Networking Infrastructure and Data Management for Cyber-Physical Systems

Song Han

Computer Science Department, University of Texas at Austin
shan@cs.utexas.edu
www.cs.utexas.edu/~shan
What is Cyber-Physical System (CPS)?

Cyber-physical system is a system featuring a tight combination of, and coordination between, the system’s computational and physical elements.
CPS Application – Cyberphysical Avatar

Dynamic Model and Control Structure Design

Skill Acquisition through Machine Learning

Real-time Avatar-Human Interaction

Prototype Testbed

Cyberphysical Avatar: A semi-autonomous robotic system (joint project with UT Human Centered Robotics Lab)
CPS Application – Network-based Mobile Gait Rehabilitation System

- Integrating heterogeneous sensors into real-time wireless platform
- Low-level motion control of rehabilitation device over wireless network
- Development of high-level decision making algorithm

Network-based Mobile Gait Rehabilitation System (joint project with Mechanical Systems Control Laboratory, UC Berkeley)
Research Overview

Theoretical Framework for Real-time Data Management Techniques

Real-time Wireless Communication Platform
Guiding Applications

Network-based Rehabilitation System

Cyberphysical Avatar

Remote and Real-time Welding System
Outline

• Research Overview

• Reliable and Real-time Wireless Platform for CPS
  – Wireless real-time communication protocol
  – Network management techniques
  – System design and implementation

• Real-time Data Management in CPS
  – Model and assumptions
  – Algorithms and analysis

• Summary and Future Work
Wireless Reliable and Real-time Communication Platform
Design Space and Required Features

- **Low-power**
  - 802.15.4-based radio

- **Real-time**
  - TDMA Data Link Layer (DLL)
  - Centralized management

- **Reliable**
  - Mesh networking
  - Data link layer ACK
  - Channel hopping mechanism

- **Secure**
  - Data integrity on DLL
  - Data confidentiality on network layer (NL)
Overview of Our Real-time Protocol Stack

- **TDMA-based Data Link Layer**
  - Guarantee timely delivery

- **Channel Hopping and Blacklisting**
  - Spread communication in all active physical channels
  - Reduce interference to provide reliable communication

- **Confidential and Secure Communication**
  - Use both public and private keys to secure communication in both join process and normal operations
TDMA-based Data Link Layer

10ms

Source

Destination
TDMA-based Data Link Layer

- Link: activity in a time slot
  - Neighbor
  - Send/Receive
  - Communication channel
TDMA-based Data Link Layer

• Superframe: a group of links
  – Repeat itself infinitely
  – A device can support several superframes
TDMA-based Data Link Layer

Priority queues for data link layer packets
How to Achieve Reliable and Real-time Services in CPS

- **Network Manager**
  - Authenticating devices
  - Forming the network
  - Constructing routing graphs
  - Scheduling DL transmissions

- **Gateway**
  - Collecting/caching sensor data
  - Process queries from other systems

- **Security Manager**
  - Manage key information
How to Achieve Reliable and Real-time Services in CPS

• Communication task definition
  – Need to solve two related sub-problems:
    1. communication graph design
    2. link scheduling

• Technical Objectives
  – Achieve reliable routing in wireless mesh networks
  – Achieve real-time communication by deterministic link and channel assignment
  – Evaluate their performance in real industrial environments
Communication Graph Design to Achieve Reliable Graph Routing

(a) Original network topology
(b) Uplink graph
(c) Broadcast graph
(d) Downlink graph to Dev 3 and Dev 4

To avoid forwarding loop:
1) Only one cycle of length 2 in $G_v$
2) Each DEV on the cycle has direct edges to $v$
Constructing Reliable Graphs

• Reliable Broadcast Graph and Uplink Graph
  – Grow the graph by greedily selecting the reliable node with minimum latency to the Gateway

• Standard Reliable Downlink Graph
  – Construct a completely new graph from GW to DEV
  – Configuration in intermediate nodes cannot be reused
  – High configuration cost and poor scalability
Sequential Reliable Downlink Routing (SRDR)

• Key Principles
  – Each node only keep a small local graph
  – Local graphs are reusable building blocks for constructing reliable downlink graph for multiple destinations

Low configuration cost
High Scalability
High Reliability
An Example of SRDR

(a) Original network topology

(b) Downlink graph: \( g_2 \)
Sequential route for Dev 2: \( g_2 \)

(c) Downlink graph: \( g_3 \)
Sequential route for Dev 3: \( g_3 \)

Avoid node failure at DEV2

(d) Downlink graph: \( g_1 \)
Sequential route for Dev 1: \( g_2, g_1 \)

Local graph

(e) Downlink graph: \( g_4 \)
Sequential route for Dev 4: \( g_2, g_1, g_4 \)

(f) Downlink graph: \( g_5 \)
Sequential route for Dev 5: \( g_2, g_5 \)
SRDR Extensions

![Diagram of SRDR Extensions]

Extended Routing Information

<table>
<thead>
<tr>
<th>Control</th>
<th>TTL</th>
<th>ASN Snippet</th>
<th>Graph ID</th>
<th>Dest Addr</th>
<th>Source Addr</th>
<th>Proxy Route</th>
<th>Payload</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>g2</td>
<td>g1 g4</td>
</tr>
</tbody>
</table>

21
SRDR Optimization

Extended Routing Information
Communication Link Scheduling

• The general scheduling problem is known to be NP-hardness [Saifullah et al. 2010]

• Key Principles:
  – Spread out the channel usage in the network
  – Apply Fastest Sample Rate First policy (FSRF)
  – Allocate the links iteratively from Src to Dest
  – Split traffic (bandwidth) among all successors
Example Schedule Construction Using the Key Principles

Channel offset will be converted into practical channel number in the runtime

Global Channel-Time Slot Matrix

Device Schedule

1 sec 2 sec 1 sec
Performance Evaluation
System Design, Implementation and Deployment
System Design, Implementation and Deployment

Hardware Platforms

- Freescale 1322x SRB Evaluation Board
- Custom Designed Mother Board with Sensor Support
- Custom Designed Board with EnergyMicro EFM32 MCU
System Design, Implementation and Deployment (Cont.)

Compliance Testing Suite

Testing Engine  
16-Channel Sniffer  
Virtual Network Approach
Network Manager and Simulator

Simulating a real-time wireless network with 100 devices:
- reliable broadcast graph
- device communication schedule
System Design, Implementation and Deployment (Cont.)

Network Manager and Simulator

Simulating a real-time wireless network with 100 devices:
- reliable uplink graph
- device bandwidth utilization
System Design, Implementation and Deployment (Cont.)

Application Layer

CoAP APP Layer

Socket API

Transport Layer

UDP

ICMP

6LoWPAN

Enhanced NWK Layer

Data Link Layer

802.15.4 PHY

Network Topology

CoAP-HTTP Server

Intra-system Service

Web Service
System Design, Implementation and Deployment (Cont.)

10 Device Testbed  UT Austin ACES 5th floor  UT Pickle Research Center  UWO Power House
Higher Sampling Rate Required in Network-based Rehabilitation System

• Challenges
  – Mechanic modules need high frequency and low jitter control
  – A platform for a wide range of wireless control applications: a good balance among sampling rate, energy consumption and real-time performance
High-speed Real-time Wireless Control

- Real-time Wi-Fi to support high speed control
  - Replacing 802.15.4 PHY with 802.11 PHY
  - Network-wide synchronization and power saving
Real-time Data Management in CPS
Maintaining Data Quality in CPS is Key

- CPS are in essential sensing and control systems
- Data quality is the key to the success of sensing and control applications
- Sensor data have time semantics, and their quality degrade with time
Maintaining Data Quality in CPS is Key

• Need to enable tradeoff between data quality and sampling rate
  – High sampling rate -> high network traffic & CPU workload
  – More power consumption & shorter network lifetime
  – Reduce sampling rate but maintain data and control quality

• Will exploit concept of validity interval to make the tradeoff
Task Model

A task is an abstraction of resource consumer; a task can be a computing task (consuming CPU cycles) or a communication task (consuming network bandwidth).

Validity intervals quantify the quality of sensor data.

Control data quality is a function of sensor data quality.
Task Model

- Sensor update task set $T^u = \{\tau_i^u\}_{i=1}^n$
  - $\tau_i^u$ is a 4-tuple: $\tau_i^u = (C_i^u, V_{i_{\text{min}}}^u, V_{i_{\text{max}}}^u, Q_{i\text{.}}^u(t))$.
  - $Q_{i\text{.}}^u(t)$ is application-dependent.

- Control task set $T^c = \{\tau_i^c\}_{i=1}^m$
  - $\tau_i^c$ is a 5-tuple: $\tau_i^c = (C_i^c, D_i^c, P_i^c, \Omega_i, Q_{i\text{.}}^c(t))$
  - $\Omega_i$ is the update tasks that $\tau_i^c$ will access and $Q_{i\text{.}}^c(t)$ is application-dependent.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_i^{u(c)}$</td>
<td>Update/Control Task $i$</td>
</tr>
<tr>
<td>$C_i^{u(c)}$</td>
<td>WCET for $\tau_i^{u(c)}$</td>
</tr>
<tr>
<td>$Q_i^{u(c)}(t)$</td>
<td>Quality function for $\tau_i^{u(c)}$</td>
</tr>
<tr>
<td>$V_{i_{\text{min}}}^c$</td>
<td>Min(max) validity interval</td>
</tr>
<tr>
<td>$D_i^c (P_i^c)$</td>
<td>Deadline (Period) of $\tau_i^c$</td>
</tr>
</tbody>
</table>

Goal: Maintain the control data quality above threshold while Minimizing update workload
Task Model

• Sensor update task set $T^u = \{\tau^u_i\}_{i=1}^n$
  – $\tau^u_i$ is a 4-tuple: $\tau^u_i = (C^u_i, V^\text{min}_i, V^\text{max}_i, Q^u_i(t))$.
  – $Q^u_i(t)$ is application-dependent.

• Control task set $T^c = \{\tau^c_i\}_{i=1}^m$
  – $\tau^c_i$ is a 5-tuple: $\tau^c_i = (C^c_i, D^c_i, P^c_i, \Omega_i, Q^c_i(t))$
  – $\Omega_i$ is the update tasks that $\tau^c_i$ will access and $Q^c_i(t)$ is application-dependent

• Simplifying Assumptions
  – No control task in the system for now
  – $V^\text{min}_i = 0$ and $V_i = V^\text{max}_i$

Validity Constraint: An update job must finish before its previous job’s validity interval expires

Goal: Guaranteeing validity constraint while minimizing the update workload.

Symbol | Meaning
--- | ---
$\tau^u_i$ | Update/Control Task $i$
$C^u_i$ | WCET for $\tau^u_i$
$Q^u_i(t)$ | Quality function for $\tau^u_i$
$V^\text{min, max}_i$ | Min(max) validity interval
$D^c_i$ | Deadline (Period) of $\tau^c_i$
From **Validity Interval Model** to **Periodic Task Model**

**Validity Interval Task Model**

How to pick the time point to perform sensing?

**Periodic Task Model**

How to pick the time point to perform sensing?
• **HH (Half-Half) Algorithm**
  - Period ($P_i$) and relative deadline ($D_i$) of an update task $i$ are each set to be one-half of the data validity length ($V_i$).
Maintaining Update Data Freshness - Baseline Scheduling Techniques

• ML (More-Less) Algorithm
  – Relative deadline ($D_i$) of an update task $i$ is set to be its worst-case response time (WCRT). Period $P_i = V_i - D_i$
Deferrable Scheduling with Fixed Priority (DS-FP)  
- From Periodic to Sporadic Task Model

**Principles**

- Adopts the sporadic task model.
- Defers the sampling time of the update job as late as possible to increases the distance of two consecutive jobs.
Deferrable Scheduling with Fixed Priority (DS-FP) - From Periodic to Sporadic Task Model

ML Schedule (Periodic)
P₁ = 10 \ D₁ = 2
P₂ = 14 \ D₂ = 4

DS-FP Schedule (Sporadic)

Release time of T₂₁ is deferred from 14 to 16

Separation time is increased
Deferrable Scheduling with Fixed Priority (DS-FP) - From Periodic to Sporadic Task Model

ML Schedule (Periodic)

\[ \begin{align*}
\text{P}_1 &= 10, \text{D}_1 = 2 \\
\text{P}_2 &= 14, \text{D}_2 = 4 \\
\end{align*} \]

**ML Schedule**

\[\begin{align*}
\text{T}_1: \{ C_1 = 2, V_1 = 12 \} & \quad \text{T}_2: \{ C_2 = 2, V_2 = 18 \}
\end{align*} \]

**Separation time is increased**

**Deadline of \( T_{2,2} \) is deferred to 34**
Deferrable Scheduling with Fixed Priority (DS-FP) - From Periodic to Sporadic Task Model

T₁: \{ C₁ = 2, V₁ = 12 \}  \quad \text{T₂: } \{ C₂ = 2, V₂ = 18 \}

ML Schedule (Periodic)

P₁ = 10 \quad D₁ = 2
P₂ = 14 \quad D₂ = 4

DS-FP Schedule (Sporadic)

Release time of T₂₂ is deferred from 28 to 32

Separation time is increased
Deferrable Scheduling with Fixed priority (DS-FP)

DS-FP significantly reduces the CPU workload incurred by update trans.

Lower priority tasks have larger relative avg. sampling periods
Deferrable Scheduling with Fixed priority (DS-FP)

• Comparison of DS-FP and ML
  – **THEOREM.** Given a synchronous update transaction set $T$ with known $C_i$ and $V_i$, if for all $i$, $f_{i,0}^{m} \leq V_i / 2$, then $T$ is schedulable with DS-FP.

• Necessary and Sufficient Schedulability Test
  – **THEOREM.** Given an update task set $T$, if it can be scheduled by DS-FP in the bounded time interval $[0, V_m - C_m + \prod_{i=1}^{m} (V_i - C_i + 1) - 1]$, then the schedule has a repeating pattern that must occur at least once in the bounded time interval $[V_m - C_m, V_m - C_m + \prod_{i=1}^{m} (V_i - C_i + 1) - 1]$.

• Overhead Reduction Algorithms
  – DS with Hyperperiod by Schedule Construction (DESH-SC)
  – DS with Hyperperiod by Schedule Adjustment (DESH-SA)
CPS Real-time Data Management Research Roadmap

- Maintaining data freshness for fixed update task set
  - Periodic task model
    - No Jitter
      - Half-Half
      - More-Less
      - JBML
      - SJBML
  - Jitter
    - Sporadic task model
      - Co-scheduling of update tasks and control tasks
        - Maintaining data freshness in flexible cyber-physical systems
          - Algorithm and analysis
            - Overhead reduction
            - Schedulability test
          - Deferrable Scheduling
Research Summary

• Reliable and real-time wireless platform for CPS
  – Wireless real-time communication protocol
  – Network management techniques
  – System design and implementation

• Theoretical framework for real-time data management in CPS
  – Models and assumptions
  – Algorithms and schedulability analysis
Ongoing and Future Work

• I believe that the next Internet resolution will be about the delivery of physical services in addition to information services over long distances.

• The economic and social impact will be enormous.
Thanks and Questions?