MobileInsight
Extracting and Analyzing Cellular Network Information on Smartphones

Yuanjie Li¹, Chunyi Peng², Zengwen Yuan¹, Jiayao Li¹, Haotian Deng², Tao Wang³
¹University of California, Los Angeles
²The Ohio State University
³Peking University
“Anytime, Anywhere” Cellular Network Service
Critical Cellular Operations to Users/Apps

- Session Management (SM)
- Mobility Management (MM)
- Radio Resource Control (RRC)
- Link Layer (MAC/RLC/PDCP)
- Physical Layer (PHY)

Software

Hardware
But They are **Closed**...
Can We Have **Open** Access to Runtime Cellular Network Operations?

- Why my 4G phone switches to slow 2G?
- Why my phone drains battery quickly?
- 4 signal bars, but why no data service?
It’s Not That Simple

- No approaches cover all necessary features

<table>
<thead>
<tr>
<th>Full coverage</th>
<th>Fine grained</th>
<th>Analysis</th>
<th>At scale</th>
<th>In-phone</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Android APIs" /></td>
<td><img src="%E2%9C%98" alt="Android APIs" /></td>
<td><img src="%E2%9C%98" alt="Android APIs" /></td>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Android APIs" /></td>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Android APIs" /></td>
</tr>
<tr>
<td><img src="%E2%9C%94%EF%B8%8F" alt="External Tools" /></td>
<td><img src="%E2%9C%94%EF%B8%8F" alt="External Tools" /></td>
<td><img src="%E2%9C%98" alt="External Tools" /></td>
<td><img src="%E2%9C%98" alt="External Tools" /></td>
<td><img src="%E2%9C%98" alt="External Tools" /></td>
</tr>
<tr>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Operator-side cellular analytics" /></td>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Operator-side cellular analytics" /></td>
<td><img src="%E2%9C%94%EF%B8%8F" alt="Operator-side cellular analytics" /></td>
<td><img src="%E2%9C%98" alt="Operator-side cellular analytics" /></td>
<td><img src="%E2%9C%98" alt="Operator-side cellular analytics" /></td>
</tr>
</tbody>
</table>
Our Solution: **MobileInsight**

- A **software tool** for commodity phones
- A **community tool** that can be built and shared together

<table>
<thead>
<tr>
<th>Feature</th>
<th>MobileInsight</th>
<th>Android APIs</th>
<th>External Tools (e.g., QXDM)</th>
<th>Operator-side cellular analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full coverage</td>
<td>✔️</td>
<td>✘</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Fine grained</td>
<td>✔️</td>
<td>✘</td>
<td>✔️</td>
<td>✘</td>
</tr>
<tr>
<td>Analysis</td>
<td>✔️</td>
<td>✘</td>
<td>✘</td>
<td>✘</td>
</tr>
<tr>
<td>At scale</td>
<td>✔️</td>
<td>✔️</td>
<td>✘</td>
<td>✘</td>
</tr>
<tr>
<td>In-phone</td>
<td>✔️</td>
<td>✔️</td>
<td>✘</td>
<td>✘</td>
</tr>
</tbody>
</table>
MobileInsight Overview

Software
- Monitor

Hardware
- SM
- MM
- RRC
- MAC/RLC/PDCP
- PHY
In-device Runtime Monitor

How to expose runtime cellular messages to user space?
Challenge: No Ordinary In-device Schemes
Solution: **Side-Channel** Across SW-HW Boundary

Raw cellular messages from **Radio Interface Layer** via USB to **/dev/diag** in **Software** leads to **Coarse-grained cellular info** for **Android APIs**. **Parsers** and **Proxy** analyze these messages for monitoring purposes.
Cellular Protocol Analytics

How to unveil runtime cellular protocol behaviors?
Two Dimensions for Each Protocol

- **State dynamics extraction**
  - Device side
  - Regulated by cellular standards

- **Operation logic inference**
  - Network side
  - Non-standardized, operator-specific

![Diagram of state transitions and handoff decision logic]
Protocol Analytics: Tracking State Dynamics

- Current protocol state, transition events and causes
  - **RRC**: Radio connectivity status and power-saving mode
  - **MM**: Device registration status
  - **SM**: Data session activity and QoS status
Protocol Analytics: Tracking State Dynamics

- **Observation**: regulated by the cellular standards
- **Reference state machine + runtime message**

![State Machine Diagram]

**Parameters**:
- $T_1 = 100\text{ms}$
- $T_{\text{shortDRX}} = 20\text{ms}$
- $T_2 = 2T_{\text{shortDRX}}$

**Downlink data**

---

**Note**: The document has been developed within the 3rd Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP. The document has not been subject to any approval process by the 3GPP Organizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPP only. The Organizational Partners accept no liability for any use of this Specification. Specifications and reports for implementation of the 3GPP™ system should be obtained via the 3GPP Organizational Partners' Publications Offices.
Protocol Analytics: Inferring Operation Logic

• Algorithm to determine protocol configurations and actions
  • Example: handoff decision logic

BS 1’s handoff decision logic:
• Switch to BS 2 (4G) if $\text{RSS}_{2(4G)} > \text{RSS}_{1(4G)} + 3 \text{ dBm}$
• Otherwise, switch to BS 3 (3G) if $\text{RSS}_{1(4G)} < -110 \text{ dBm}$ and $\text{RSS}_{3(3G)} > -90 \text{ dBm}$
Inferring Operation Logic is Not Simple

- **Challenge #1:** Non-standardized, carrier-specific operations
- **Challenge #2:** Internal logic, not visible by end device

BS 1’s handoff decision logic:
- Switch to BS 2 (4G) if $\text{RSS}_{2(4G)} > \text{RSS}_{1(4G)} + 3 \text{ dBm}$
- Otherwise, switch to BS 3 (3G) if $\text{RSS}_{1(4G)} < -110 \text{ dBm}$ and $\text{RSS}_{3(3G)} > -90 \text{ dBm}$
Observation: Operation Logic is Not Arbitrary

- Many network-side operations are stateful

BS 1’s handoff decision logic:
- Switch to BS 2 (4G) if $\text{RSS}_2^{(4G)} > \text{RSS}_1^{(4G)} + 3 \text{ dBm}$
- Otherwise, switch to BS 3 (3G) if $\text{RSS}_1^{(4G)} < -110 \text{ dBm}$ and $\text{RSS}_3^{(3G)} > -90 \text{ dBm}$
Observation: Operation Logic is Not Arbitrary

- Many network-side operations are **stateful** and **interactive**

**Solution:** **Online state machine inference**

- **Monitor 4G**
  - $\text{RSS}_2 > \text{RSS}_1 + 3 \text{dBm}$
  - $\text{RSS}_1 < -110 \text{dBm}$
- **Monitor 3G&4G**
  - $\text{RSS}_1 < -110 \text{dBm}$
  - $\text{RSS}_3 > -90 \text{dBm}$

**Meas Control:** Monitor 4G
**Meas Report:** $\text{RSS}_2 > \text{RSS}_1 + 3$
**Handoff command:** to BS2

- **BS 1 (4G)**
- **BS 2 (4G)**
- **BS 3 (3G)**
State Machine Inference: Partial Recovery

- Runtime sample sequence 1
State Machine Inference: Partial Recovery

• Runtime sample sequence 2
State Machine Inference: Aggregation

- **Monitor 4G**: RSS\(_2 > \) RSS\(_1 + 3\) dBm
  - Handoff to 4G

- **Monitor 3G&4G**: RSS\(_1 < -110\) dBm
  - Handoff to 3G
  - RSS\(_1 < -110\) dBm, RSS\(_3 > -90\) dBm

- **Meas Control**: Monitor 4G
  - Meas Report: RSS\(_1 < -110\)

- **Meas Control**: Monitor 3G&4G
  - Meas Report: RSS\(_2 > 90\)

- **Handoff command**: to BS3

BS 2 (4G)  BS 3 (3G)  BS 1 (4G)
MobileInsight APIs

```python
src = OnlineMonitor()
lte_rrcAnalyzer = LteRrcAnalyzer()
wcdma_rrcAnalyzer = WcdmaRrcAnalyzer()
lte_rrcAnalyzer.set_source(src)
wcdma_rrcAnalyzer.set_source(src)
src.run()
```
Showcase Examples

How can MobileInsight stimulate new apps and research?
Example 1: Fix Our Phone’s Network Failures

- **How:** Track protocol state dynamics
- **Root cause:** device-side misconfiguration
- **Fix:** disable VoLTE when device is in 3G

4 signal bars, but why no data service?
Example 2: Boost Our Phone’s Data Speed

• **How:** Analyze inferred handoff decision logic
• **Root cause:** suboptimal FCFS strategy
• **Advice:** disable 2G when 4G is available

Why switch to slow 2G despite good 4G coverage?
Research Empowered by MobileInsight

• Security loophole detection, failure resolution, handoff advisor, etc.

• iCellular [NSDI’16]: Device-customized multi-carrier roaming

• MMDiag [SIGMETRICS’16]: mobility misconfiguration detection
Evaluation

Coverage, performance, accuracy and system overhead
Wide Coverage of Phone Models

- **Current version**: rooted Android with Qualcomm chipset
- **MTK/Intel and iOS support**: under development

<table>
<thead>
<tr>
<th>Mobile OS</th>
<th>Chipset</th>
<th>Feasibility</th>
<th>Current Version (2.1.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>Qualcomm</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>MediaTek</td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td></td>
<td>Intel XMM</td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td>iOS</td>
<td>All</td>
<td>✔</td>
<td>✗</td>
</tr>
</tbody>
</table>
Wide Coverage of Cellular Protocols/Messages

- 3G/4G signaling messages and 4G-L1/L2 messages
- Characterization of cellular message patterns

<table>
<thead>
<tr>
<th>Dataset size</th>
<th>245.24GB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total messages</td>
<td>72,389,300</td>
</tr>
<tr>
<td>Protocol Layers</td>
<td></td>
</tr>
<tr>
<td>4G-PHY</td>
<td>71.8%</td>
</tr>
<tr>
<td>4G-MAC</td>
<td>9.0%</td>
</tr>
<tr>
<td>4G-PDCP</td>
<td>8.3%</td>
</tr>
<tr>
<td>3G/4G-RRC</td>
<td>10.0%</td>
</tr>
<tr>
<td>3G/4G-MM/SM</td>
<td>0.6%</td>
</tr>
<tr>
<td>3GPP2-EvDo/CDMA</td>
<td>0.3%</td>
</tr>
</tbody>
</table>
Real-time Processing of Cellular Messages

• 99% messages’ parsing and analyzing within 0.8ms
  • Worst case observed: 33ms
Accurate Cellular Protocol Analytics

- **Tracking Protocol State Dynamics**: identical as QXDM
  - Same cellular message sources

- **Inference of Handoff Operation Logic**
  - 10-fold cross validation: 87.5%~95.3% prediction accuracy

Table 9: Accuracy for predicting upcoming handoffs.

<table>
<thead>
<tr>
<th></th>
<th>AT&amp;T</th>
<th>T-Mobile</th>
<th>Sprint</th>
<th>Verizon</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Samples</td>
<td>11,050</td>
<td>10,178</td>
<td>10,042</td>
<td>2,741</td>
</tr>
<tr>
<td>Accuracy</td>
<td>90.7%</td>
<td>91.8%</td>
<td>95.3%</td>
<td>87.5%</td>
</tr>
</tbody>
</table>
Acceptable System Overhead

• **CPU utilization**: 1%–7%

• **Memory**: 30MB at maximum

• **Energy**: 11–58mW extra power (on Samsung S5)
New Version: v2.1.1

- More cellular protocol support
- Cellular data sharing
- New APIs for mobile applications
- In-phone cellular log browser
- ...

...
Toward Open and Large-Scale Cellular Datasets

• **Initial dataset release**
  • 30+ users, 8 US/Chinese network operators
  • 13-month collection (Jul 2015 – Sep 2016)
  • ~245GB 3G/4G cellular traces

• **Everyone can contribute to the dataset anywhere, anytime!**
  • Online trace submission or background data sharing

More information:
http://metro.cs.ucla.edu/mobile_insight/insightshare.html
New Research Opportunities Made Possible

Mobile big data analytics

Cellular protocol refinements

Security threats detections

Cross-layer app enhancements
Conclusion

• Open access to cellular operations benefits everyone
  • Mobile users, researchers, developers and even operators

• **MobileInsight**: a first effort toward an open cellular world

• More **community efforts** are needed for extension
  • A tool *for* the community and *by* the community
Try **MobileInsight** and explore more!

http://metro.cs.ucla.edu/mobile_insight

**Yuanjie Li**¹, Chunyi Peng², Zengwen Yuan¹, Jiayao Li¹, Haotian Deng², Tao Wang³

¹University of California, Los Angeles
²The Ohio State University
³Peking University