Understanding and Taming Wireless Interference in Homes: An SDN-centric Approach

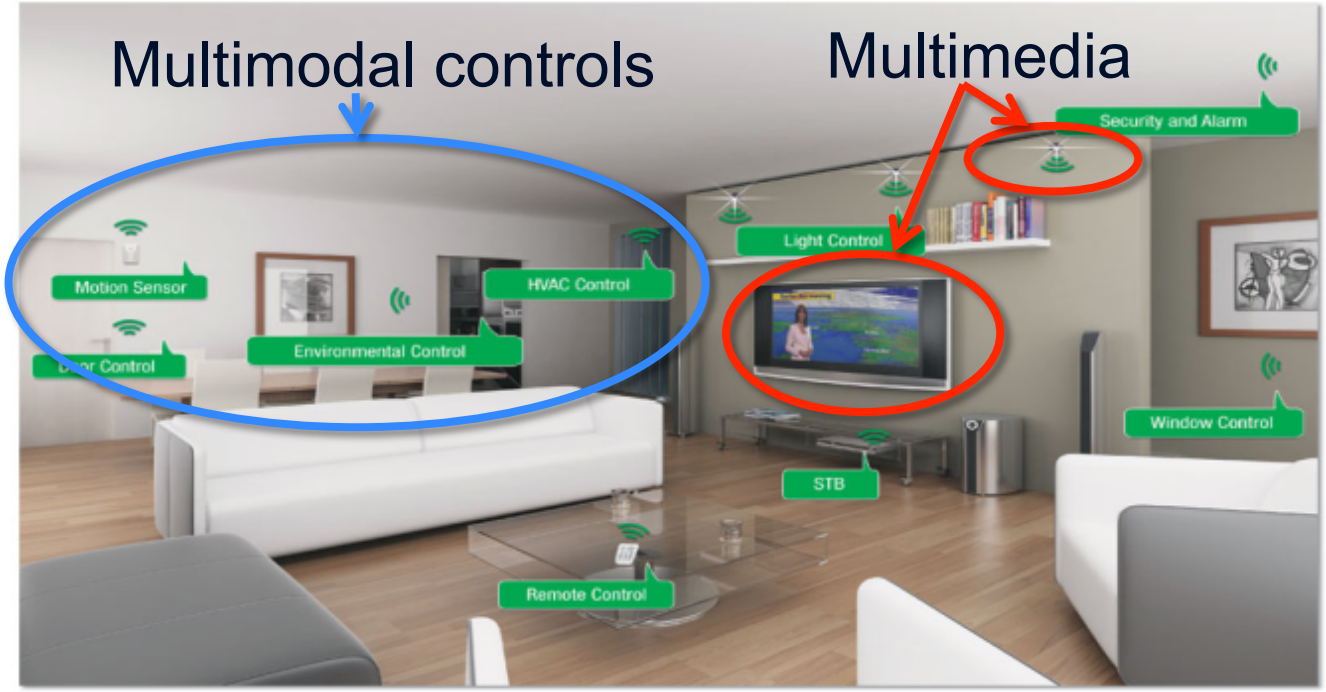
Suman Banerjee
(suman@cs.wisc.edu)

Wireless @ Home

Wireless is the dominant technology in homes for accessing multimedia services

- ISPs have built strong experience in managing the wired connection into the home
- Wireless link has typically been left unmanaged

Wireless management is critical for user experience



- TV
- Tablets
- Storage
- Security
- Sensors

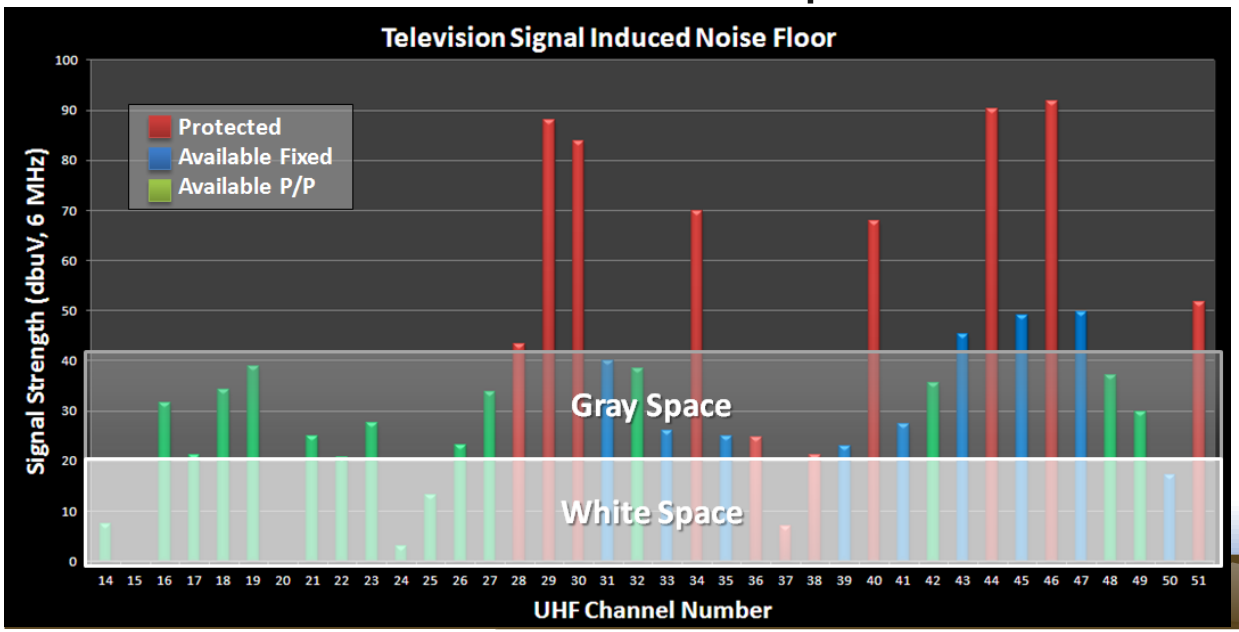
Unlicensed spectrum shared in an uncoordinated manner

Wireless@Home Technology Landscape

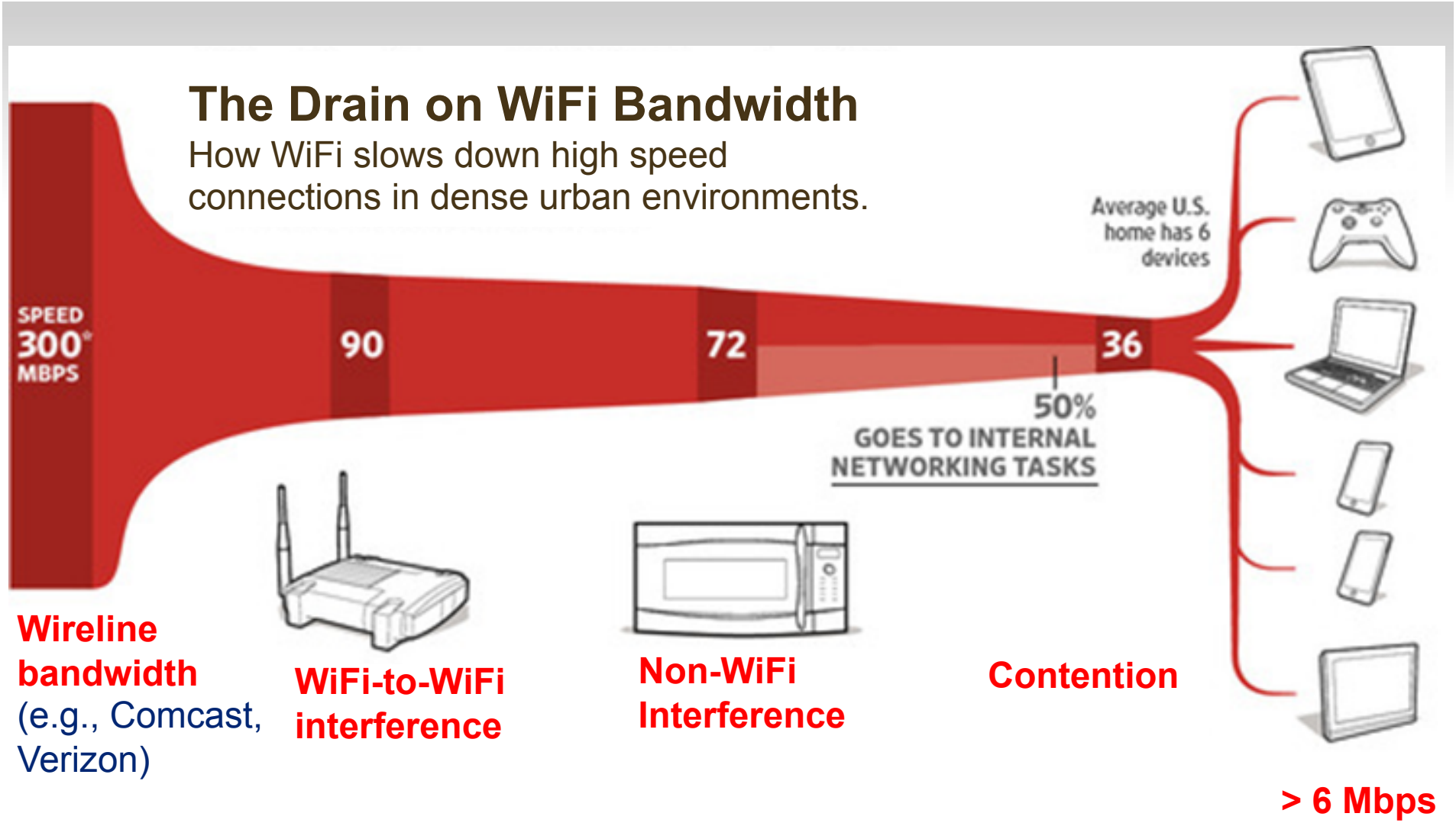
Wi Fi™

Zigbee

TV Whitespace



X10

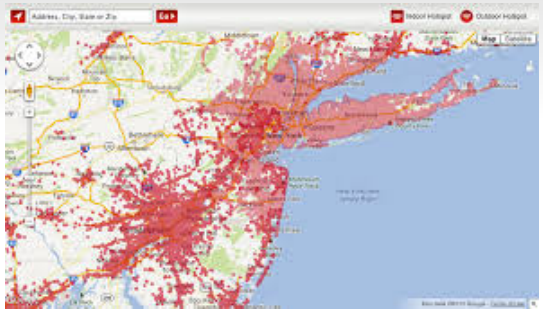


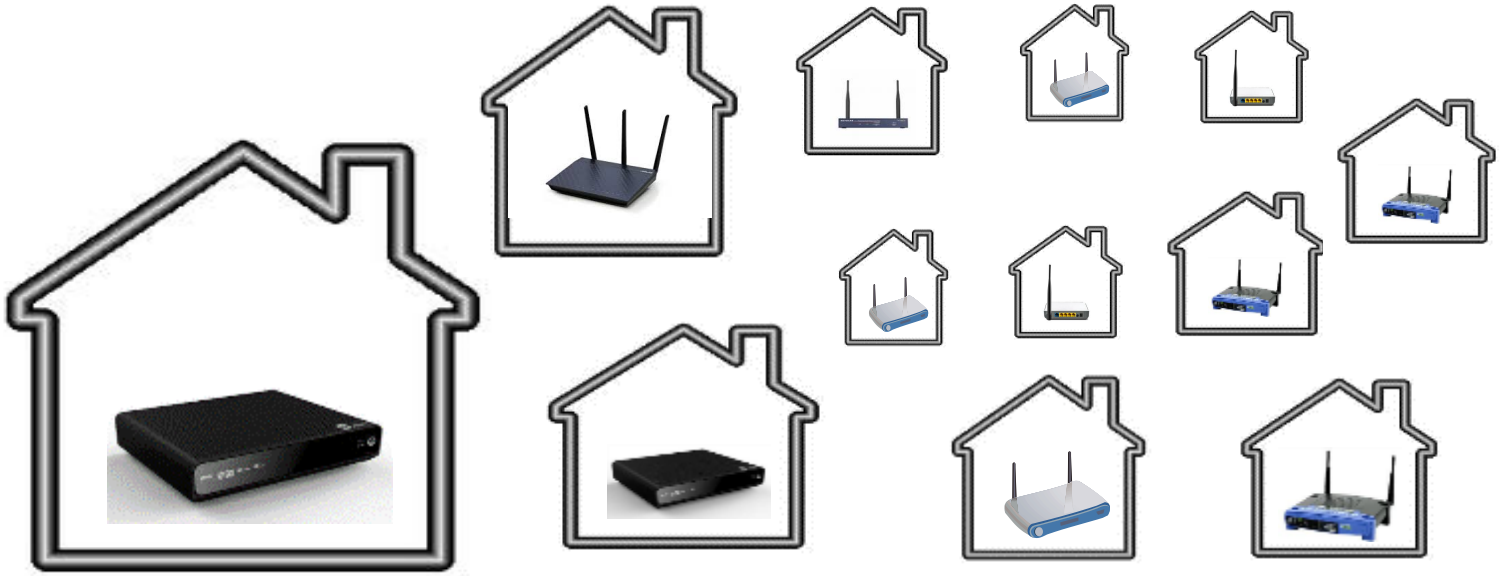
Source: WSJ, Comcast, 5Nines

Goal 1: Understanding interference

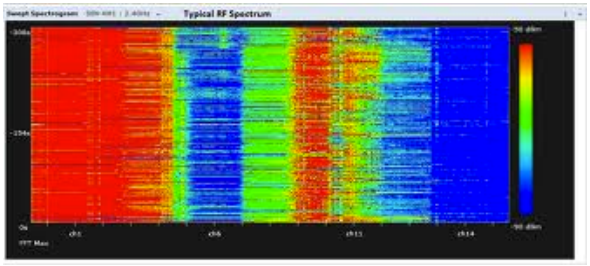
“Blood test” for wireless networks

Goal 2: Taming interference





Toolbox



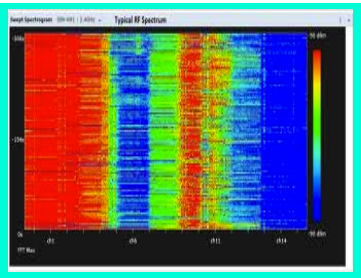
RF analytics

SDN

All implemented in commodity WiFi software and hardware

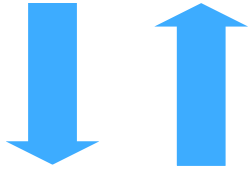
Overall architecture

Cloud-based
Wireless
Management



RF analytics

SDN

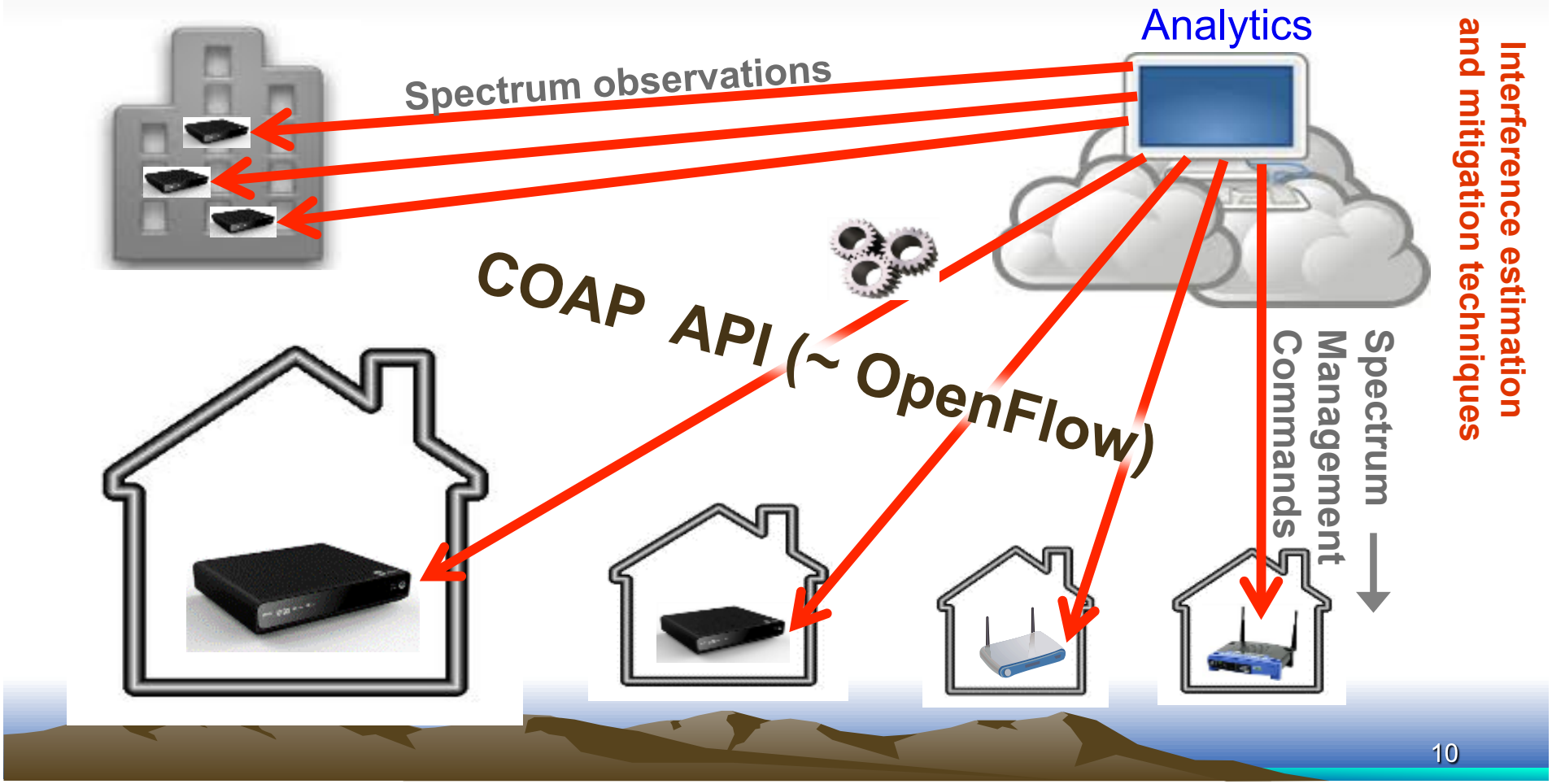


Commodity AP Hardware

(Broadcom, Qualcomm, Marvell …)

Cloud-based RF Management

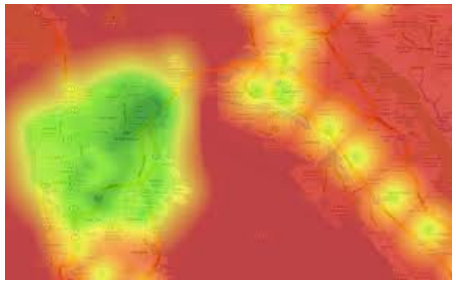
COAP: Coordination framework for Open Access Points



Performance Optimization & Analytics

- ✓ Interference Detecting & Quantification
  - WiFi, Bluetooth, Analog phones, Microwaves, Game controllers

- ✓ Coverage Maps
  - RSSI, CINR at (x,y,z) coordinates



- ✓ Localization of Interferers
  - "Your WiFi box is too close to microwave oven. Please move it farther preferably…."
  - "There is a cordless phone close by. Please move the cordless phone farther away from your tablet."

11

Performance Optimization & Analytics

- ✓ WiFi interference mitigation
  - Frequency, Power, Cross-Layer Optimizations
  - Machine Learning & Statistical Analysis

- ✓ Network Performance Matrices
  - Quality of Experience
  - RF Heat Maps …

- ✓ Customer Support Assistance
  - Go back in time: see interferers & AP performance
  - See neighboring APs and details of these APs
  - Observe Centaur driven updates and their results

Understanding Interference

Our solution

Real-time Interference estimators

WiFi to WiFi Interference

COLLIE (Collision Inferencing Engine)

PIE (Passive Interference Estimator)

Non-WiFi to WiFi Interference

(use WiFi-only hardware!)

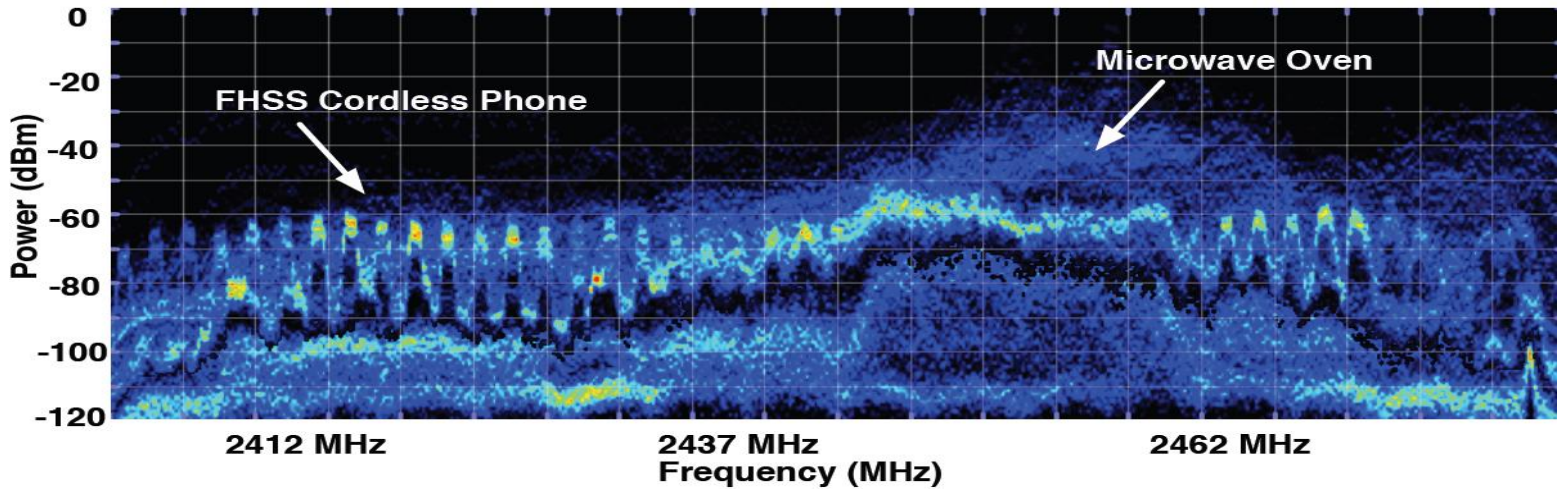
Airshark

1. detect non-WiFi devices

WiFiNet

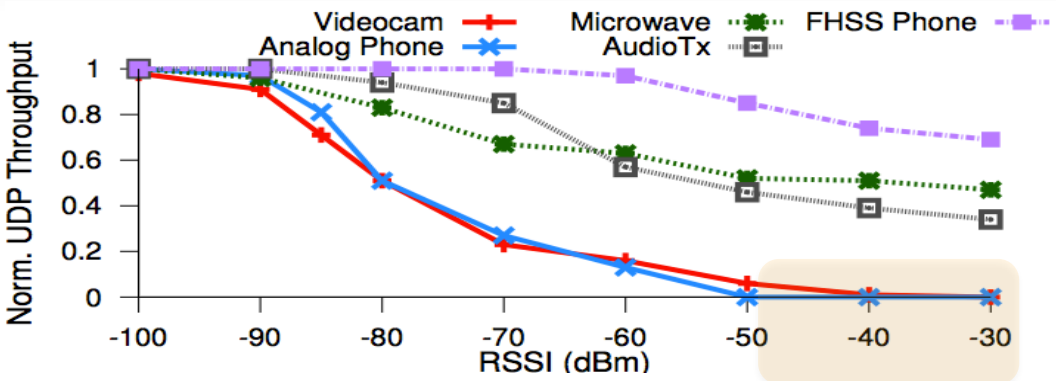
1. quantify interference impact
2. pin point device location

Spectrum at a university cafe



High powered non-WiFi devices share the spectrum with WiFi devices

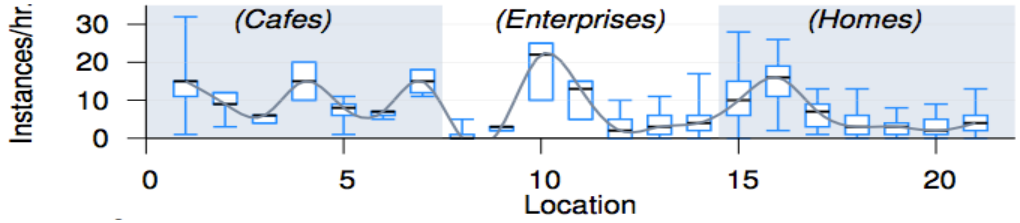
Is non-WiFi interference a real problem?



How severe is the impact?

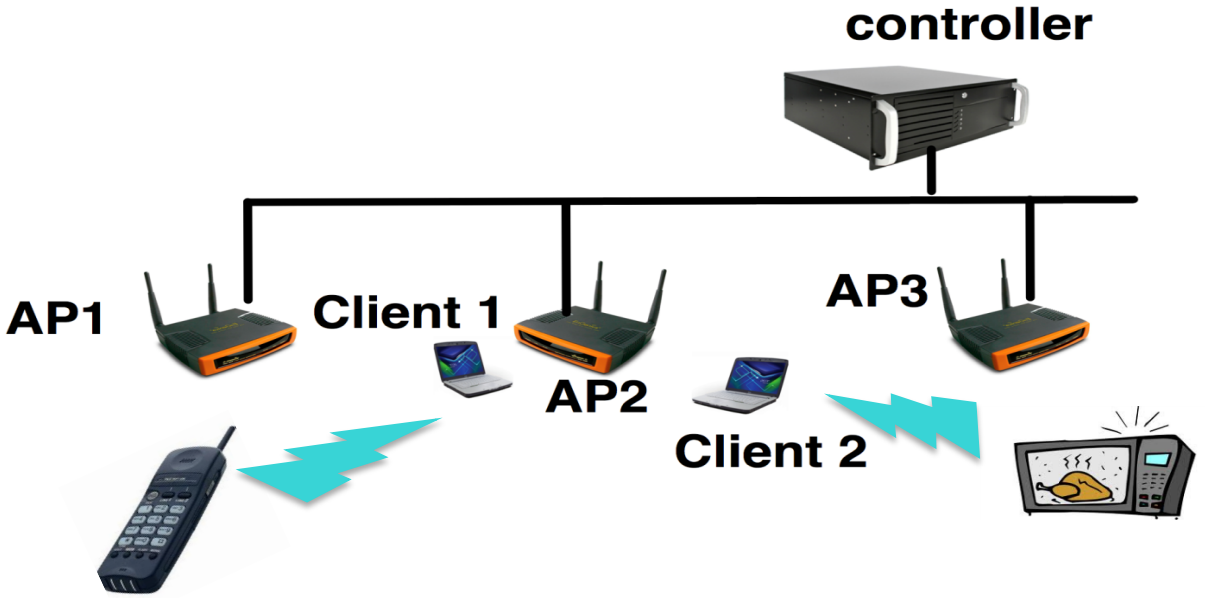
More than 50% throughput drop and, in some cases, throughput drops to zero!

How prevalent are the devices?



Across locations, many devices frequently appeared with a high signal strength

Interference detection goals



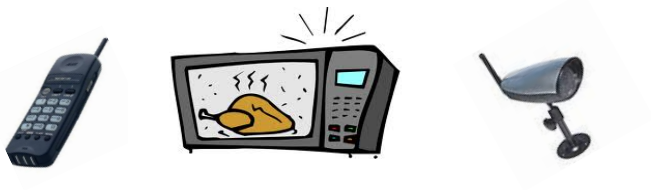
Key questions:

- Are there any non-WiFi devices in the medium?
- Which non-WiFi devices caused interference?
- How do we locate these non-WiFi devices?

What is hard?

- Use *existing* WiFi interfaces to detect and quantify

  - **No** dedicated spectrum sensor
  - Who is the interferer?

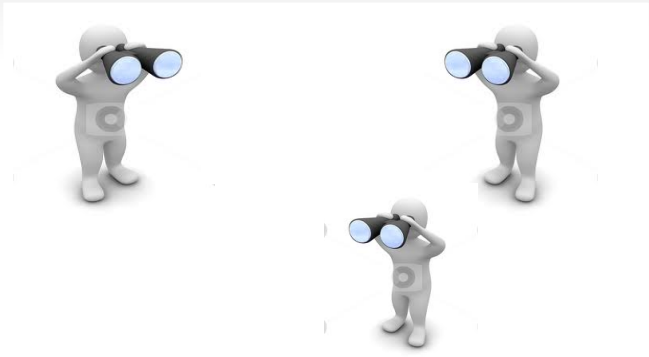


  - Can I tell multiple identical devices apart?
    - Needed for impact quantification

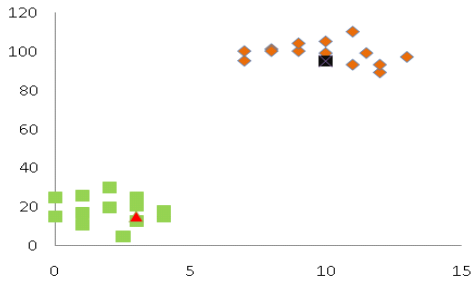
  - How to localize?

Solution Approach

- Exploit distributed observers



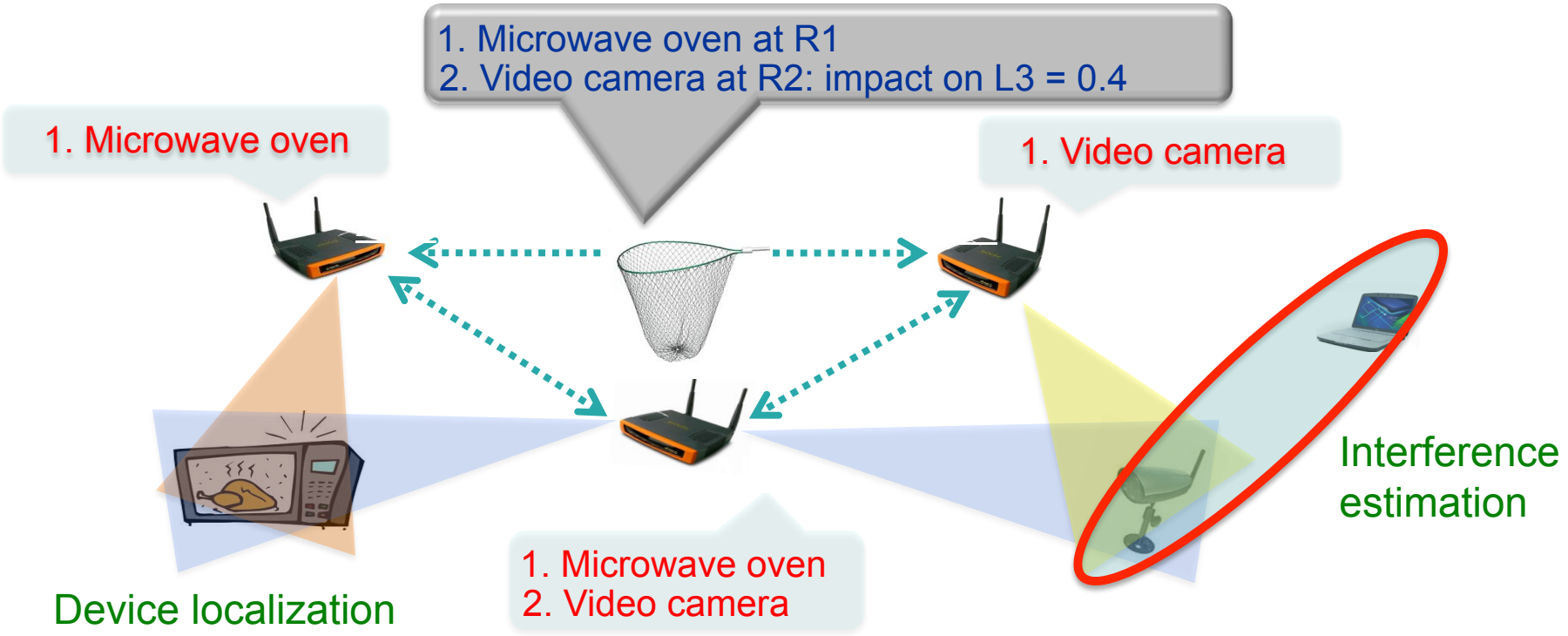
- Timing information and statistical techniques



- Learn and adapt

Solution overview

- **1. AirShark:** Non-WiFi device detection
- **2. WifiNet:** Quantify interference and localize

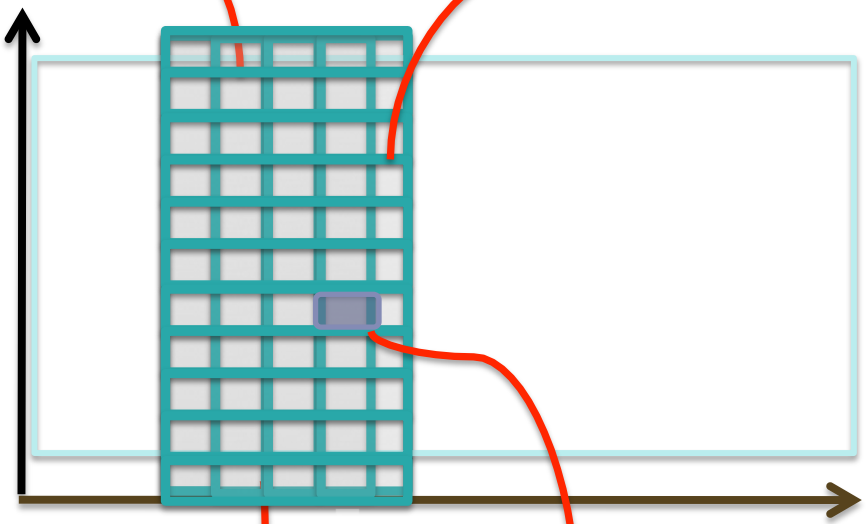


Using a coarse-grained WiFi lens

Coarse-grained freq. resolution

312 kHz
Sub-carriers

Coarse-grained time resolution



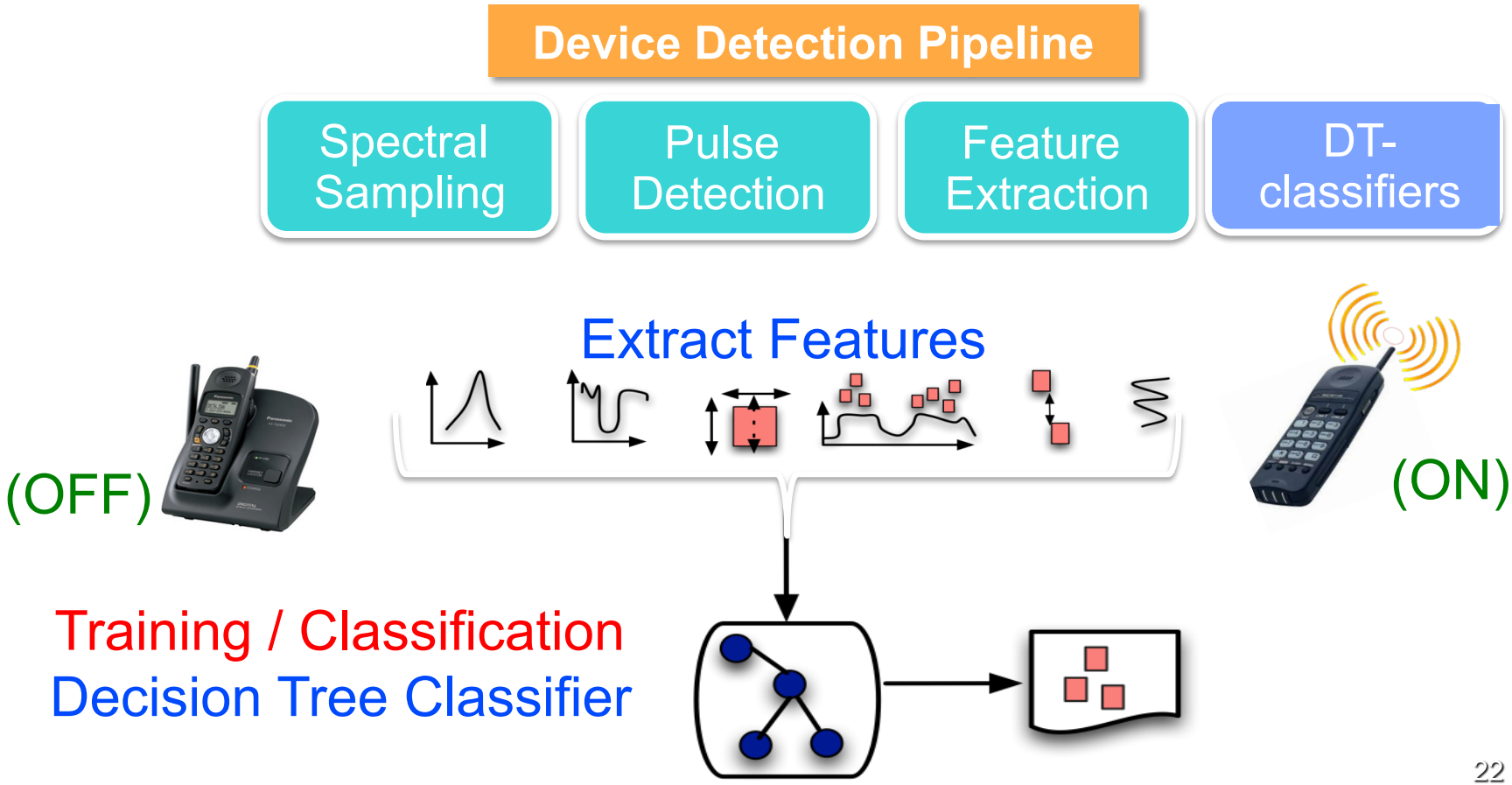
$O(10^3)$ to $O(10^4)$ samples/sec

22 MHz

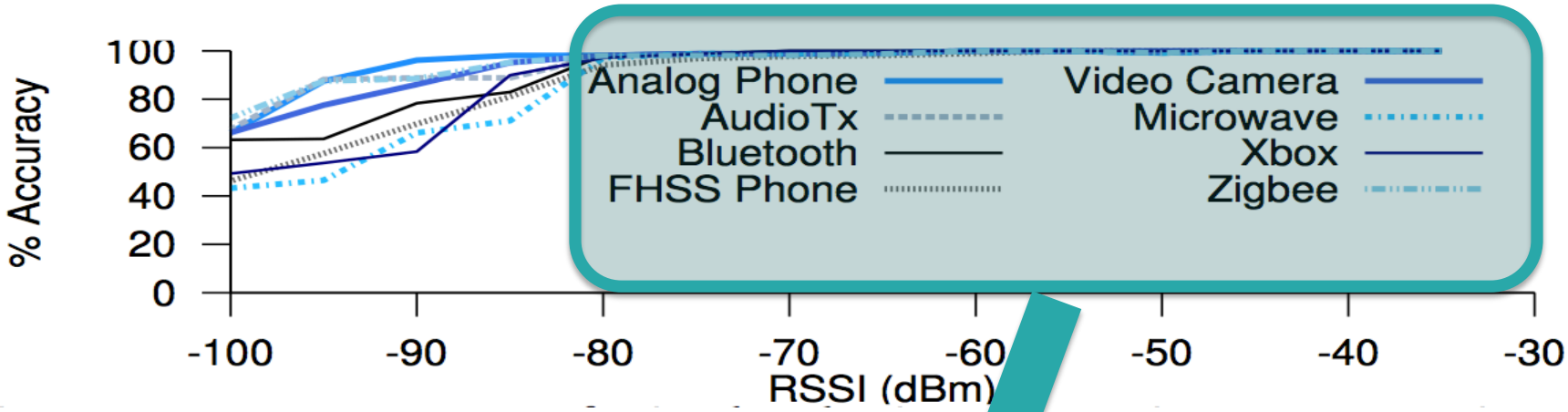
RSSI/power per sub-carrier

Limited spectrum view

Airshark: *how it works*

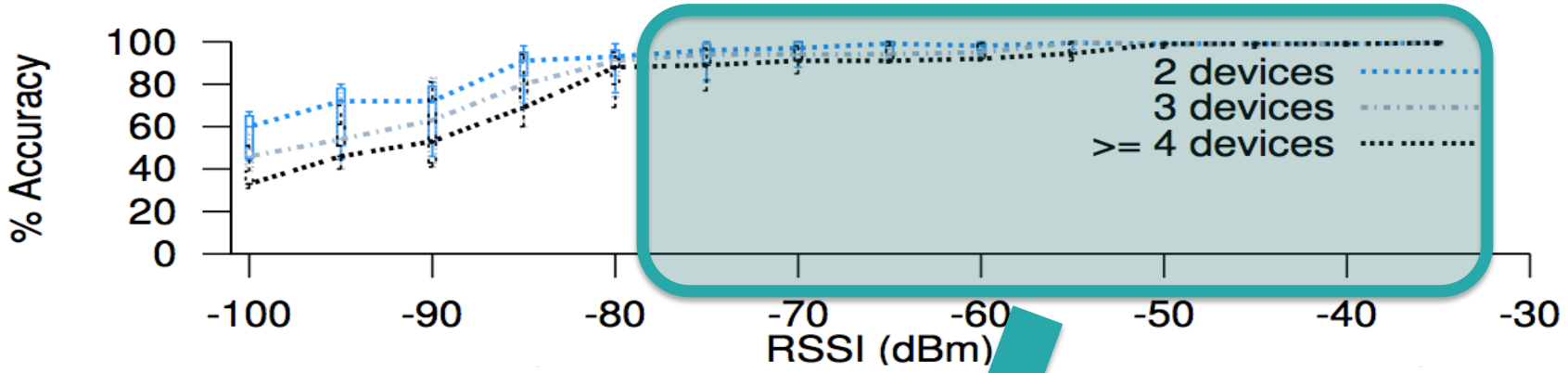


Detection Accuracy: Single device



> 98% accuracy at signal strengths >= -80 dBm

Detection Accuracy: Multiple devices



> 91% accuracy at signal strengths >= -80 dBm

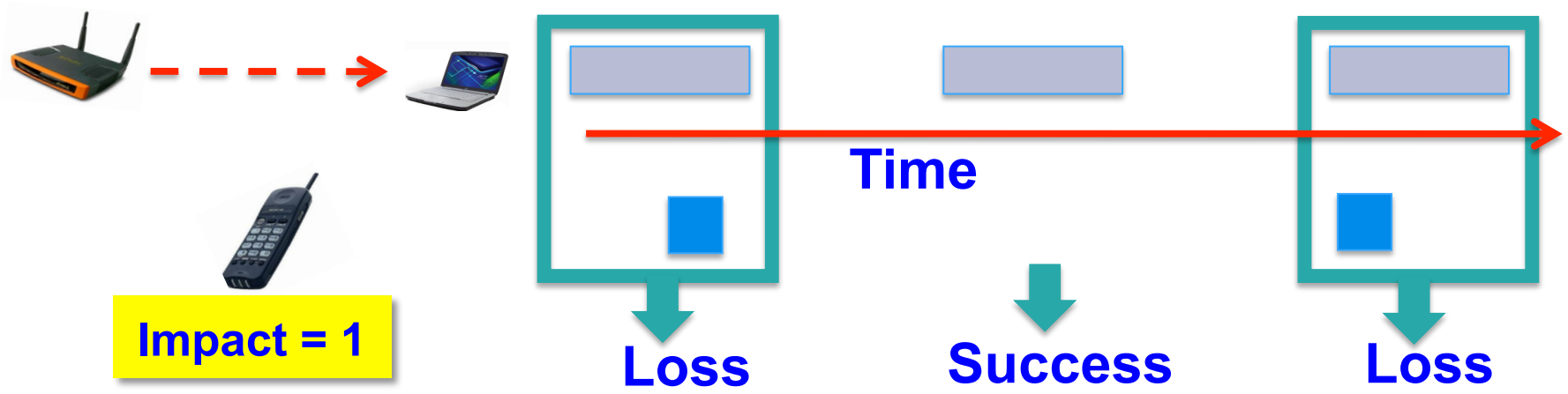
Many trials

Including integrating with a commercial off-the-shelf AP platform

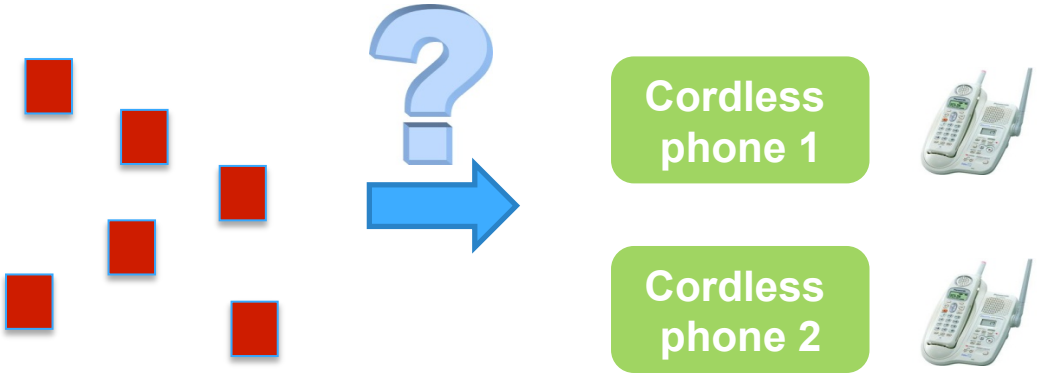
Demo: Airshark

WiFiNet: Interference estimation concept

- Quantify the “impact” of each device on each WiFi link
  - Identify the transmission overlaps between WiFi frames and non-WiFi pulses
  - Correlate frame losses and transmission overlaps



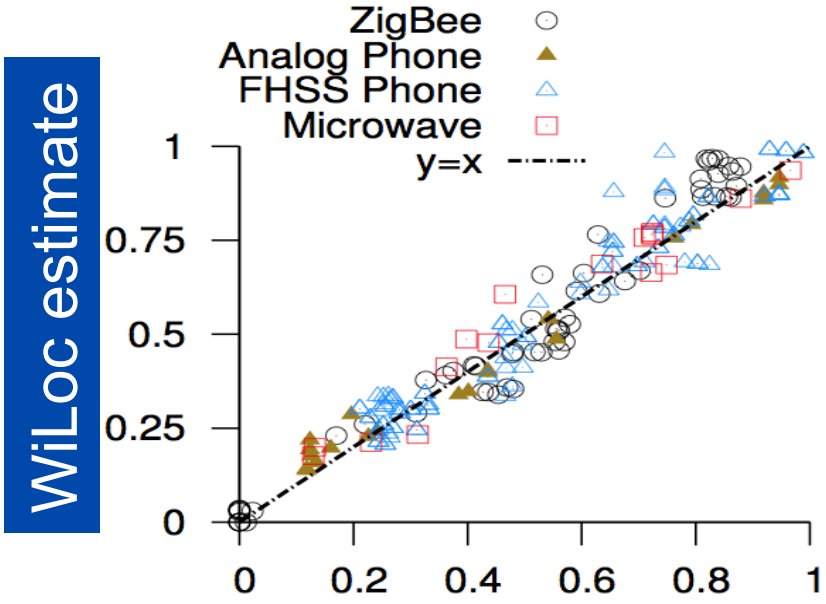
WiFiNet: Additional challenges



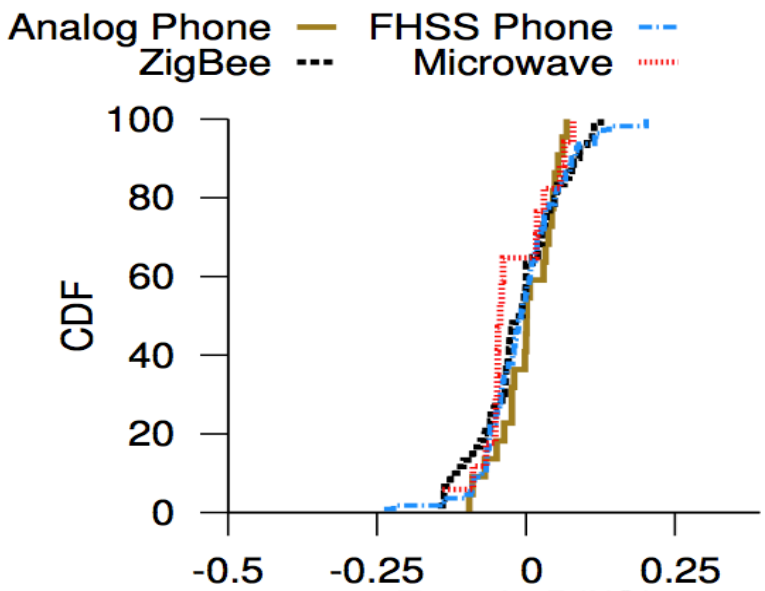
What about multiple devices of same type?
Which RF activity belongs to which device?

Also have to them discern interference impact of each device type

Interference Estimation Results: *Single* non-WiFi interferer



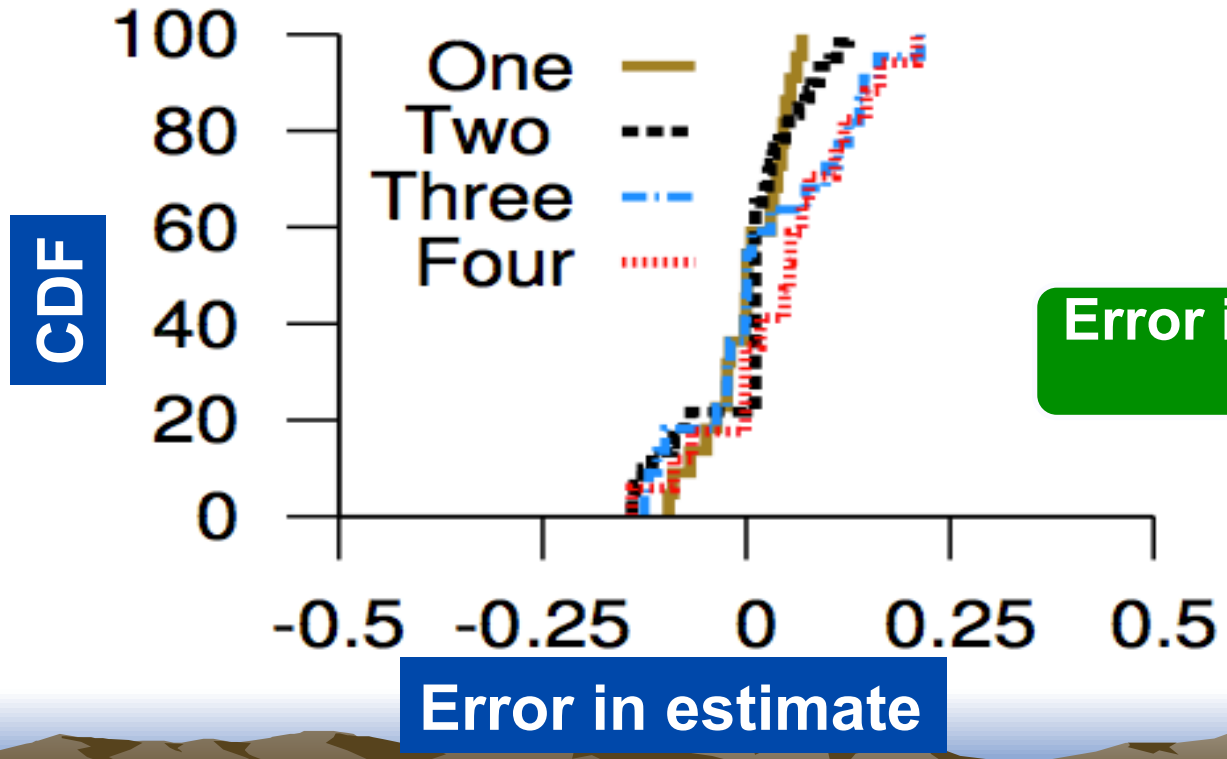
Ground truth



Error in estimate

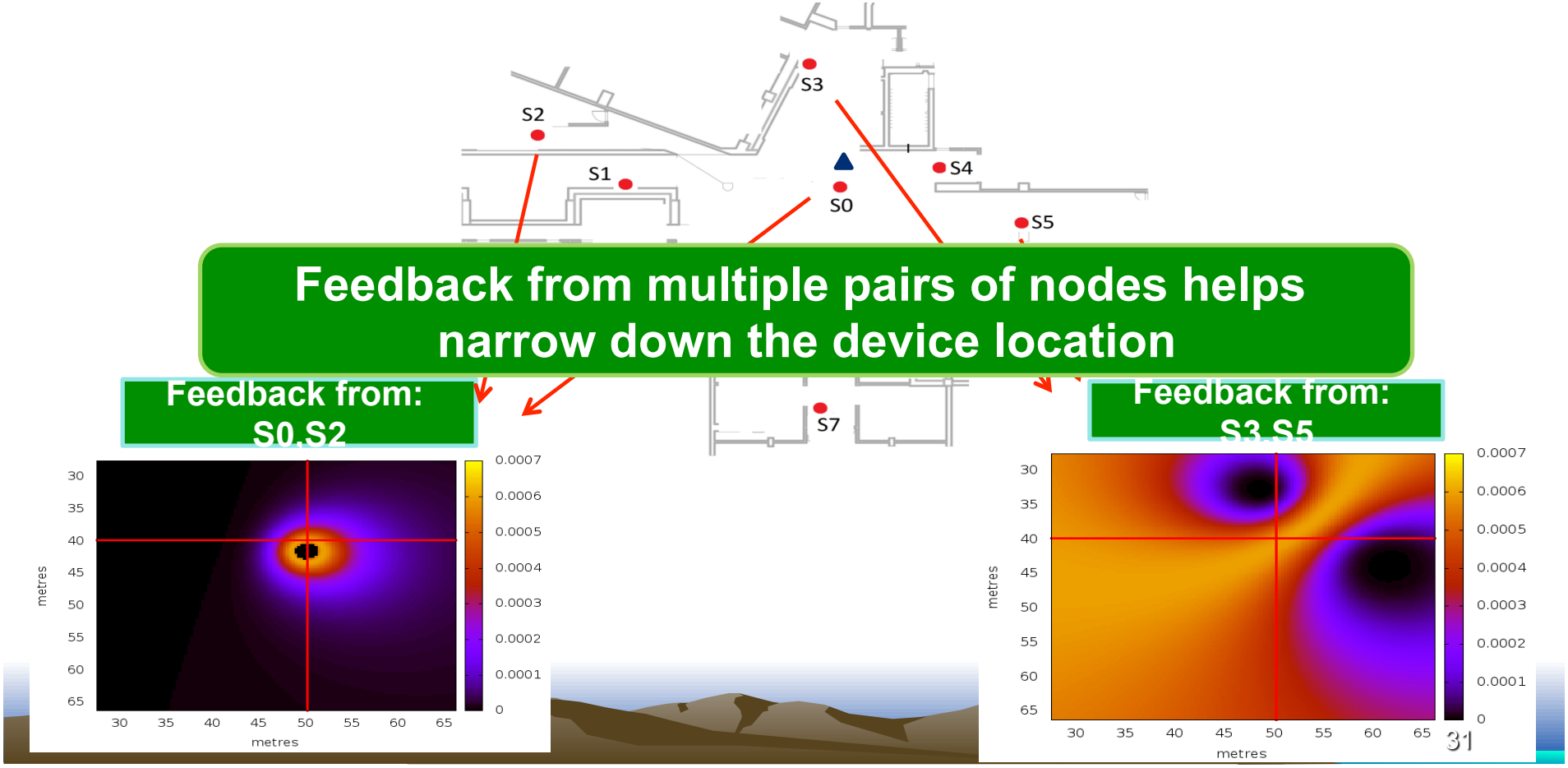
Error is within 10% for 95% cases

Interference Estimation Results:
Multiple non-WiFi interferers

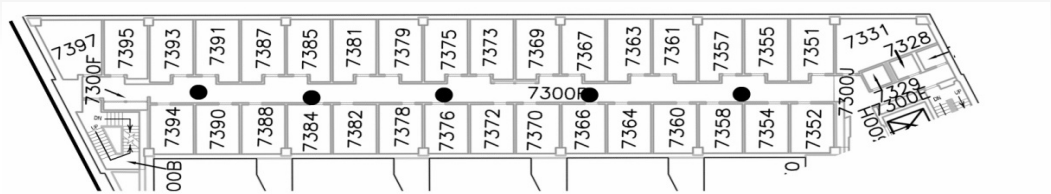


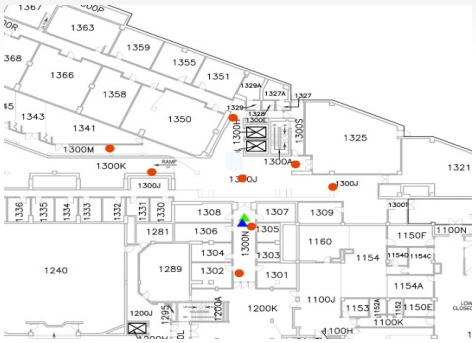
Error is <15% for >85% of the cases

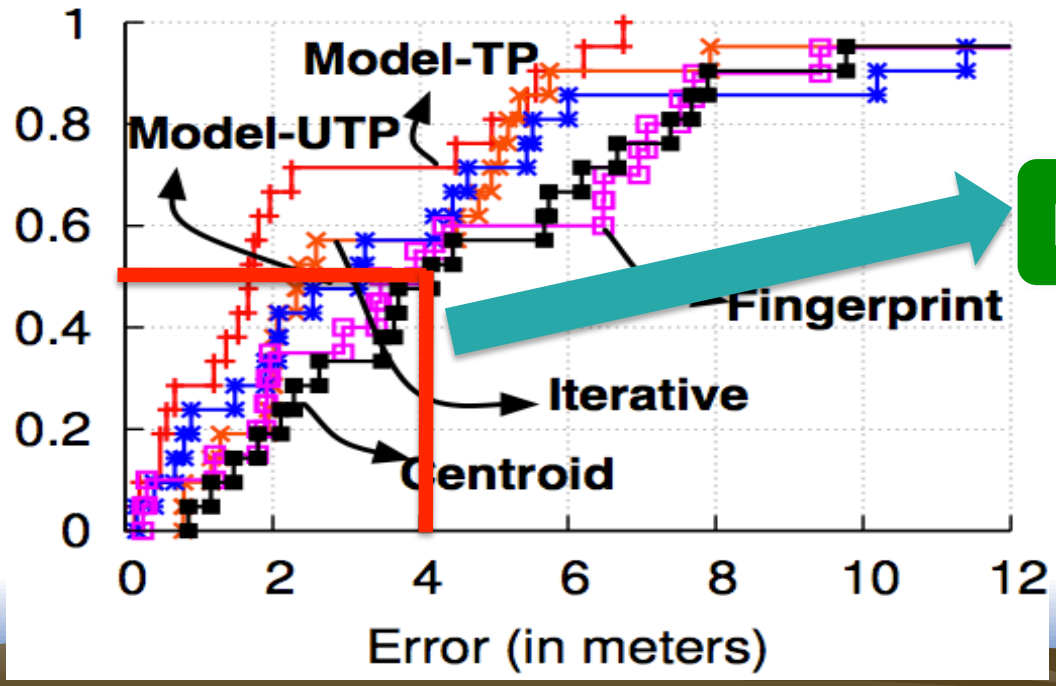
Model based localization



Device Localization Results







Median error <= 4 meters

Taming Interference

Today's world of **Home** WiFi APs

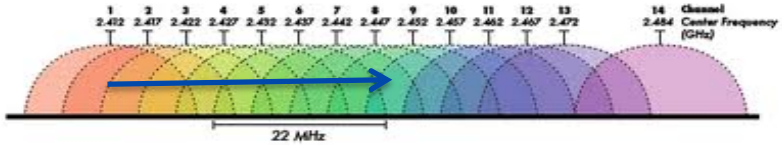
Each AP has to fend for itself

The SDN approach

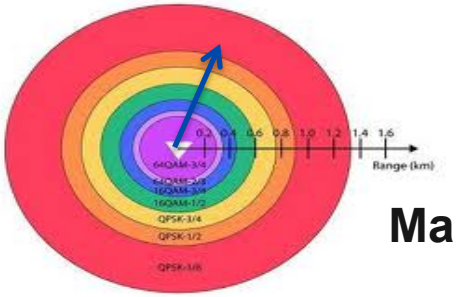
(brings enterprise-style diligence to home APs and much more)

Controller

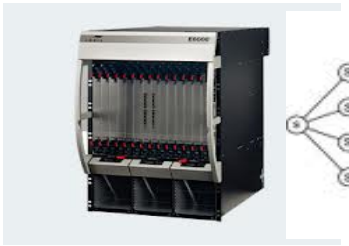
Interference Mitigation Strategies

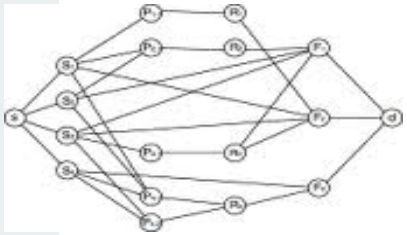


Coordinated channel changes



Manage Tx Power



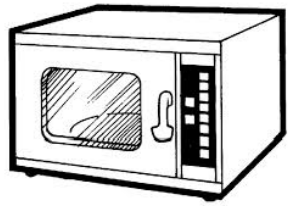


Epoch-based scheduling

Managed: Channel Assignments, Power Control,
Cross-layer Scheduling, Load Balancing & more

Learning-based configuration

- Microwave comes on at 7pm and stays on for about 30 minutes



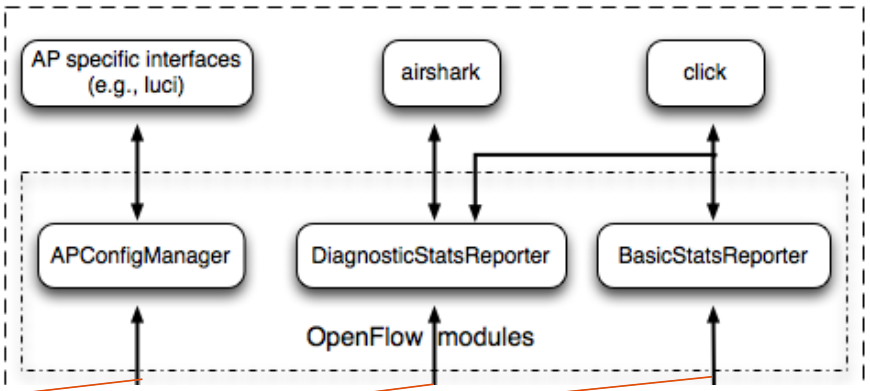
- Neighbor starts a Netflix flow late evening

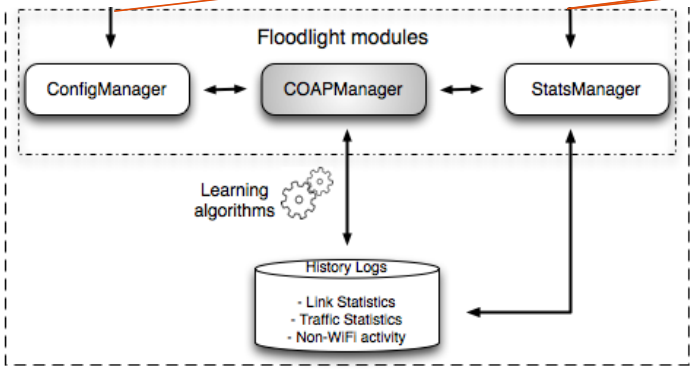
Make proactive configuration decisions based on learnt context

COAP reference implementation

COAP:
Coordination framework for Open Access Points

Home AP





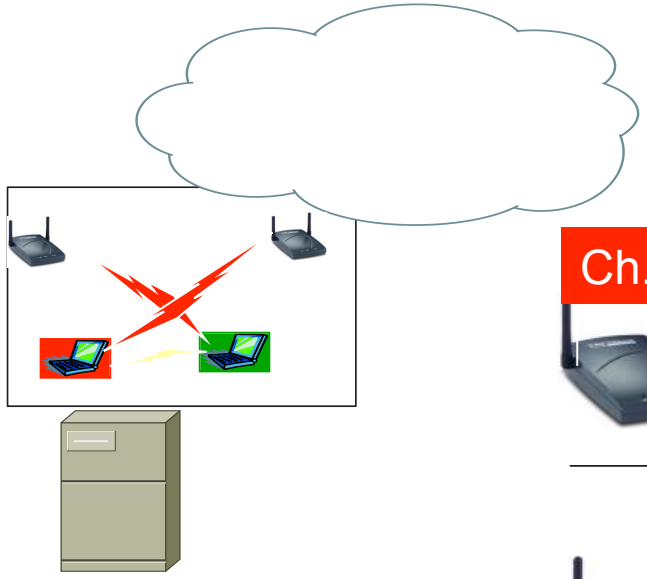
Controller

Implemented using a Floodlight controller, OpenFlow, and OpenWRT APs

COAP API components

Function
SetParameters (channel, power)
SetAirtimeAccess (slotDur, txBitmap)
GetNeighborInfo()
GetAirtimeUtilization(epoch)
GetClientInfo()
GetLocalLinkStats()
GetTrafficInfo()
GetNonWiFiDevices()
GetPacketSummaries()

A simple example



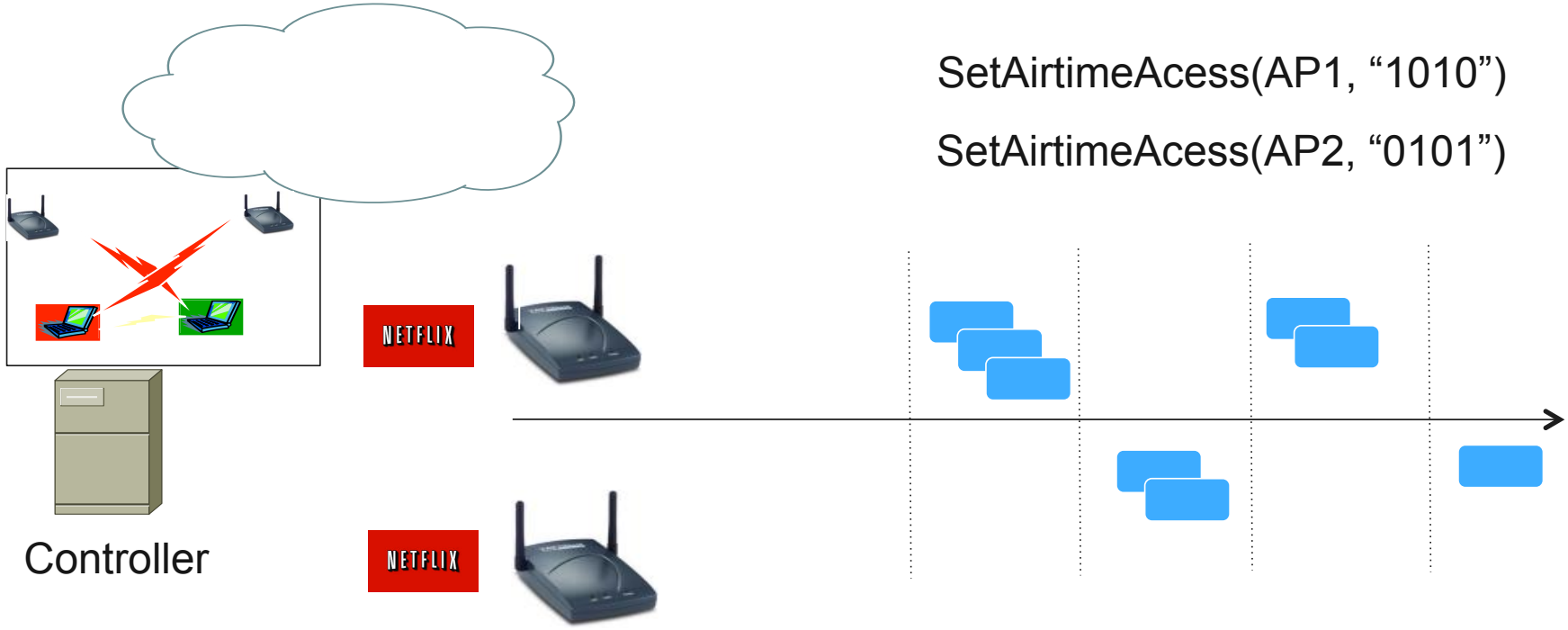
Controller

SetParameters(AP1, Channel, 1)

SetParameters(AP2, Channel, 6)

A simple example

Handling a hidden terminal scenario



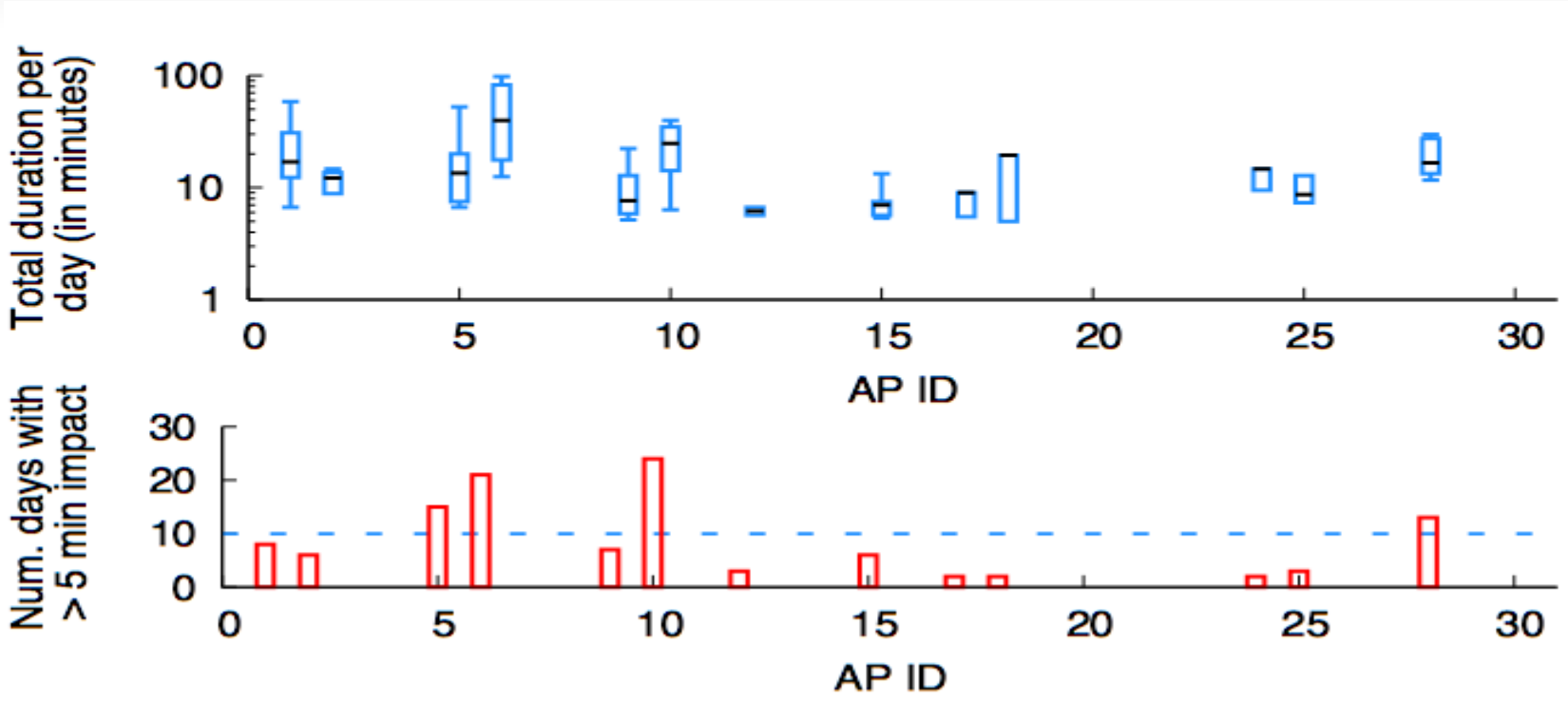
Field Trials

- OpenWRT based APs
  - ALIX 2d2 platform: (500 MHz AMD Geocode CPU, 256 DDR RAM, flash storage)
- 30+ APs deployed in homes & apartment complexes for 2+ years
- COAP Cloud controller hosted in off-the-shelf Linux server

Understanding poor performance

Indicators				Bldg 1		Bldg 2	
A ↑	S ↓	L ↑	R ↓	V Poor	Poor	V Poor	Poor
Y	X	X	X	0	18.4	0	1
X	X	Y	X	24.2	**49.5**	25.2	**78.1**
Y	X	Y	X	**61.8**	26.7	2.1	1.4
X	Y	Y	X	2.3	1.1	20	15.8
X	Y	Y	Y	9.4	0	**51.6**	3.4
Others				2.3	4.3	1.1	1.3

Contention experience

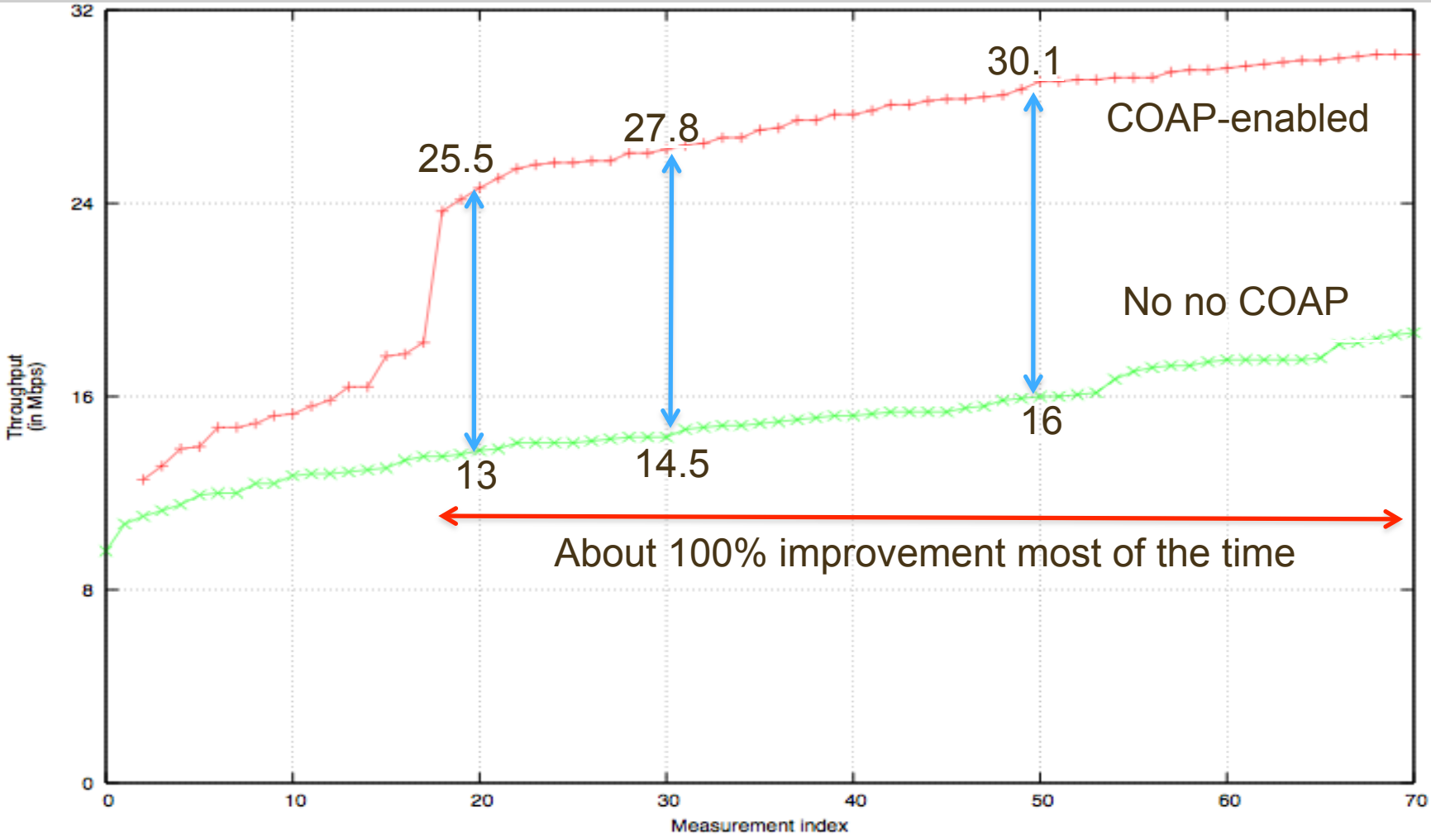


(In a 40 day period)

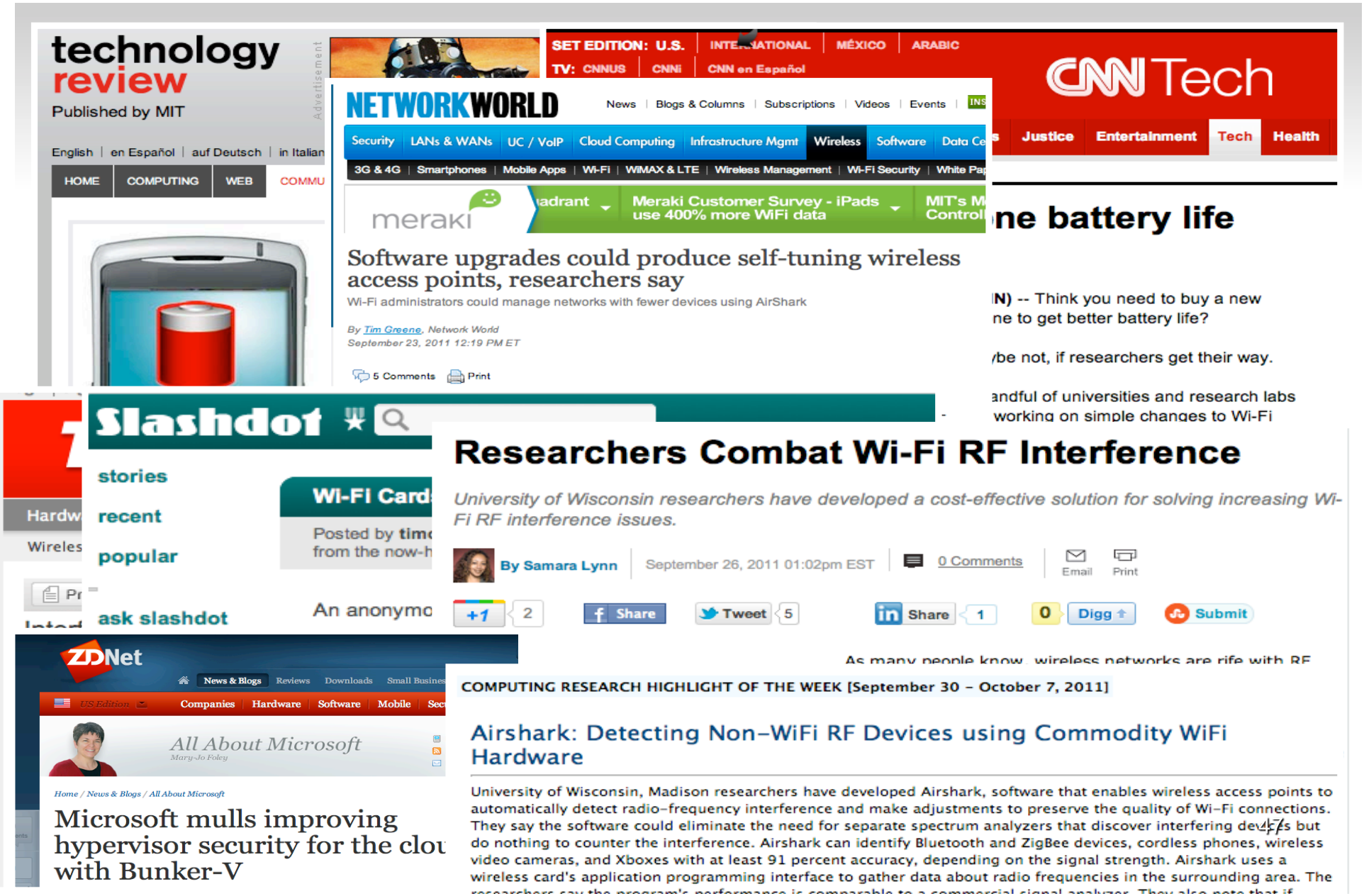
Solution advantage

- Compared solution in one of our apartment buildings

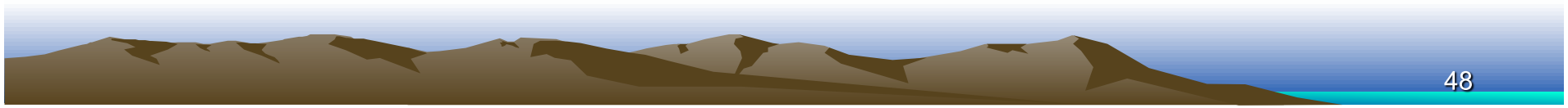
- Approach
  - Day 1: COAP completely disabled
  - Day 2: COAP managed
  - Alternated for nearly two weeks



Data is sorted for the two cases for easier viewing


technology review
Published by MIT
English | en Español | auf Deutsch | in Italian
HOME
COMPUTING
WEB
COMMU
SET EDITION: U.S.
INTERNATIONAL
MÉXICO
ARABIC
TV: CNNUS
CNNi
CNN en Español
CNN Tech
Justice
Entertainment
Tech
Health
NETWORKWORLD
News | Blogs & Columns | Subscriptions | Videos | Events
Security
LANs & WANs
UC / VoIP
Cloud Computing
Infrastructure Mgmt
Wireless
Software
3G & 4G | Smartphones | Mobile Apps | Wi-Fi | WiMAX & LTE | Wireless Management | Wi-Fi Security
meraki
Meraki Customer Survey - iPads use 400% more WiFi data
Software upgrades could produce self-tuning wireless access points, researchers say
Wi-Fi administrators could manage networks with fewer devices using AirShark
By Tim Greene, Network World
September 23, 2011 12:19 PM ET
5 Comments
Print
ne battery life
N) -- Think you need to buy a new
ne to get better battery life?
be not, if researchers get their way.
andful of universities and research labs
working on simple changes to Wi-Fi
Slashdot
stories
recent
popular
ask slashdot
Researchers Combat Wi-Fi RF Interference
University of Wisconsin researchers have developed a cost-effective solution for solving increasing Wi-Fi RF interference issues.
By Samara Lynn
September 26, 2011 01:02pm EST
0 Comments
Email
Print
+1
2
Share
Tweet
5
Share
1
0
Digg
Submit
ZDNet
News & Blogs
Reviews
Downloads
US Edition
Companies
Hardware
Software
Mobile
All About Microsoft
Mary-Jo Foley
Home / News & Blogs / All About Microsoft
Microsoft mulls improving hypervisor security for the clou with Bunker-V
COMPUTING RESEARCH HIGHLIGHT OF THE WEEK [September 30 – October 7, 2011]
Airshark: Detecting Non-WiFi RF Devices using Commodity WiFi Hardware
University of Wisconsin, Madison researchers have developed Airshark, software that enables wireless access points to automatically detect radio-frequency interference and make adjustments to preserve the quality of Wi-Fi connections. They say the software could eliminate the need for separate spectrum analyzers that discover interfering devices but do nothing to counter the interference. Airshark can identify Bluetooth and ZigBee devices, cordless phones, wireless video cameras, and Xboxes with at least 91 percent accuracy, depending on the signal strength. Airshark uses a wireless card's application programming interface to gather data about radio frequencies in the surrounding area. The

One more thing



Looking for AP platform partners to deploy COAP

Go to
research.cs.wisc.edu/wings/projects/coap

suman@cs.wisc.edu

Students: Ashish Patro, Dale Willis, Arkodeb Dasgupta, Prakhar Panwaria,
R Sivasubramanian