An Example of Wireless Distributed Computing Networks on CORNET

Xuetao Chen, Tamal Bose, Haris Volos, Joseph D. Gaeddert, and Jeffrey H. Reed

ECE Department, University of Arizona
Wireless@VT, Virginia Tech

Wireless Innovation Forum 2012
January 2013

Outline

Introduction

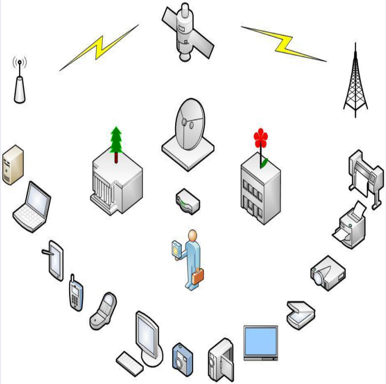
Motivation

1. Progress of wireless communication and computing technologies.
   - 4G systems, cloud computing, mobile Internet
2. Evolution of wireless devices.
   - Current smart phone has the comparable computing capability of desktops we have ten years ago.

Why WDCN

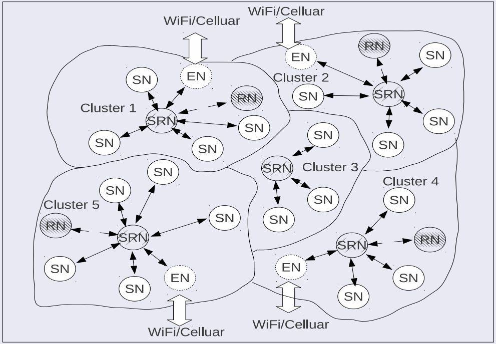
1. Service closer to users
2. Lessons learned from PC
3. Potential energy saving and performance improvement, e.g. response time (RT)

WDCNs



- Heterogeneous network
  - Channel, Device, and Application
- Categories
  - Service Provided
    - Utility-oriented
    - Computation-oriented
    - Data-oriented
  - Architecture
    - Loosely coupled
    - Closely coupled
- Network topology is application dependent.
- “No one-size-fits-all" solution for resource allocation.

WDCNs



- Request node (RN) has a computing task to be allocated. Service Nodes (SNs) have the computing resources. Service request node (SRN) is the agent for RN and manage workload/resource allocation. Edge node (EN) has an access to external networks.
- Cluster based flat network through wireless channels.

Research Challenges

Energy/Power Efficiency

- Distributed computing over unreliable channel
- Communication power consumption is the “price" to pay

Latency Performance

- Latency determined by the slowest node.
- Flatten network topology may help.

Software Architecture

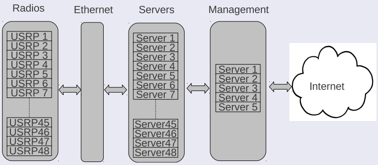
- Resource description
- Service discovery
- Fault tolerance
- Security

Protocol Design

- Wireless channel is the driving factor.
- Efficiency and Robustness.
- Cross-layer issues.

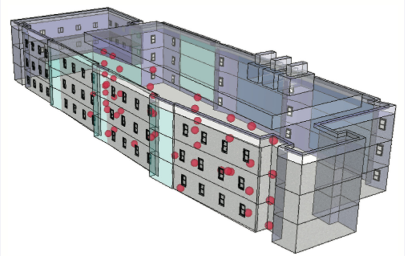
CORNET

System Architecture



- One node = one USRP2 + one Server
- Remote log in and control capability
- Ethernet with CAT6 cable for management network
- Provides hardware/software resource for Cognitive Radio Network research

Node Distribution



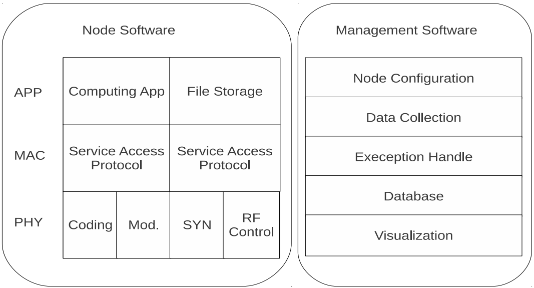
- 48 nodes throughout ICTAS building.
- 12 nodes on each floor
- In progress of upgrading to UHD driver
- Remote power cycle in plan
- mobile node in plan

Considerations

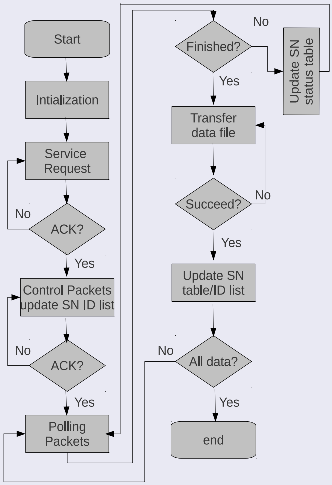
- Flexibility to change radio parameters, MAC protocol, and network topology
    - Fully control of the protocol stack
    - Open source code for protocol implementation
    - Potential data mining capability with server cluster
    - Integration of hybrid nodes, e.g. mobile nodes.
- Limitation of node structure
    - Latency and jitters between GPP and USRP
    - Contention based MAC, such as CSMA, is low efficient with large number of simultaneously Tx nodes.
- Tradeoffs
    - Splitting MAC between USRP and host server
    - TDMA protocol with sufficient guarding time.

Software Design

- Two Parts: Node software and management software
- Three Layers: PHY, MAC, APP
- Four Functions for MAC:
  - Service Access: *wdc_control_tx* and *wdc_control_rx*
  - Service Provision: *wdc_tx_buffer* and *wdc_rx_buffer*

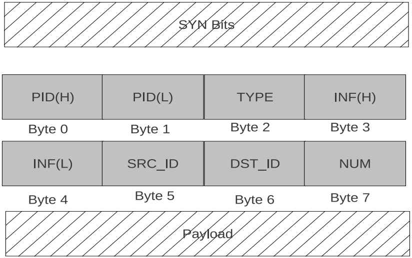


Control Flow for SRN



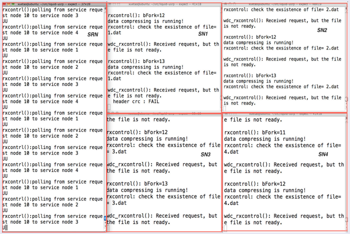
- Multiple processes
  - Communication process
  - Computing process
- Two phases
  - Service Access
    - Network initialization
    - Service Request
    - Resource Allocation
  - Service Provision
    - Computing/link status monitor
    - Data process
- SRN maitains a SN Table/ID List

Packet Structure



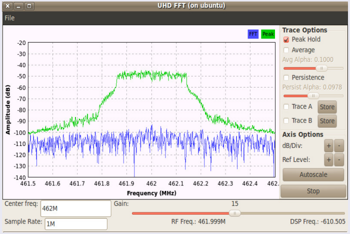
SRC_ID and DST _ID determines link. Different connections within one link are identified with INF. INF and NUM help buffer allocation and decision of Tx/Rx. Payload length may vary according to different requirements. There are four packets types: control packets, data packets and their ACKs. PID H/L: packets ID; Type: packets type;INF H/L: Information bytes; SRC_ID: source ID; DST_ID: destination ID; NUM: the number of packets

Debug Mode



- SRN with NID = 10 distributes computing workloads to four SNs NID = (1,2,3,4) through wireless links.
- Wireless links have different outage probabilities including distance and fading effects.

OTA Waveform



- RF parameters can be tuned as needed
- OFDM waveform with flexible bandwidth
- FSR band with $B = 300KHz$ for this example

Algorithm

- H.264 video encoder as the computing application with 12 frames raw data.
- One SRN, four SNs. SRN is also RN. There is no EN in this demo
- Two SN node sets
    - $\mathbb{S}_b$ with bad channels of outage probability $p_o > 0$ $\mathbb{S}_g$ with good channels with $p_o = 0$
- Two Delays
    - Communication Delay
    - Computing Delay
- Two methods are compared in term of delay performance
    - Evenly distribution: three frames for each SN
    - Considering channel difference: next slide

Algorithm

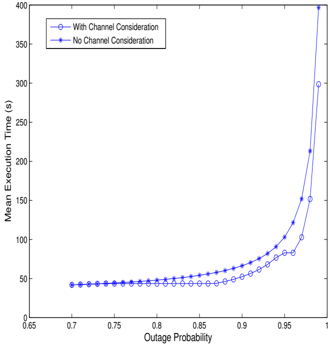
- p_o affects the retransmission times, which leads to different communication delay.
- Workload allocation method considering channel heterogeneity:

$$
\begin{aligned}
T1(f) &= \sum_{n\in\mathbb{S}_b} \frac{N}{2(1-p_o)} T_s k(n) + \tau(n,m) \\
T(f) &= \max\{T1(f), \tau(n, \frac{2M}{N} - m)\} + \sum_{n\in\mathbb{S}_g} \frac{N}{2} T_s k(n)
\end{aligned}
$$

The optimal workload allocation: $f^* = \operatorname*{argmin}_{f\in\mathbb{F}} T(f)$.

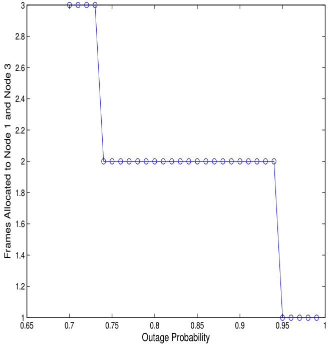
$T1(f)$: average execution time for SNs with bad channels; $T(f)$: average execution time. N: number of SNs; Ts: the time for Tx; k(n): the number of packets, k, for node n; $\tau(n,m)$: computing time for node n with the allocated frames, m. The communication delay is hidden behind computing delay for SN with bad channels.

Results



- When outage probability increase, the difference between good and bad channel increases.
- When outage probability is smaller than a threshold, considering channel difference has no improvement on delay performance.

Results

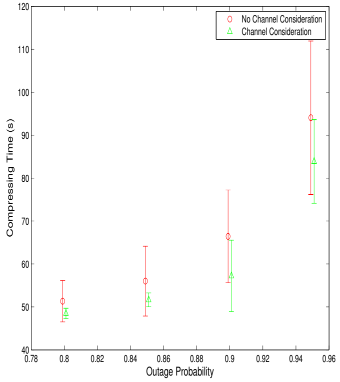


- Optimal workload allocation as a function of outage probability p_o. Staircase shape due to the minimum unit is one frame
- When channel difference becomes smaller between $\mathbb{S}_b$ and $\mathbb{S}_g$, p_o decreases, three frames = Evenly distribution

Results

- Considering channel difference will reduce both mean and variance of computing delay.
- The benefits diminishes as channel difference decreases.
- Computing delay dominates the results when channel difference is small.
- The variance of communication time can be hidden behind computing delay.

Experiment Data



Results

Table: Experiment Data for Execution Time in Seconds

	Reference Method		Proposed Method	
p_o	Mean	Variance	Mean	Variance
0.8	51.3	23.2	48.5	1.5
0.85	56.0	66.1	51.6	2.6
0.9	66.4	116.8	57.2	69.2
0.95	94.0	319.6	83.9	94.7

Summary

- An implementation of WDCNs based on SDR and CORNET is shown.
- Workload allocation ideas are demonstrated based on the dedicated software.
- Over-the-air data and performance analysis show the benefits of considering channel difference.
- The dedicated software is easy to be expanded to investigate other performance of WDCNs.

Thank you!