part i-active mass driver system. *Earthquake Engineering and Structural Dynamics*, 27(11):1127–1139, 1998.

[71] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. An empirical study of low power wireless. *ACM Transactions on Sensor Networks*, 2010.

[72] Zhuoxiong Sun, Bo Li, Shirley Dyke, and Chenyang Lu. Evaluation of performances of structural control benchmark problem with time delays from wireless sensor network. In *Joint Conference of the Engineering Mechanics Institute and ASCE Joint Specialty Conference on Probabilistic Mechanics and Structural Reliability (EMI/PMC'12)*, 2012.

[73] Zhuoxiong Sun, Bo Li, Shirley J. Dyke, and Chenyang Lu. Benchmark problem in active structural control with wireless sensor network. *Structural Control and Health Monitoring*, 2015.

[74] Zhuoxiong Sun, Bo Li, S.J. Dyke, and Chenyang Lu. A novel data utilization and control strategy for wireless structural control systems with tdma network. In *Proc. ASCE IWCCE 2013*, 2013.

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

[76] Y. Wang, R.A. Swartz, J.P. Lynch, K.H. Law, and C.-H. Loh. Performance evaluation of decentralized wireless sensing and control in civil structures. In *Proceedings of SPIE 14th International Symposium on Smart Structures and Materials and Nondestructive Evaluation and Health Monitoring*, March 2007.

[77] Yang Wang and Kincho Law. Structural control with multi-subnet wireless sensing feedback: experimental validation of time-delayed decentralized h? control design. *Advances in Structural Engineering*, 14(1):25–39, 2011.

[78] Yongqiang Wang, Steven X Ding, Hao Ye, and Guizeng Wang. A new fault detection scheme for networked control systems subject to uncertain time-varying delay. *Signal Processing, IEEE Transactions on*, 56(10):5258–5268, 2008.

[79] Chengjie Wu, Dolvara Gunatilaka, Abusayeed Saifullah, Mo Sha, Paras Tiwari, Chenyang Lu, and Yixin Chen. Maximizing network lifetime of wireless sensor-actuator networks under graph routing. In *Technical Report Number: WUCSE-2015-004 (2015). All Computer Science and Engineering Research*, 2015. `http://openscholarship.wustl.edu/cse_research/508/`.

[80] Chengjie Wu, Dolvara Gunatilaka, Mo Sha, Chenyang Lu, and Yixin Chen. Conflict-aware real-time routing for industrial wireless sensor-actuator networks. In *Technical Report WUCSE-2015-005 (2015). All Computer Science and Engineering Research*, 2015. `http://openscholarship.wustl.edu/cse_research/507/`.

[81] Chengjie Wu, Mo Sha, Dolvara Gunatilaka, Abusayeed Saifullah, Chenyang Lu, and Yixin Chen. Analysis of EDF Scheduling for Wireless Sensor-Actuator Networks. In *IEEE/ACM Symposium on Quality of Service (IWQoS'14)*, May 2014.

[82] Sadaf Zahedi, Marcin Szczodrak, Ping Ji, Dinkar Mylaraswamy, Mani Srivastava, and Robert Young. Tiered architecture for on-line detection, isolation and repair of faults in wireless sensor networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1–7. IEEE, 2008.

[83] Zinon Zinonos, Ricardo Silva, Vasos Vassiliou, and Jorge Sa Silva. Mobility solutions for wireless sensor and actuator networks with performance guarantees. In *Telecommunications (ICT), 2011 18th International Conference on*, pages 406–411. IEEE, 2011.

Vita

Bo Li

Degrees

Ph.D. Computer Science, 2015
M.S. Computer Science, 2013
M.E. Electrical Engineering, 2008
B.E. Electrical Engineering, 2006

Publications

B. Li, Y. Ma, T. Westenbroek, C. Wu, H. Gonzalez and C. Lu, Wireless Routing and Control: a Cyber-Physical Case Study, submitted to ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'16), April 2016.

B. Li, L. Nie, C. Wu, H. Gonzalez, C. Lu, and L. Linderman, Incorporating Emergency Alarms in Reliable Wireless Process Control, In ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'15), April 2015.

B. Li, Z. Sun, K. Mechitov, C. Lu, S. J. Dyke, G. Agha, and B. F. Spencer, Realistic Case Studies of Wireless Structural Control, In ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'13), April 2013.

Z. Sun, B. Li, S. J. Dyke, and C. Lu, Benchmark problem in active structural control with wireless sensor network, Structural Control and Health Monitoring, 2015.

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, In Proceedings of the IEEE, accepted to appear.

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 (RTSS'15), December 2015.

Z. Sun, B. Li, S. J. Dyke, and C. Lu. A novel data utilization and control strategy for wireless structural control systems with TDMA network. In 2013 ASCE International Workshop on Computing in Civil Engineering (ASCE IWCCE 2013), pp. 23-25. 2013.

Z. Sun, B. Li, S. J. Dyke, and C. Lu, Evaluation of Performances of Structural Control Benchmark Problem With Time Delays From Wireless Sensor Network, In EMI/PMC 2012, Notre Dame, IN, June 17-20, 2012.

B. Li, D. Wang, F. Wang, and Y.Q. Ni, High Quality Sensor Placement for SHM Systems: Refocusing on Application Demands, In IEEE INFOCOM'10, San Diego, CA, Mar. 2010.

Y. Peng, B. Li, D. Liu, and X. Peng, A high speed DMA transaction method for PCI Express Devices, In IEEE Circuits and Systems International Conference on Testing and Diagnosis, Chengdu, China, Apr. 2009.

Y.Q. Ni, B. Li, K.H. Lam, D. Zhu, Y. Wang, J.P. Lynch, and K.H. Law, In-construction vibration monitoring of a super-tall structure using a long-range wireless sensing system, Smart Structures and Systems, 7(2): 83-102, 2011.

B. Li, Y. Peng, D. Liu, and X. Peng. A high speed DMA transaction method for PCI Express Devices, Journal of Electronic Science and Technology of China, Vol. 7, No.4, Dec. 2009.

B. Li, D. Liu, and Y. Peng. A high speed DMA transaction method for PCI Devices, In Journal of Electronic Measurement and Instrument 22 (1), 705-710, China, 2008.

B. Li, D. Wang, and Y.Q. Ni, Demo: An Imote2 Compatible High Fidelity Sensing Module for SHM Sensor Networks, In IEEE INFOCOM'10, San Diego, CA, Mar. 2010.

B. Li, D. Wang, and Y.Q. Ni, Demo: On the High Quality Sensor Placement for Structural Health Monitoring, In IEEE INFOCOM'09, Rio de Janeiro, Brazil, Apr. 2009.

December 2015