Reliability and Maintainability of IoT systems

Prof. Tajana Simunic Rosing
UCSD
Internet Evolution

Starts with a network and evolves into everything that can be connected with a network

Source: Perera et al. 2014
Sensor Networks (SNs)

Consist of a number of sensor nodes communicating in a wireless multi-hop fashion

Source: Perera et al. 2014
Sensor Networks (SNs)

• SNs have been designed, developed, and used for specific application purposes
  – Environmental monitoring, agriculture, medical care, event detection etc.

• SNs use middleware for:
  – Abstraction support, data fusion, resource constraints, dynamic topology, adaptability, scalability, security, and QoS support, etc.
HPWREN wireless connectivity covers 20k sq. mile area with numerous sensors

- Motion detect cameras
- Wildfire tracking cams
  Cal Fire Dept.
- Environmental sensors & cams
  Wolf howls at the CA Wolf Center
- Acoustic sensors
  Wolf howls at the CA Wolf Center

HW Braun, F. Vernon, T. S. Rosing, UCSD
HPWREN: 16 Years of Experience

• Began in 1999 due to the need to provide reliable connectivity for seismic sensors; started by Frank Vernon, SIO & Hans W. Braun, SDSC, UCSD.
  – Solar cell powers a battery, seismic sensors and computation done underground; large scale antennas
• It grew to 20,000 square miles, from San Clemente Island in the Pacific Ocean (72 miles of wireless link), to the inland valleys and onto the high mountains, reaching an altitude of more than 8,700 feet, across the desert, reaching sites close to the Arizona border
• Original funding was from NSF (2000, 2004, 2009 grants), but public safety sector and other agencies got involved since:
  – San Diego Gas & Electric
  – NASA, US Navy
  – California Dept. for forestry and fire protection, San Diego City Fire and Life Safety Departments, Ramona Fire Department, Gillespie Helitack Base
  – San Diego County’s Sheriff’s Department, San Diego City Police Department
  – Mt Laguna & Palomar observatories
  – Three Native American reservations
  – US National Park Service, NOAA, Santa Margarita Reserve, California Wolf Center, SIO, UCSB Earth Research Institute, Seismic Warning Systems, UNAVCO
Mount Laguna Sensors

- Temperature
- Relative humidity
- Fuel moisture
- Fuel temperature
- Data logger
- Barometric pressure
- 3D ultrasonic anemometer
- Pan-tilt-zoom camera
- Solar radiation
- Tipping rainbucket
- Support equipment
Wind Gusts on Mt. Laguna
Interactive Spectrum Analysis of Real-Time Seismic Data

(magnitude 1.6, near Banning, Riverside County, CA)

~59 miles

~34 miles

~75 miles

~73 miles
Santa Margarita Ecological Reserve

75 Cluster Heads connected via WLAN
U.S. Navy Deep Submergence Unit

CitiSense Deployment: Wearables and Environmental Sensors

- Sensors and phones given to commuters using various transportation means throughout the greater San Diego area
  - Results found “urban valleys” where buildings trapped pollution
  - Major routes reported contrastingly high/low AQI for the same location
  - Ease of deployment and in-network adaptation is key: learn from context!
Gillespie Helitack Base: Monitoring Wild Fires by Helicopters
Eagle Fire HPWREN connection, May 2004

HPWREN access at SDSU/SMER

Preparation for relay battery transport at the ICP site

Local SAM connection at the ICP

Eagle Fire as seen from HWY79 on May 3rd, 2004
Volcan Fire HPWREN connection, September 2005

Incident Command Post site

Volcan relay site
Summary of Learnings from Firefighting Over the Last Decade

- **Rapid response is key:**
  - It started with enabling connectivity for Incident Command Post deployed for large wildfires, and connectivity to fire stations, camps and air bases
  - Real-time event detection based on weather sensors sent automated pager and email alarms to public safety personnel, alerting them in real time of Santa Ana conditions
    - During 2007 fire HPWREN provided data to both fire fighters and public on the current conditions; enabling much more accurate and scalable troop deployment

- **Key problem:** we need scalable, adaptable, manageable infrastructure
  - The question is not if failures will happen, but how proactive the infrastructure will be at avoiding them and solving them -> big issue for the IoT!
The Internet of Things allows people and things to be connected Anytime, Anyplace, with Anything and Anyone, ideally using Any path/network and Any service.
Characteristics of IoT

1. Intelligence
   - Knowledge extraction from the generated data
2. Heterogeneous Architecture
   - Supports many different types of devices and connectivity
3. Complexity
   - A diverse set of dynamically changing objects
4. Huge Size
   - Scalability
5. Timing
   - Billions of parallel and simultaneous events
6. Space
   - Localization
7. Everything-as-a-service
   - Consuming resources as a service
### Top Industries in IoT

<table>
<thead>
<tr>
<th>Industry</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>IoT platform / middleware</td>
<td>41.6%</td>
</tr>
<tr>
<td>Home automation</td>
<td>41.1%</td>
</tr>
<tr>
<td>Industrial automation</td>
<td>36.4%</td>
</tr>
<tr>
<td>Connected / smart cities</td>
<td>33.4%</td>
</tr>
<tr>
<td>Energy management</td>
<td>33.3%</td>
</tr>
<tr>
<td>Building automation</td>
<td>26.1%</td>
</tr>
<tr>
<td>Agriculture</td>
<td>25.5%</td>
</tr>
<tr>
<td>Healthcare</td>
<td>22.7%</td>
</tr>
<tr>
<td>Automotive</td>
<td>21.7%</td>
</tr>
<tr>
<td>Transportation</td>
<td>20.1%</td>
</tr>
<tr>
<td>Education</td>
<td>17.2%</td>
</tr>
<tr>
<td>Environment</td>
<td>16.4%</td>
</tr>
<tr>
<td>Utilities</td>
<td>16.1%</td>
</tr>
<tr>
<td>Wearables</td>
<td>14.2%</td>
</tr>
<tr>
<td>Security / public safety</td>
<td>12.9%</td>
</tr>
<tr>
<td>Public utilities</td>
<td>11.3%</td>
</tr>
<tr>
<td>Retail</td>
<td>10.9%</td>
</tr>
<tr>
<td>Security / defense</td>
<td>9.4%</td>
</tr>
<tr>
<td>Fitness</td>
<td>7.9%</td>
</tr>
<tr>
<td>Banking / financial / fintech</td>
<td>7.9%</td>
</tr>
<tr>
<td>Vending</td>
<td>7.0%</td>
</tr>
<tr>
<td>Collaborative and sharing</td>
<td>4.5%</td>
</tr>
</tbody>
</table>

IoT Developer Survey 2017 - Copyright Eclipse Foundation, Inc.
IoT Architecture

SW: Middleware and Applications

HW: Sensing And Communication

Source: ZTE
IoT HW & SW

HW components used

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensors</td>
<td>86.8%</td>
</tr>
<tr>
<td>Actuators</td>
<td>50.8%</td>
</tr>
<tr>
<td>Gateway/hub device</td>
<td>50.2%</td>
</tr>
<tr>
<td>Edge node device</td>
<td>36.2%</td>
</tr>
<tr>
<td>Camera/video capture</td>
<td>35.1%</td>
</tr>
<tr>
<td>LCD display</td>
<td>33.5%</td>
</tr>
<tr>
<td>Touch screen</td>
<td>25.4%</td>
</tr>
<tr>
<td>Audio playback/speaker</td>
<td>17.4%</td>
</tr>
<tr>
<td>None</td>
<td>4.5%</td>
</tr>
<tr>
<td>Other</td>
<td>4.1%</td>
</tr>
</tbody>
</table>

SW features used

<table>
<thead>
<tr>
<th>Feature</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data analytics</td>
<td>52.6%</td>
</tr>
<tr>
<td>Mobile application</td>
<td>47.0%</td>
</tr>
<tr>
<td>Integration with existing back-end systems</td>
<td>43.9%</td>
</tr>
<tr>
<td>Cloud hosted application</td>
<td>43.5%</td>
</tr>
<tr>
<td>User application running on a device</td>
<td>37.4%</td>
</tr>
<tr>
<td>Machine learning</td>
<td>29.5%</td>
</tr>
<tr>
<td>Computer vision</td>
<td>19.9%</td>
</tr>
<tr>
<td>Trusted execution environment</td>
<td>13.4%</td>
</tr>
<tr>
<td>Voice recognition</td>
<td>13.2%</td>
</tr>
<tr>
<td>None</td>
<td>5.5%</td>
</tr>
<tr>
<td>Other</td>
<td>3.1%</td>
</tr>
</tbody>
</table>
50 Billion connected devices

- Impossible to manage using centralized control -
  distributed resource management and control are key
Hidden Costs of IoT

• Initial costs:
  – Design, HW sourcing and manufacturing, implementation

• Operational Expenses (OpEx) are recurring costs:

  - **Network communication**
    - 33-50 percent
    - Monthly/device subscription fee
    - Monthly/device usage fee

  - **Administrative labor**
    - 20-50 percent
    - Every deployed device requires at least 15 interactions/year
    - Each interaction takes at least 5 minutes

  - **Technical support**
    - 10-33 percent
    - 10% of deployed devices require support
    - T1 MTTR: 25 minutes
    - T2 MTTR: 3 to 5 hours

  - **Time to market**
    - Time to provision devices and services
    - Testing devices and services before deployment

Source: Jasper, Cisco
Average Annual Administrative Costs for 100,000 Devices

- Data analysis of more 3500 companies worldwide
- Assumptions:
  - Average salary for professional operations administrator in the USA is $48/hr
  - Each interaction requires 5 minutes of labor
  - Number of interactions varies by industry:

<table>
<thead>
<tr>
<th>Touches</th>
<th>Industry</th>
<th>Without platform:</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Security &amp; home automation</td>
<td>$5.2M</td>
</tr>
<tr>
<td>15</td>
<td>Point-of-sales company</td>
<td>$6.0M</td>
</tr>
<tr>
<td>5</td>
<td>Industrial equipment</td>
<td>$2.0M</td>
</tr>
<tr>
<td>7</td>
<td>Transportation &amp; logistics</td>
<td>$2.8M</td>
</tr>
</tbody>
</table>
Average Annual *Technical Support* Costs for 100,000 Devices

- Data analysis of more 3500 companies worldwide
- Assumptions:
  - Average tier 1 engineer salary is $20/hr, and call time is 25min
  - Average tier 2 engineer salary is $40/hr, and the mean time to resolution is 4hrs
  - 20% of support has to be escalated from Tier 1 to Tier 2
  - 10-30% of devices require technical support

### With platform:

- **Security & home automation**
  - 30% require technical support
  - Without platform: $1.45M
  - With platform: $1M for T1
  - $1.2M for T2

- **Point-of-sales company**
  - 20% require technical support
  - Without platform: $967K
  - With platform: $167K for T1
  - $800K for T2

- **Industrial equipment**
  - 25% require technical support
  - Without platform: $1.2M
  - With platform: $208K for T1
  - $1M for T2

- **Transportation & logistics**
  - 10% require technical support
  - Without platform: $483K
  - With platform: $83K for T1
  - $400K for T2
Sources of failure

• User errors
• Installation errors
• Communication problems
• Power issues (battery lifetime)
• Hardware failures
  – Soft errors: transient
  – Hard errors: permanent
Reliability, MTTF & Maintainability

• Reliability $R(t)$:
  – Probability function that a system will operate correctly in $[0,t)$

• Mean time to failure (MTTF): $\int_{0}^{\infty} R(t)\,dt$

• Maintainability: a measure of ability to restore a device to specified condition when maintenance is performed
  – If we maintain systems at an interval $T$, with the total number of maintenance events $N$, the reliability of the maintained system during the time $NT \leq t < (N + 1)T$ is:

$$R_M(t) = R(T)^N R(t - NT)$$
Permanent Failures

- Reliability during useful lifetime:
  - Modeled with exponential distribution
  - Constant failure rate \( (\lambda_f) : \)
    \[
    R(t) = e^{-\lambda_f t}
    \]
Power, Thermals, Reliability & Variability

Interrelated problems!

- Technology is rapidly scaling
- Accuracy of fabrication process is limited

Power
- Multiple power hungry subsystems
- Intense application requirements
- Limited battery capacity

Temperature
- Power dissipation increases temperature
- Big issue with no active cooling (fans)

Reliability
- Utilization and temperature stress reduces device lifetime

Variability
- Performance
- Power
- Degradation Rate

Performance-related metrics
Energy Optimization

Max(Perf) s.t. Target Battery Lifetime

- Algorithm activation time, Available energy
- Maximum performance
- Normal operation
- Battery lifetime extension
- Minimum performance

- Target battery lifetime
- Final battery energy
Integrating Temperature: Proactive Approach

Max(Perf) s.t. Target Battery Lifetime
or
Max(Battery Lifetime) s.t. App Perf

Compact Thermal Model

\[ T_{k+1} = AT_k + BF_{k+1} \]

Example: \( F = 0 \)
\[ T_{k+1} = AT_k \]

Mean error = 1.1°C
Integrating Temperature: Proactive Approach

Max(Perf) s.t. Target Battery Lifetime
or
Max(Battery Lifetime) s.t. App Perf

Energy savings:

- Nexus 5 smartphone
- Profile of popular application
- Comparison between off-the-shelf baseline and proposed framework
Reliability Degradation due to Hot Spots

- Thermal hot spots [Failure Mechanisms for Semiconductor Devices, JEDEC]
- **Electromigration:**
  - Transport of material caused by gradual movement of ions in the interconnect lattice
  - Eventually leads to opens and shorts
- **Stress migration:**
  - Mechanical stress causes metal atoms in the interconnect to migrate
  - Similar failures as electromigration
- **Time dependent dielectric breakdown:**
  - Result of the gate dielectric’s gradual wear-out
  - Leads to transistor failure

\[
\lambda \propto e^{-\frac{E_a}{kT}} \quad \lambda: \text{Failure rate; } T: \text{temperature}
\]

- \( E_a \): Activation energy, \( k \): Boltzmann’s constant
- Increasing temperature increases \( \lambda \)
- 10 – 15 C increase in temperature causes \(~2\times\) increase in failure rate
Reliability Degradation due to Thermal Cycles

- **Thermal cycling** [JEDEC]
  - Different expansion coefficients
  - Fatigue failures

\[ \lambda \propto |\Delta T|^q f \]

- Increases with:
  - \( \Delta T \): Magnitude of variation
  - \( f \): Frequency of cycles
- \( \Delta T \) increases by 10°C
  (\( q=4 \) for metallic structures)
  - Failures happen 16 times sooner

[Figures showing temperature variation and MTTF calculations with different labels and values]

[Rosing et al., TVLSI'07]
Reliability Management

- **Reliability**: “measure” of device integrity
  - Sensors (available only on prototypes)
  - Mathematical models (Online Emulation)
- Depend on **temperature** and voltage stress:
  
  \[ R(T, V, t) \]

**Dynamic Reliability Management:**

\[ R(t_{LIFE}) \geq R_{TARGET} \]

**Borrowing strategy**

\[
\begin{align*}
V_{average} & \leq V_{LTC} \\
T_{average} & \leq T_{LTC}
\end{align*}
\]
Reliability Emulation (“RelDroid”)

RelDroid is used for prototyping on real devices

![Diagram of Reliability Emulation System]

- **Reliability Emulator Module and Driver**
- **Reliability Model**
- **Reliability Log**
- **Sensor Driver**
- **Power Management Driver**
- **Temperature Sensor**
- **Voltage Sensor**
- **Voltage/Frequency Selection**
- **MPSoc**

**Variables and Parameters**
- $R(t)$
- $T_{AVG}(t)$
- $V_{AVG}(t)$
- $T(t)$
- $V(t)$
- $Freq(t+1)$
- $Perf$
**Optimal Reliability Controller**

**Optimal Long Term Controller**: computes the average target values for voltage and **temperature** \( V_{LTC} \) and \( T_{LTC} \)

**Optimal Short Term Controller**: determines frequency to apply \( F \) and **task allocation** \( X_{alloc} \)

Solve optimal problem using **convex optimization**

→ Large computation overhead for OSTC
Workload Aware Reliability Manager

Local Reliability Manager
- OLTC → STC

Processor
- core_1 → task_1
- core_N

Thermal Controller
- task_M

Long Intervals

Medium Intervals

Short Intervals

**Local Reliability Manager**

- OLTC: Reference voltage
- STC: Reference temperature
- $V_{LTC}$: Reference voltage
- $T_{LTC}$: Reference temperature
- $f_{APP}/V_{APP}$: Applied frequency/voltage

**Processor**

- core_1

**Thermal Controller**

- task_1
- task_M

**Short Term Controller:**
- Short Interval (SI, ms)

**Thermal Controller:**
- Medium Intervals (MI, s)
- Assigns H tasks to cooler cores first ($X_{alloc}$)

Borrowing strategy:

- $V_{average} \leq V_{LTC}$
- $T_{average} \leq T_{LTC}$
Implementation

Current executing apps

Reference voltage and temperature

Average voltage and temperature

Process allocation

Voltage and temperature settings

Voltage and temperature sensors

Slow rate  Medium rate  Fast rate

App Monitor

RelDroid

WARM
Implementation

USER

APP MONITOR DRIVER

SCHED

Kernel

Hardware

App Monitor

RelDroid

WARM

Current PID

foreground app PID

Currently executing apps

Reference voltage and temperature

Average voltage and temperature

Voltage and frequency settings

Voltage and temperature settings

Voltage and temperature sensors

Faulty apps

Favorite apps

OLTC

TC

RELIABILITY MODEL

Implementation overhead

<table>
<thead>
<tr>
<th>Overhead</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature sensor reading</td>
<td>160 cycles</td>
</tr>
<tr>
<td>Voltage sensor reading</td>
<td>170 cycles</td>
</tr>
<tr>
<td>Frequency change</td>
<td>16824 cycles</td>
</tr>
<tr>
<td>STC algorithm</td>
<td>74 cycles</td>
</tr>
<tr>
<td>Passing process ID</td>
<td>5 µs</td>
</tr>
<tr>
<td>LTC execution</td>
<td>342 ms</td>
</tr>
<tr>
<td>TC execution</td>
<td>1.5 ms</td>
</tr>
</tbody>
</table>
Testing and Validation

Odroid XU3 development board:
- ARM big.LITTLE architecture
- OS is Android 4.4.4 with Linux kernel 3.10.9.
- The device has 4 integrated voltage/current/power monitoring sensors (LITTLE cluster, big cluster, GPU and memory)

Virtual platform:
- 4 cores
- configured based on measurements from the Cortex A15 cores of the ARM big.LITTLE
- CVX for solving optimization

**Temperature model**
\[ T_{next} = thermal\_model(T, P) \]

**Power model:**
\[ P(f) = af^3 + bf \]

**Virtual cores**
- \( f_0 \)
- \( f_1 \)
- \( f_2 \)
- \( f_3 \)

**Frequency adaptation**
Comparison Results: Optimal Policy vs. Heuristic (simulation)

Both optimal and WARM are effective in meeting the average constraints.

WARM runs more than 400x faster.

1 jiffy = 5ms (sched. tick)
Measurement Comparison with Standard Governors

**Without background activity:**
Performance **comparable** to max frequency for high quality (H) applications (**in the 4%**).

**With background activity:**
it improves performance while meeting the target lifetime. **100% lifetime improvement**
Impact of Variability

**STATIC VARIATIONS:**
- Channel length
- Threshold voltage
- Operating frequency (performance)
- Power consumption
- Degradation rate (DR)

**DYNAMIC VARIATIONS:**
- Voltage
- Temperature
- Workload
- User
- Ambient temperature

**DYNAMIC VARIABILITY MANAGEMENT (DVM)**

[Graph showing the impact of DVM on performance, power, and degradation rate with and without DVM]

**TRANSISTOR**

**PROCESSOR**
A comprehensive management framework for power, temperature, reliability, & variability management in individual devices

User Experience Requirements

Comprehensive Management Framework

- Variability
- Reliability
- Temperature
- App perf
- Battery
- Power

Control decisions

Target system

Online Reliability Emulation

Online Variability Emulation

System feedback

Average target → Unexploited margins

LTC: Long Term Controller
STC: Short Term Controller
Networks of IoT Devices

Internet of Things:
- Energy and reliability are important issues -> maintenance costs
- Challenges: heterogeneity, distributed nature, large scale

Research directions:
- Framework for energy / reliability management of the IoT infrastructure
- Control of multiple heterogeneous control variables
- Automatic recognition of user experience requirements
Reliability of Networks of Devices

• Series combination: all components must operate for the system to function correctly - \( R = R_A \times R_B \times R_C \)

• Parallel combination: system fails if all components fail: \( F_s(t) = F_a(t) F_b(t) F_c(t) \) where \( F(t) = 1 - R(t) \)

• Most IoT systems are mixed
  – Soft redundancy: imperfect replacements
Research Challenge

Designing an IoT infrastructure that meets performance requirements while maximizing reliability and minimizing maintenance costs