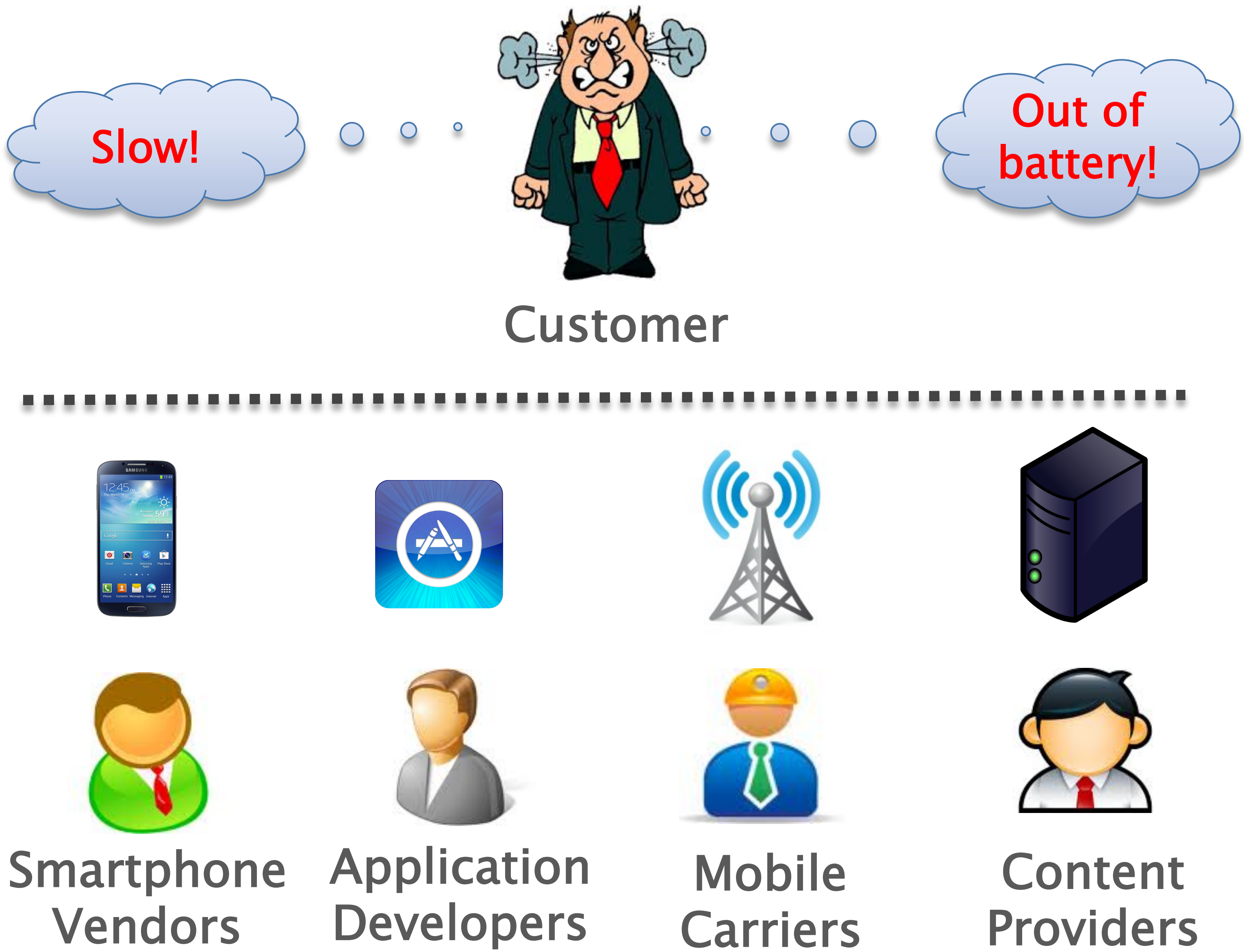
# Performance and Power Characterization of Cellular Networks and Mobile Application Optimizations

Junxian Huang
University of Michigan

April 25, 2013

# Smartphone Ecosystem



# Key Motivations

- *Performance* and *power* characteristics of cellular networks and smartphone applications are not well studied
  - especially for the new LTE networks
- Many smartphone applications are not well designed with inefficient use of network, radio and power resources
  - mainly due to the unawareness of the cellular specific characteristics

# My Research Goal

To design practical and effective methods to characterize the resource utilization (bandwidth, energy, radio resource…) in cellular networks and improve the mobile application design.

- Challenges
  - Multiple entities involved
  - The solution needs to be general and practically deployable
  - Requires evaluation using real cellular workload

# Thesis Statement

*We devise a suite of systematic methodology and tools to accurately measure the performance and power characteristics of cellular networks (both 3G and the LTE 4G networks) as well as the mobile applications, and to optimize the mobile application design based on such knowledge*

# Roadmap

- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

# Roadmap

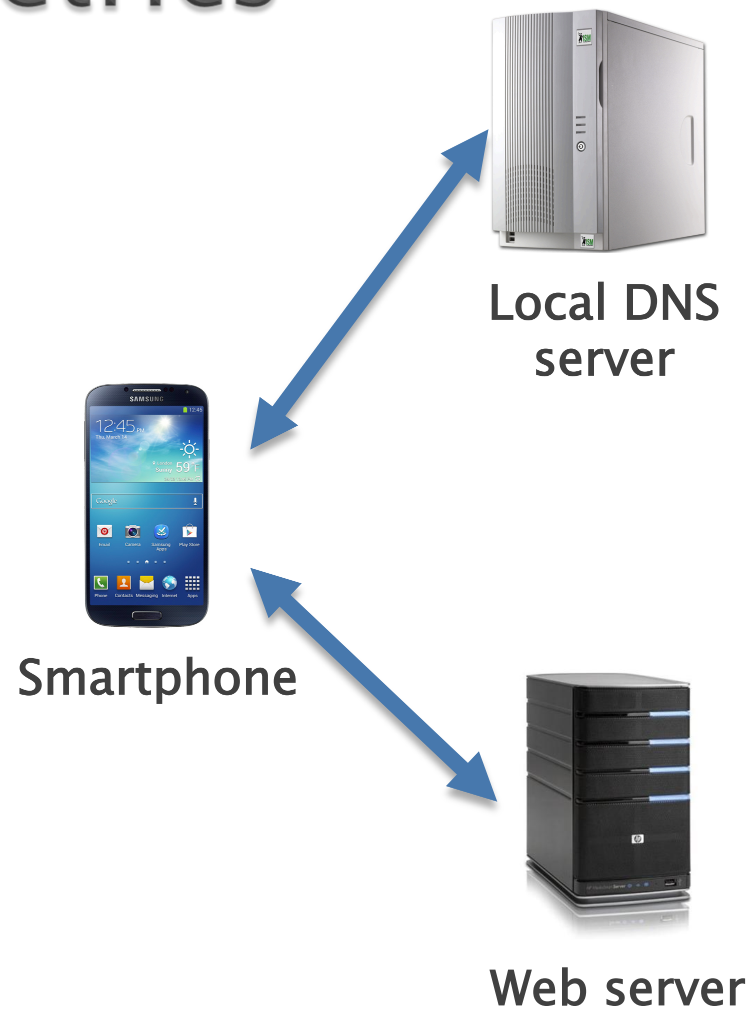
- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

# Design Goals of MobiPerf


- Measure most relevant network characteristics of cellular networks
- Measurement results not be affected by any external factors
- Attract a large number of users covering different carriers and geographical regions
- Benefit the research community in terms of both tool and data

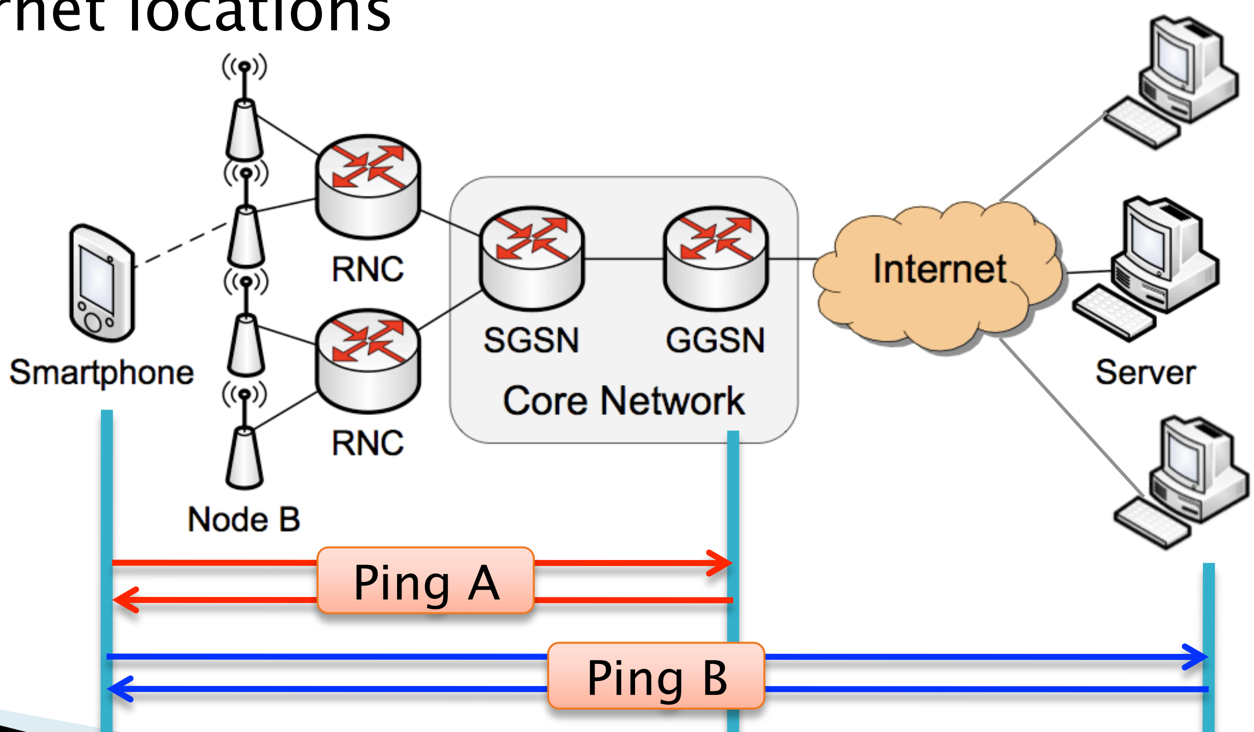
# Cellular Network Metrics

- DNS (domain resolution)
  - Local DNS response time
- TCP
  - Common TCP metrics: downlink/uplink throughput, round-trip time (RTT), packet loss rate, etc.
  - For TCP throughput test, we choose the closest nodes among the *120+ global M-Lab nodes* and use parallel TCP connections

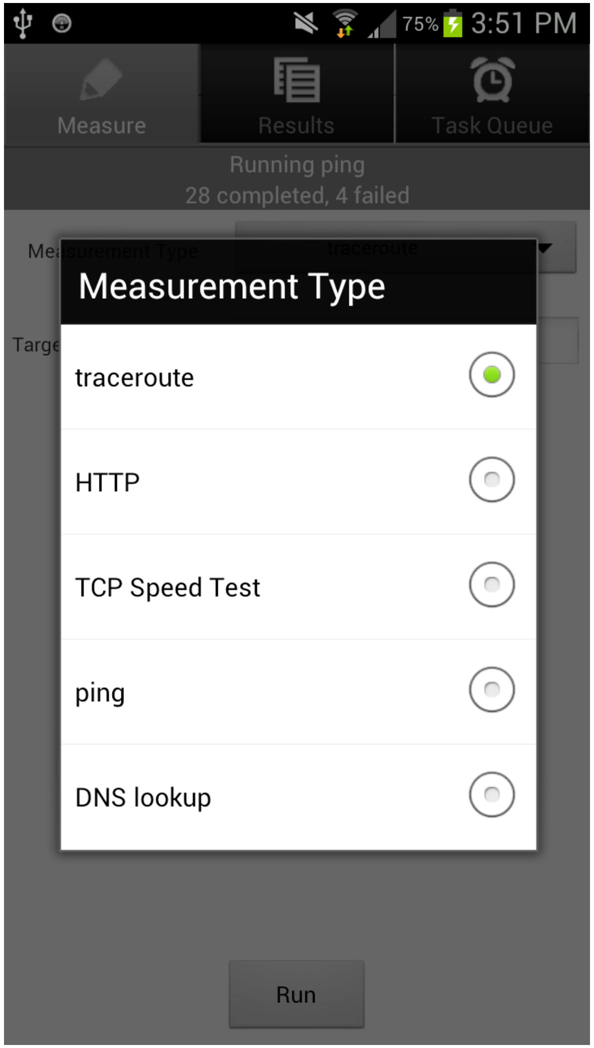


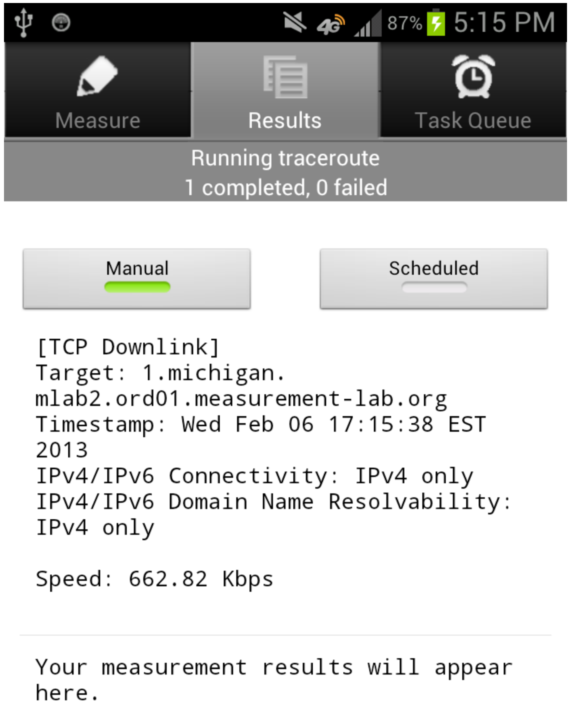
# Cellular Network Metrics

- Ping (Impact of diverse Internet servers)
  - Ping A: Ping latency to the first responsive IP hop
  - Ping B: Ping latency to landmark servers at diverse Internet locations



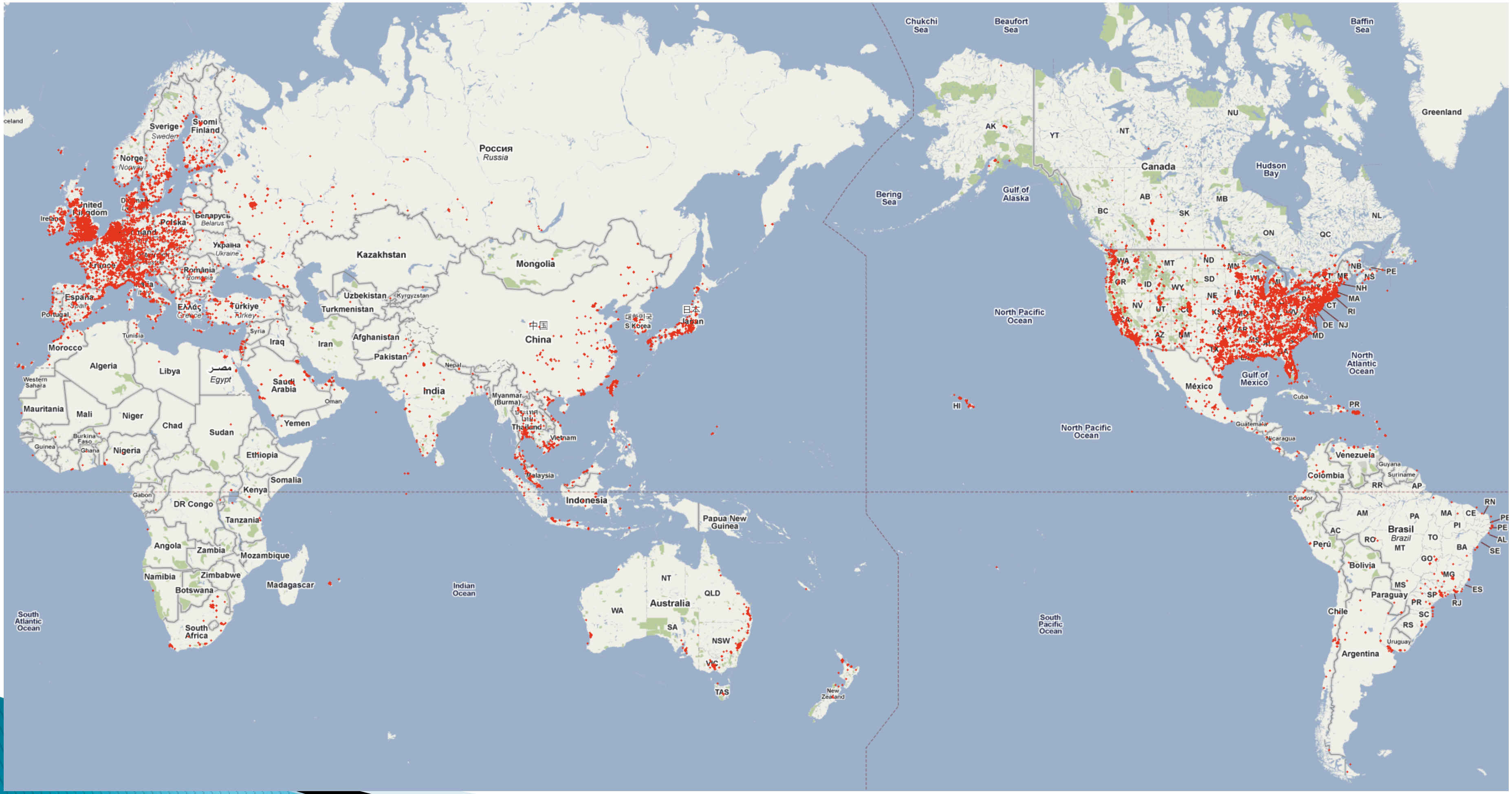
# http://mobiperf.com

# Coverage of MobiPerf Data

70K+ users 200K+ runs 84 carriers 147 countries



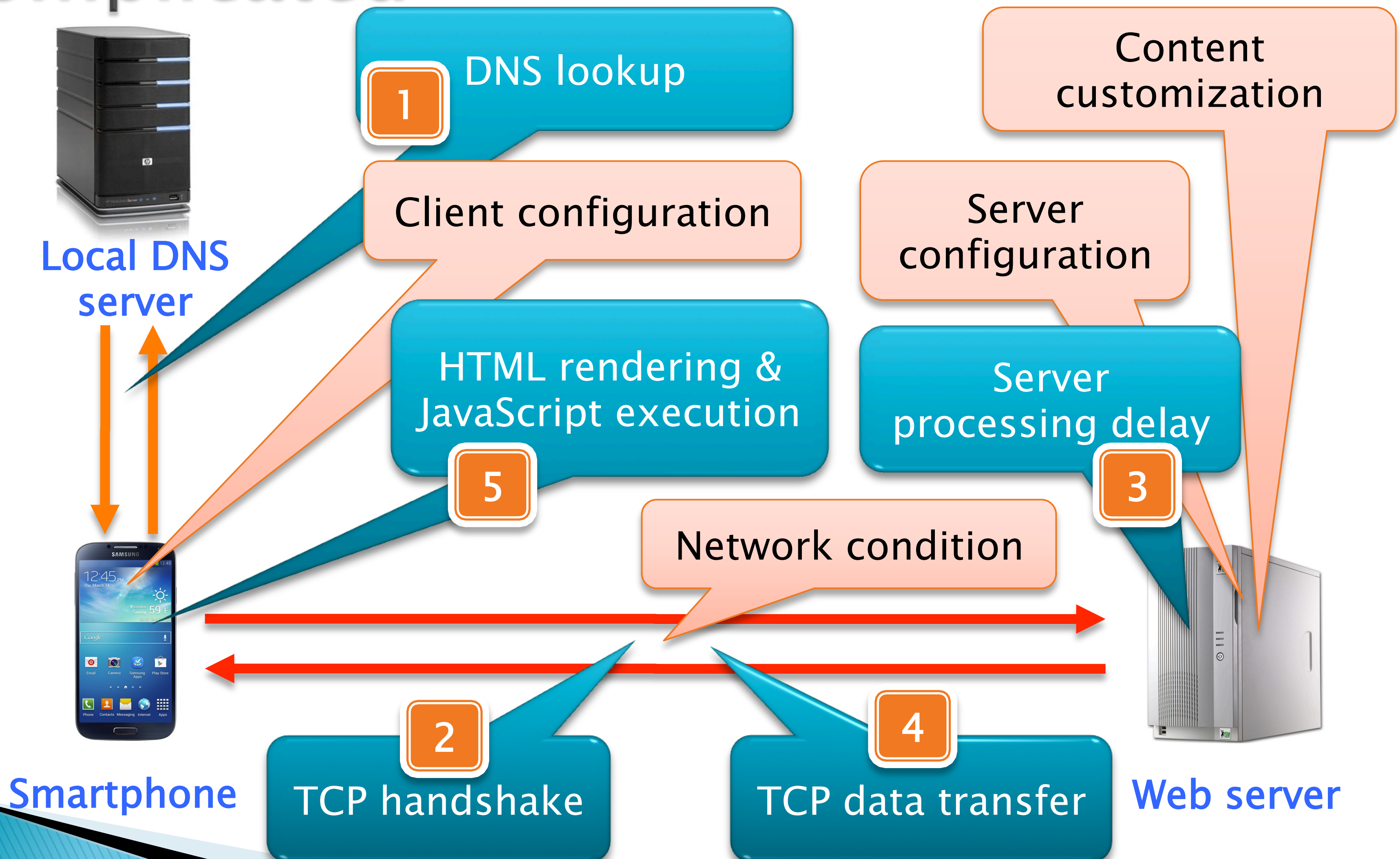
# MobiPerf Benefits the Research Community

- Open source
  - https://github.com/Mobiperf/MobiPerf
  - Other researchers can easily add modular measurement tasks into MobiPerf tool
- Open data
  - gs://openmobiledata_public (Google storage)
- Recognition
  - In 2011, *People's Choice Award* and *Open Internet App Reward* in the *FCC Open Internet Apps Challenge*

# Roadmap

- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

# Smartphone web browsing is complicated



# Key challenges of web performance analysis

- Multiple factors have joint impact
  - Network condition
  - Client settings
  - Server configurations
  - Content differences
- Lack of low-level information and access
  - No visibility into link layer
  - Difficult to instrument client OS and application

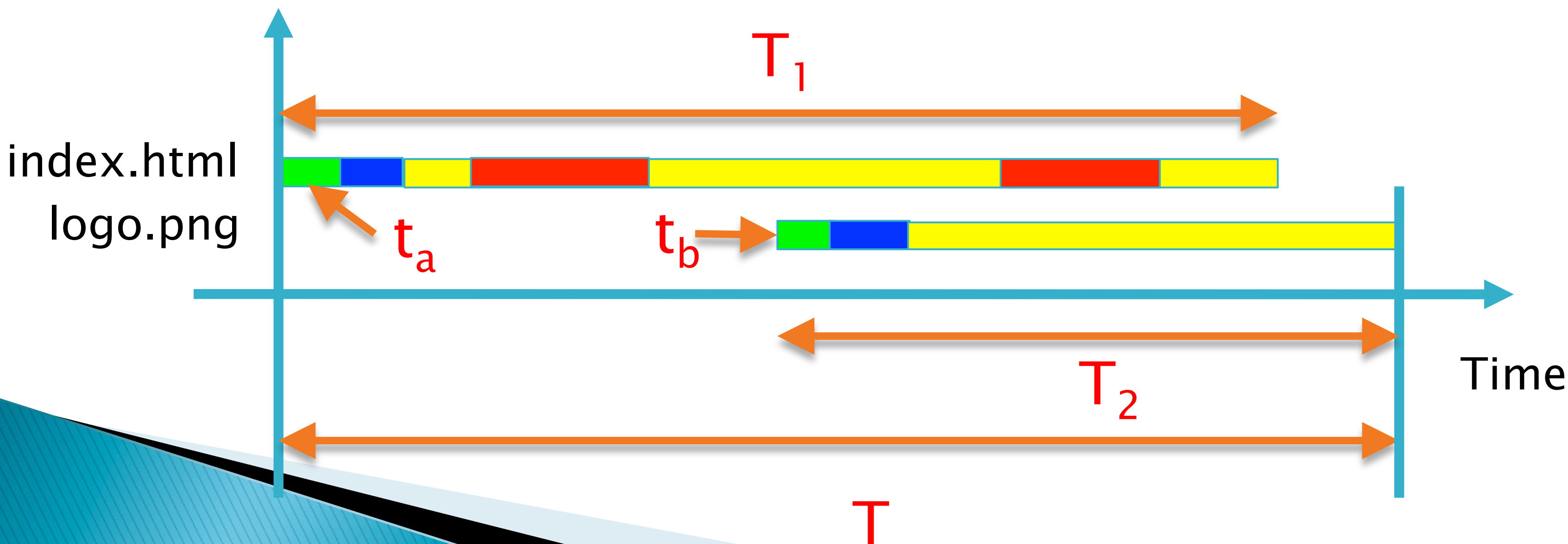
# How to isolate impact of individual factor?

- Vary one factor at a time
  - Network condition
    - Simulate cellular network with WiFi via *iptables*
  - Client settings
    - Control TCP concurrency
  - Server configurations
    - Change compression mode and server parallelism
  - Content difference
    - Host static website snapshots locally

# How to identify performance bottlenecks?

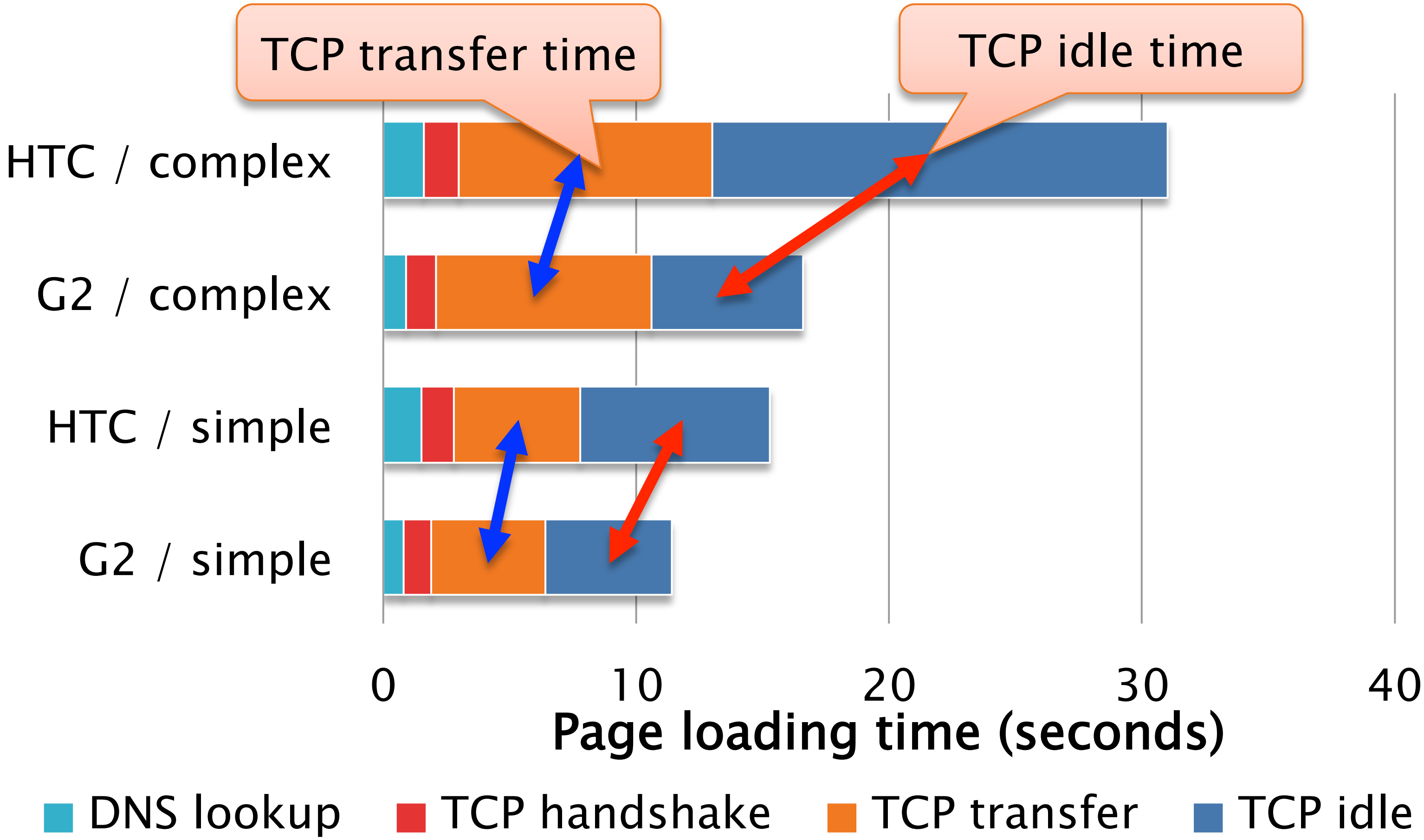
- TCP flow analysis technique to break down page loading time
  - DNS lookup time
  - TCP handshake time
  - TCP idle time
  - TCP transfer time

$$T_{DNS} = \frac{(t_a + t_b)}{(T_1 + T_2)} T$$



# Web browsing breakdown

- Web browsing performance bottlenecks
  - Client execution (TCP idle time)
  - Network transfer (TCP transfer time)



# Lessons from web browsing performance study

- Performance bottlenecks of web browsing
  - Client execution
  - Data download (less for LTE)
- Higher **concurrency** significantly improves performance
- **Compression** is usually helpful
- **Content customization** is beneficial
  - Data URL scheme
  - Minification
  - Redirection reduction

# Roadmap

- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

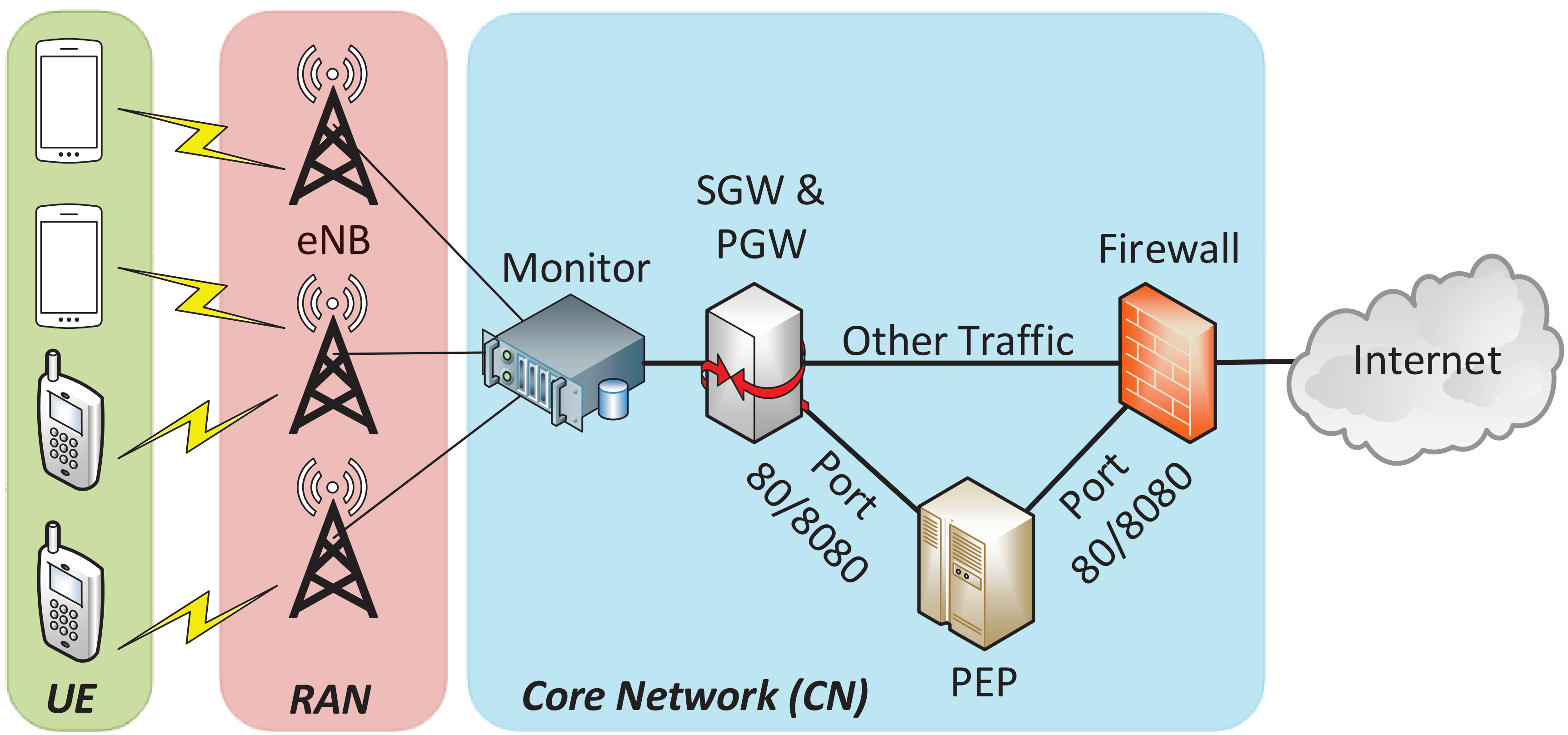
# LTE is New, Requires Exploration

- 4G **LTE (Long Term Evolution)** is future trend
  - Initiated by 3GPP in 2004
    - 100Mbps DL, 50Mbps UL, <5ms latency
  - Entered commercial markets in 2009 and now available in more than 10 countries

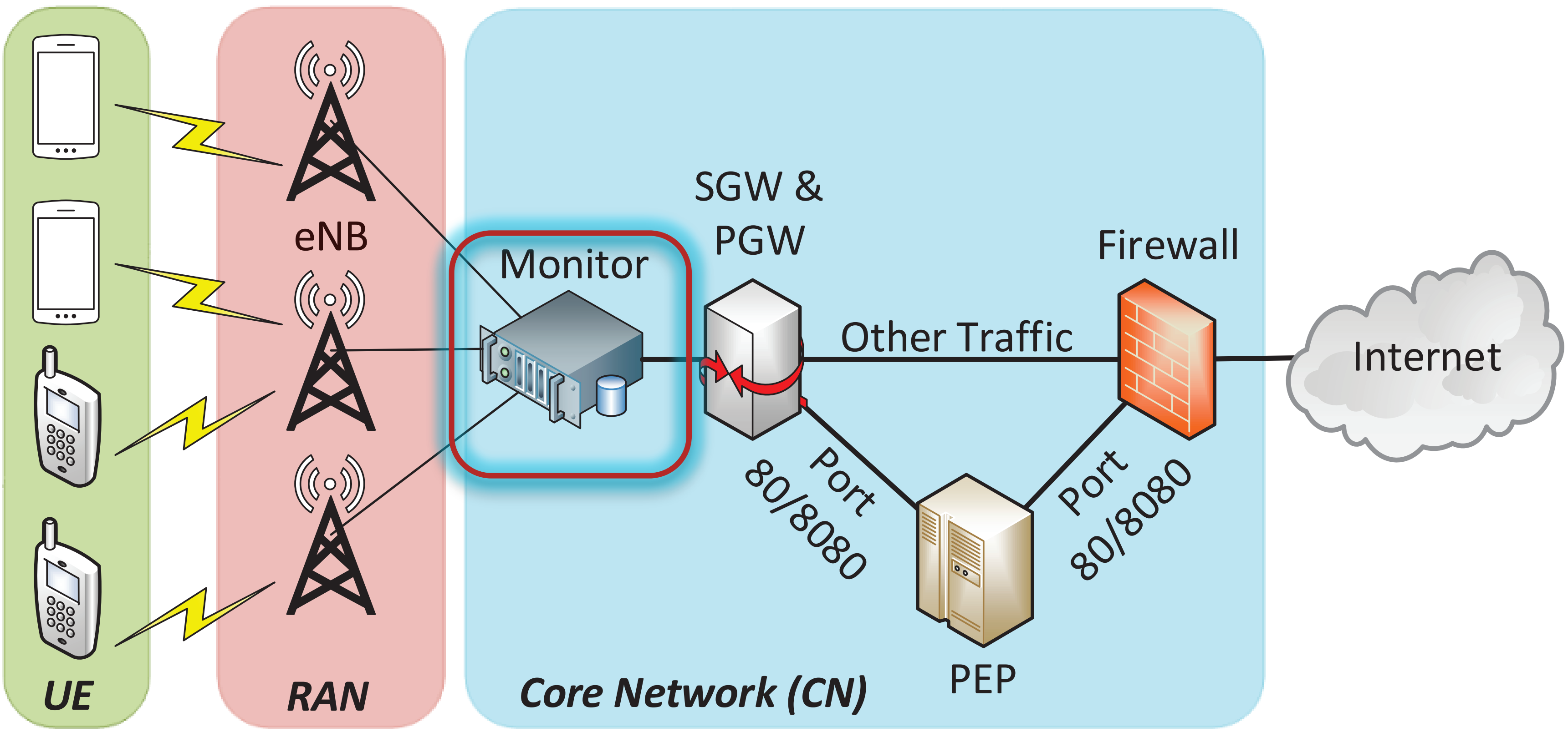
# Motivations

- LTE has not been extensively studied empirically in a commercial network setting
- Study how network resources are utilized across different protocol layers for real users
- Evaluate the benefits of increased bandwidth for mobile applications and essential network protocols to identify their limitations

# LTE network topology of the studied carrier



# LTE network topology of the studied carrier



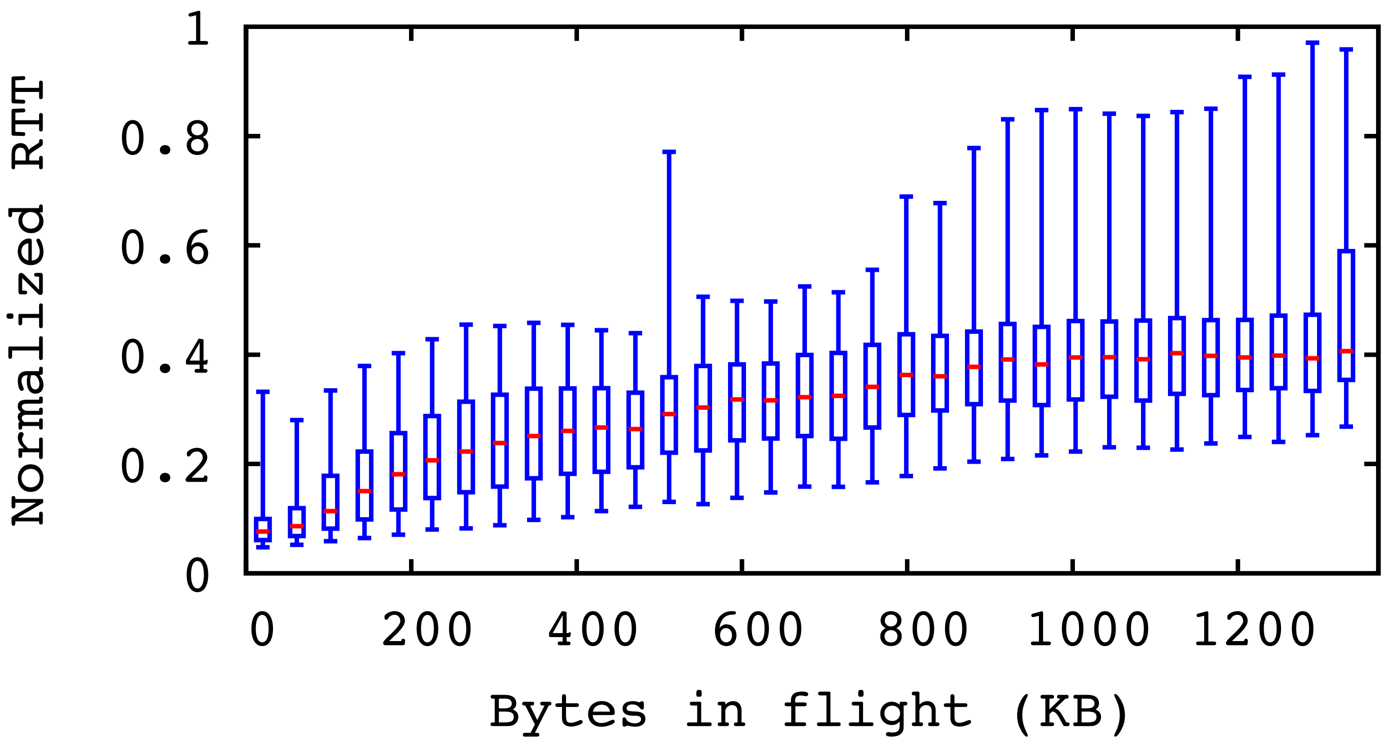

Data collection point

# Data set studied

- Data set statistics
  - Covering a fixed set of 22 eNBs at a large metropolitan area in the U.S.
  - Over 300,000 users
  - 3.8 billion packets, corresponding to 3 TB of LTE traffic
  - Collected over 10 consecutive days
- Data contents: packet header trace consisting of
  - IP and transport-layer headers
  - A 64-bit timestamp for each packet
  - No payload data is captured except for headers of HTTP

# Queueing delay

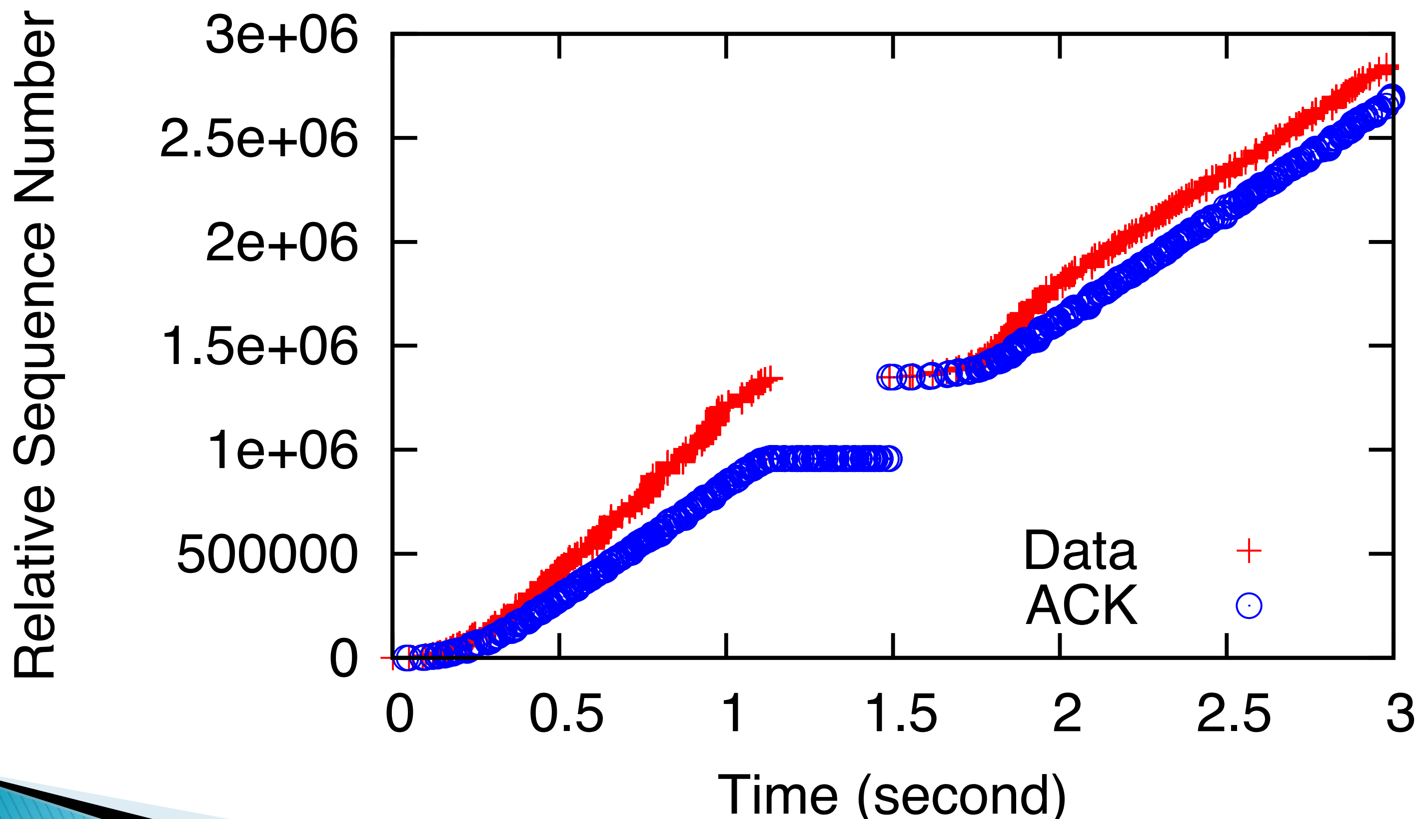
- Large buffers in the LTE networks may cause high queuing delays



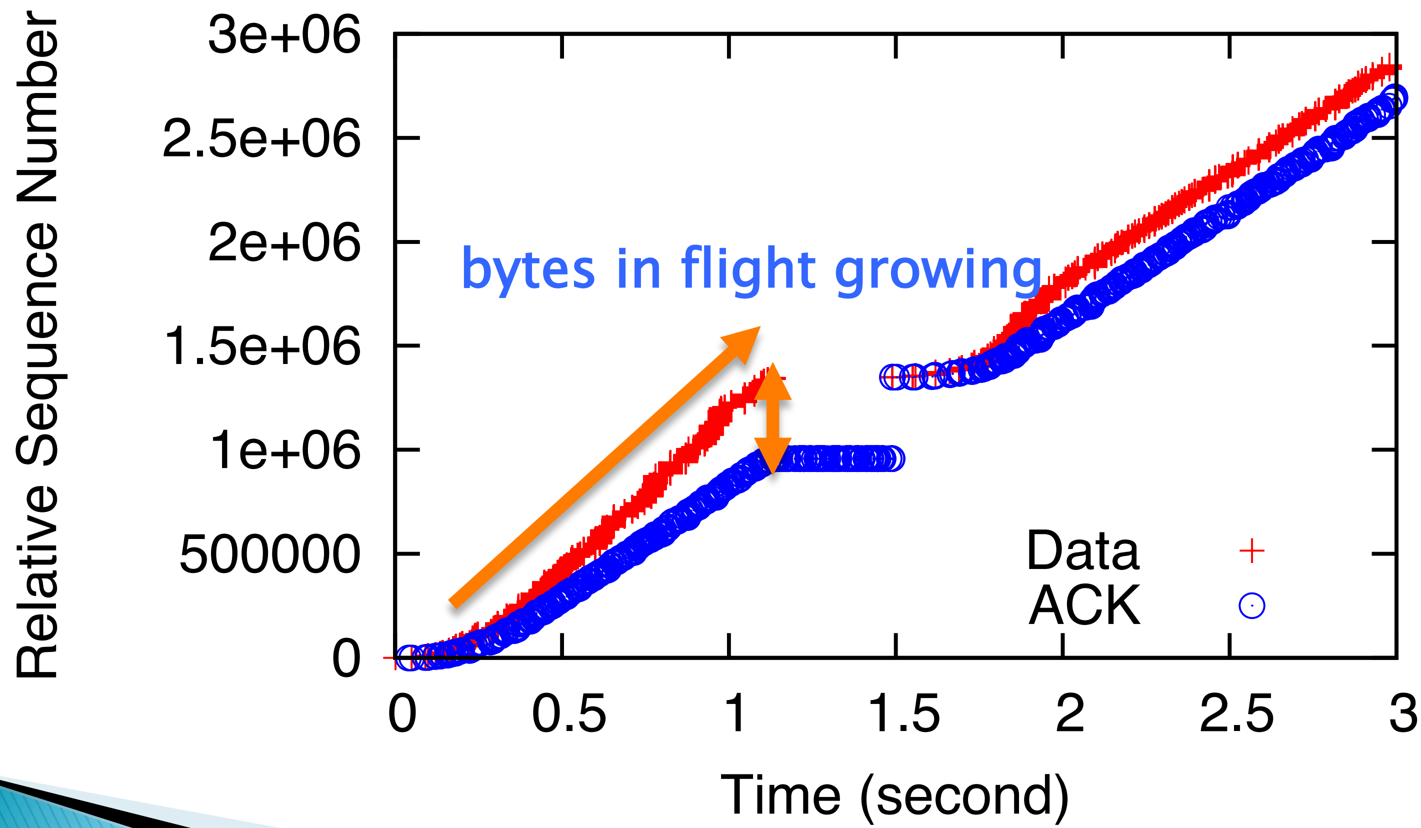

Bytes in flight – unacknowledged TCP bytes

# High queueing delay causes unexpected TCP behavior

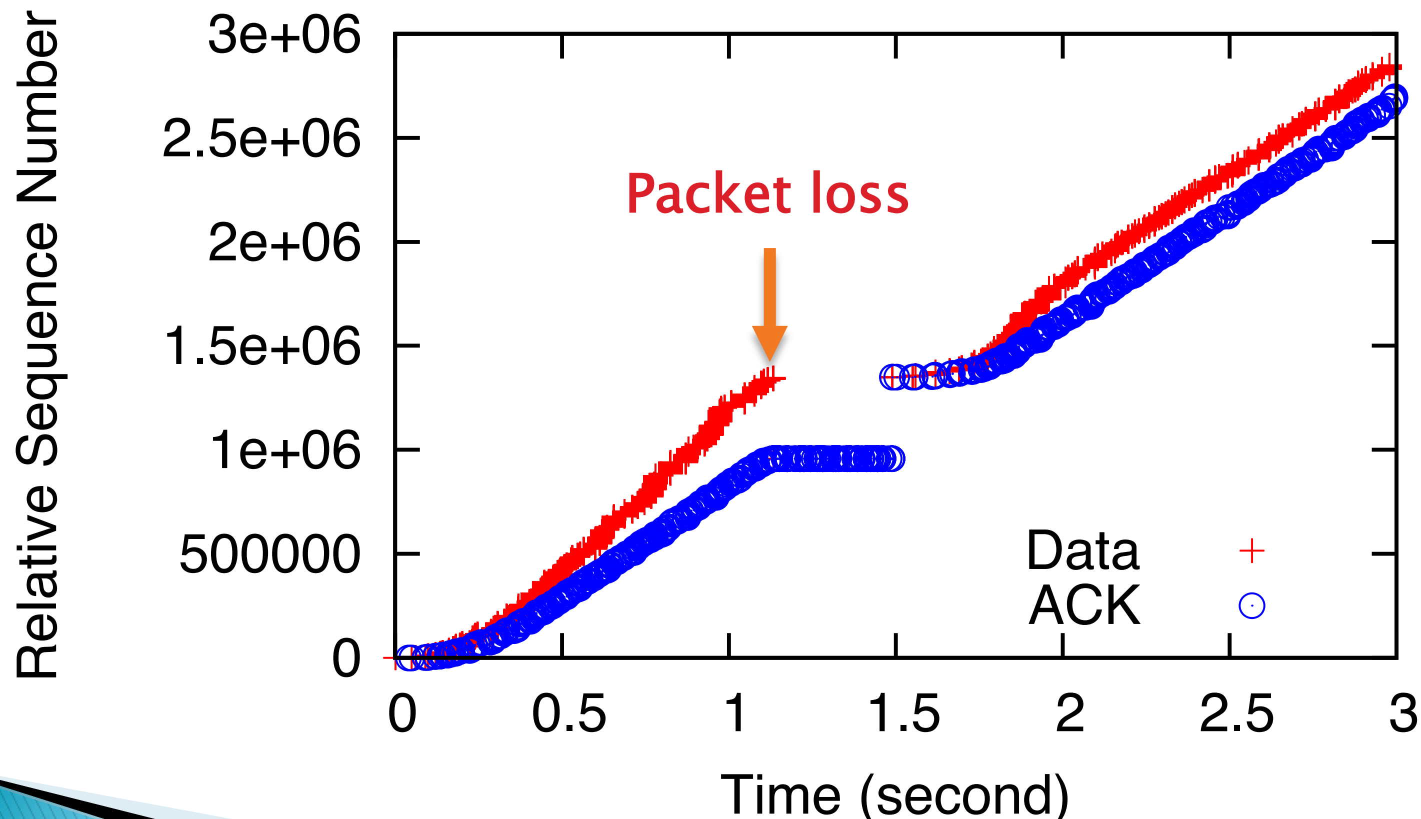
When one LTE user is downloading data …
Each dot corresponds to either a data packet or an ACK packet



# High queueing delay causes unexpected TCP behavior

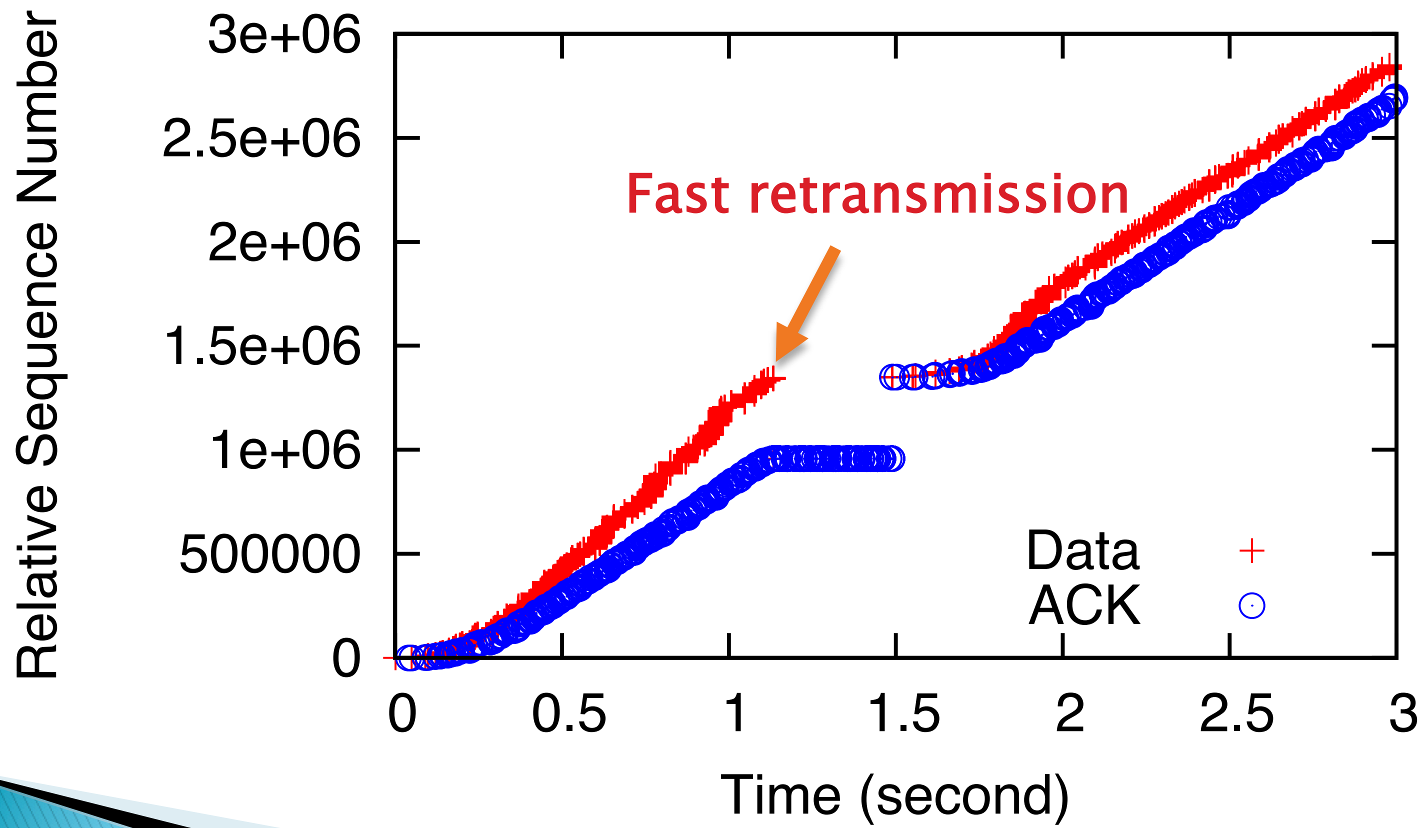


# High queueing delay causes unexpected TCP behavior



# High queueing delay causes unexpected TCP behavior

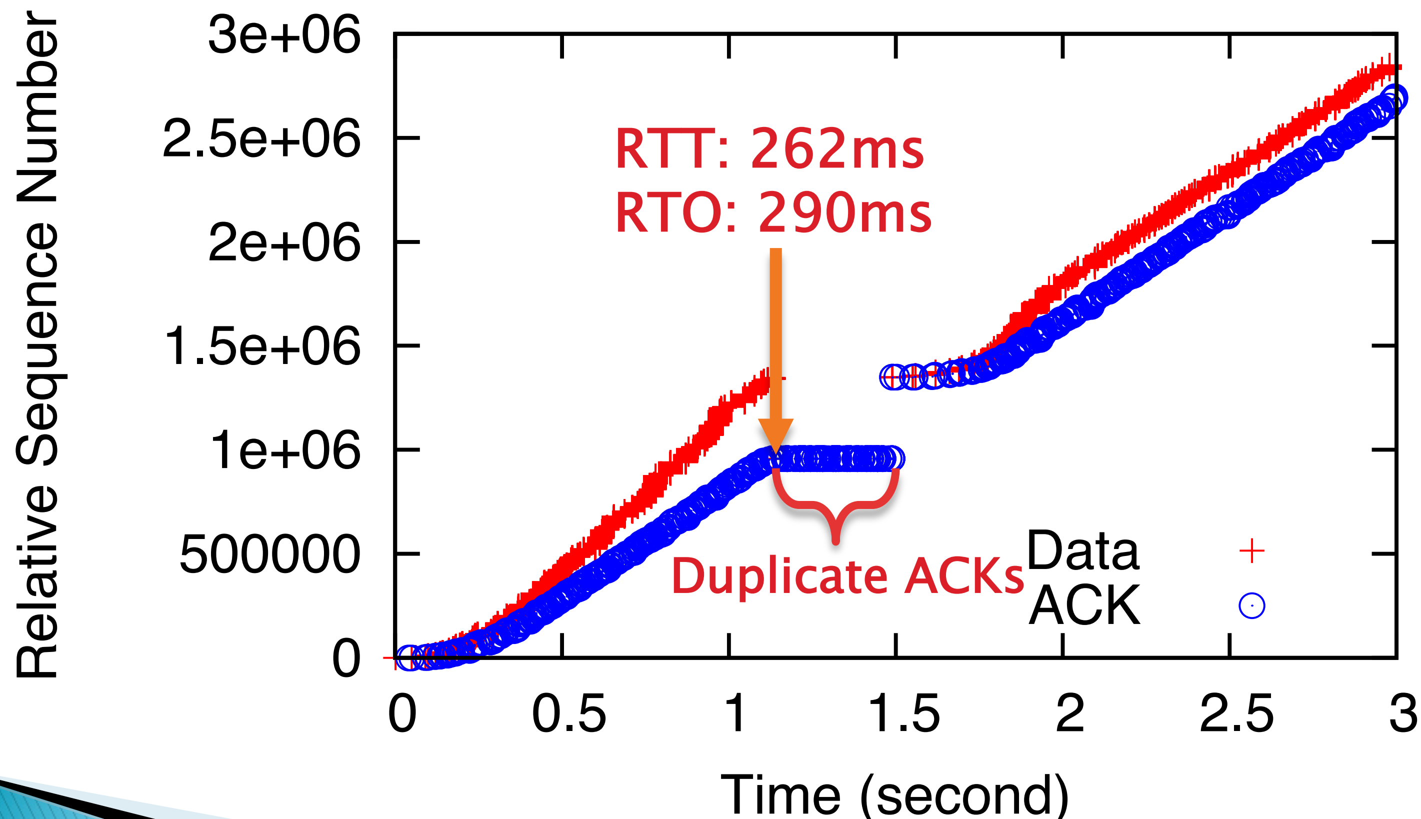
Fast retransmission allows TCP to directly send the lost segment to the receiver possibly preventing retransmission timeout



# High queueing delay causes unexpected TCP behavior

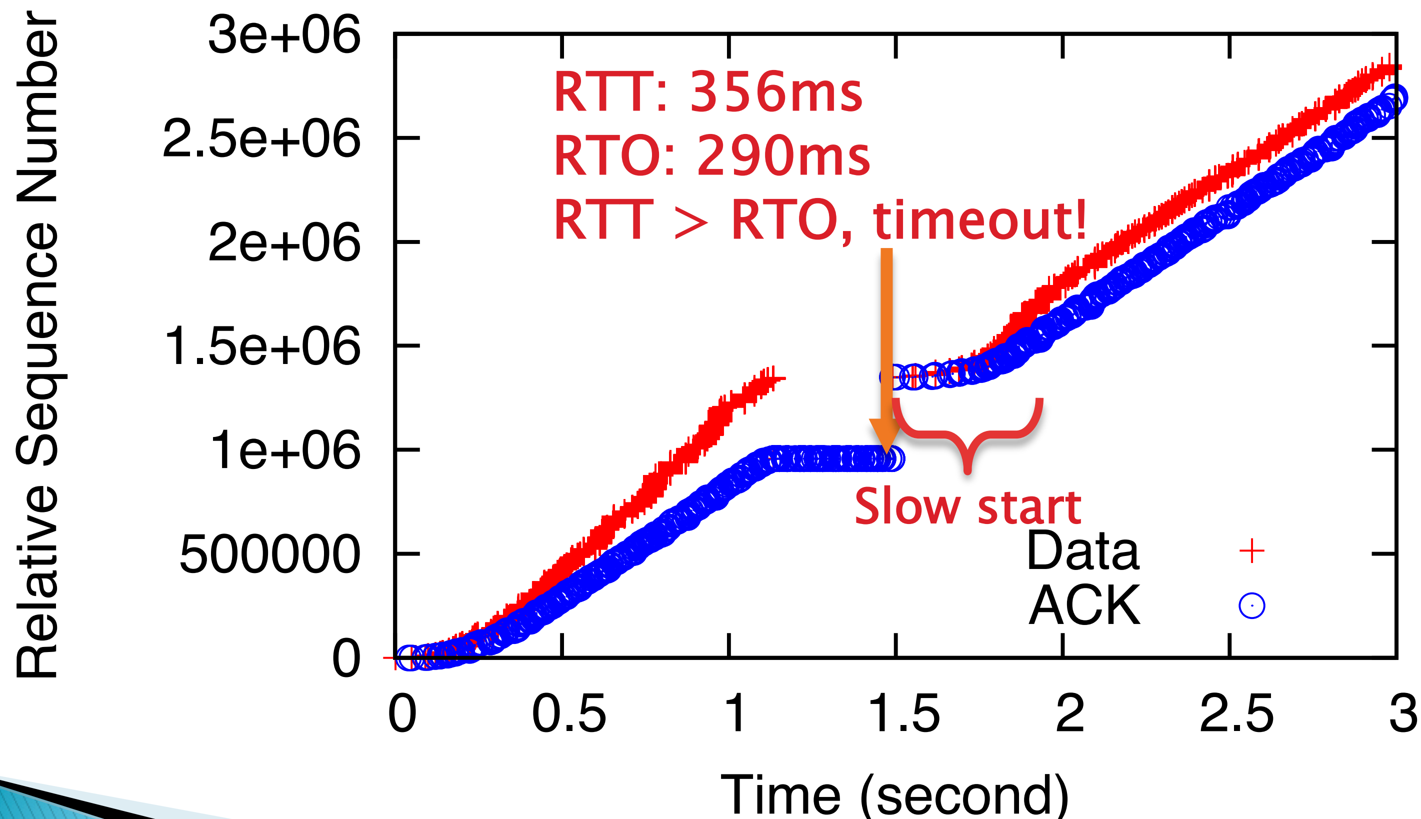
TCP uses RTT estimate to update retransmission timeout (RTO)
However, TCP does **not** update RTO based on duplicate ACKs

$$RTO \approx RTT + 4RTTVAR$$



# High queueing delay causes unexpected TCP behavior

Retransmission timeout happens, the congestion window drops to 1 segment, triggering slow start, hurting TCP performance



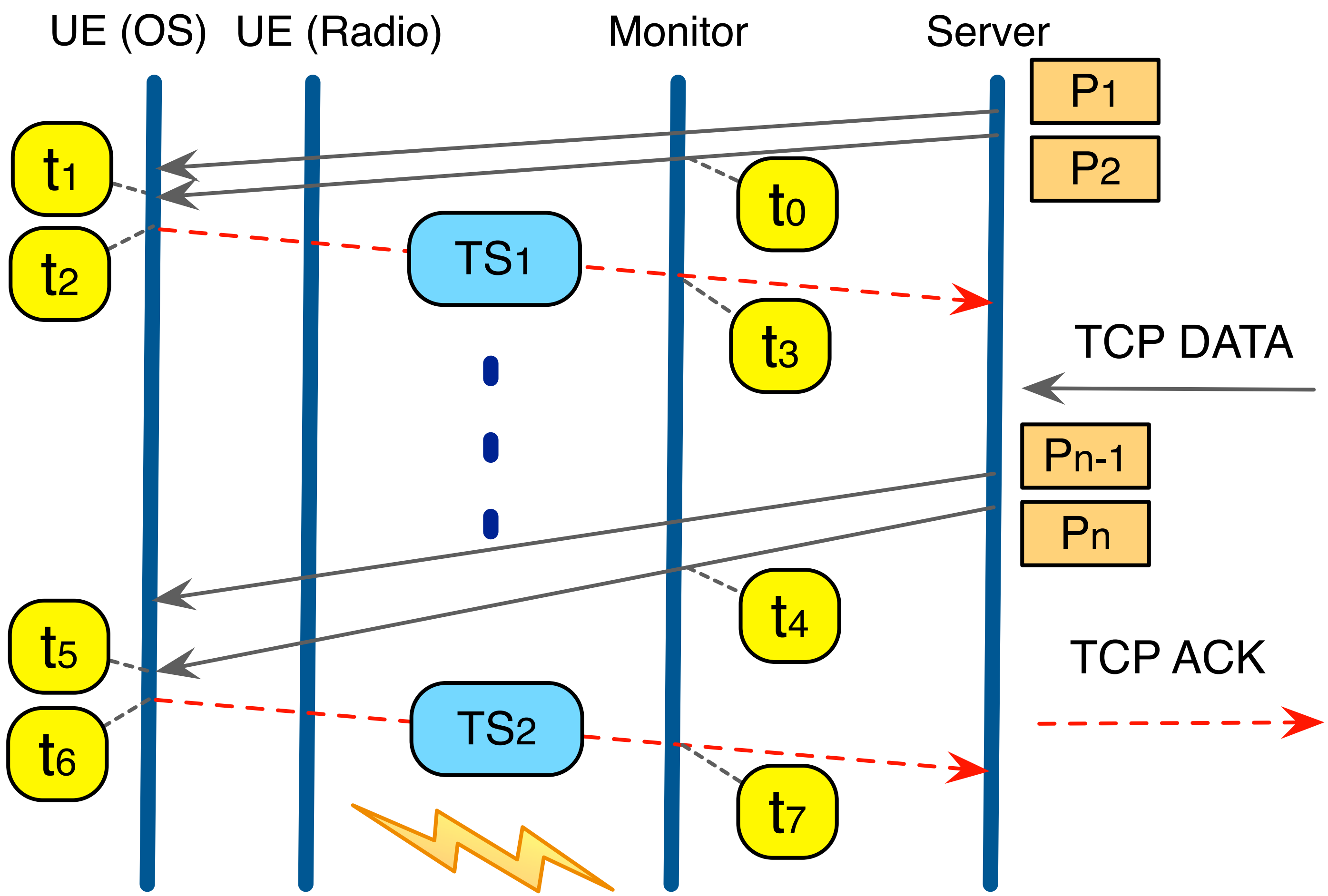
# Prevalence of the slow-start problem

- For all the large TCP flows with $\geq$ 1 packet loss, 20% of them suffer from the slow start problem, which consists of 12.3 % of all large TCP flows
- In one case, a 3-minute flow even experience 50 slow starts, resulting in an average throughput of only 2.8Mbps, while the estimated bandwidth actually larger than 10Mbps

# Passive bandwidth estimation algorithm based on server-side traces

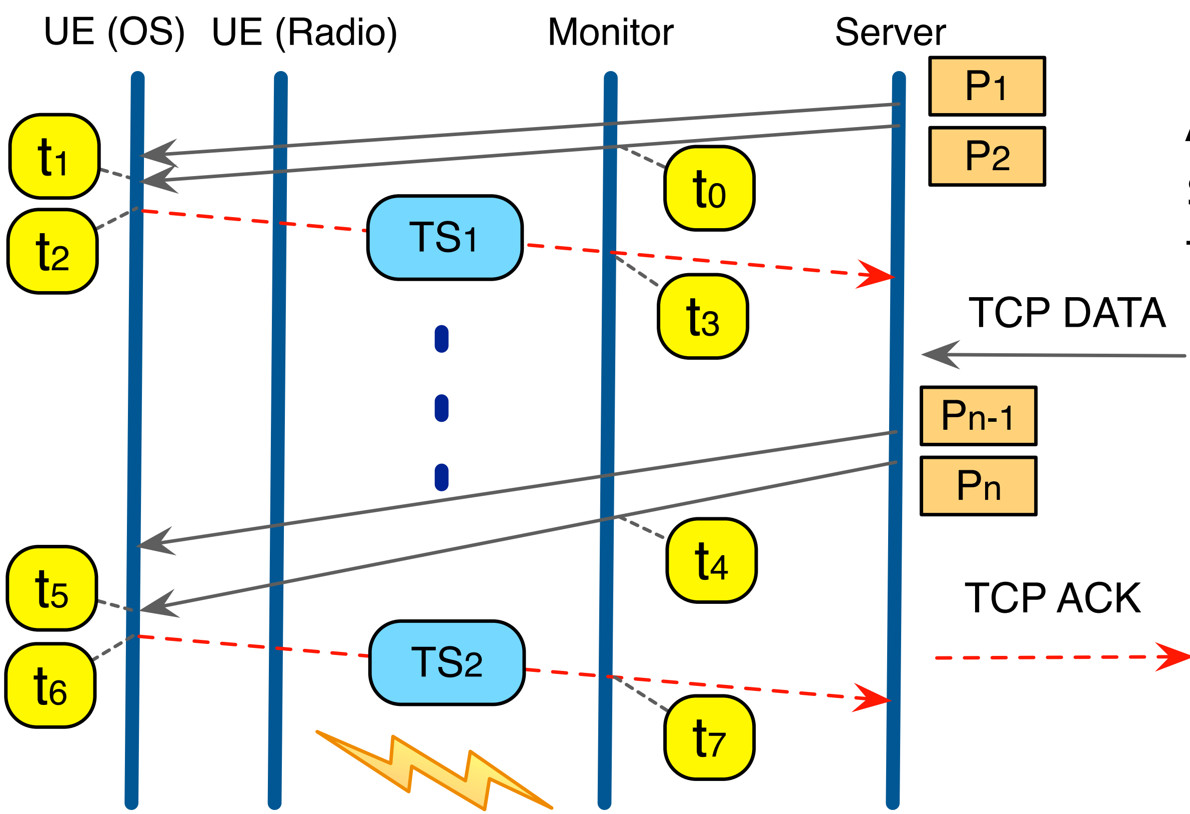
- Import for understanding the network utilization efficiency of mobile applications
- Active probing methods do not apply since the network condition is highly variable and it is impossible to launch measurement for all users

# Bandwidth estimation algorithm




Typical TCP data transfer

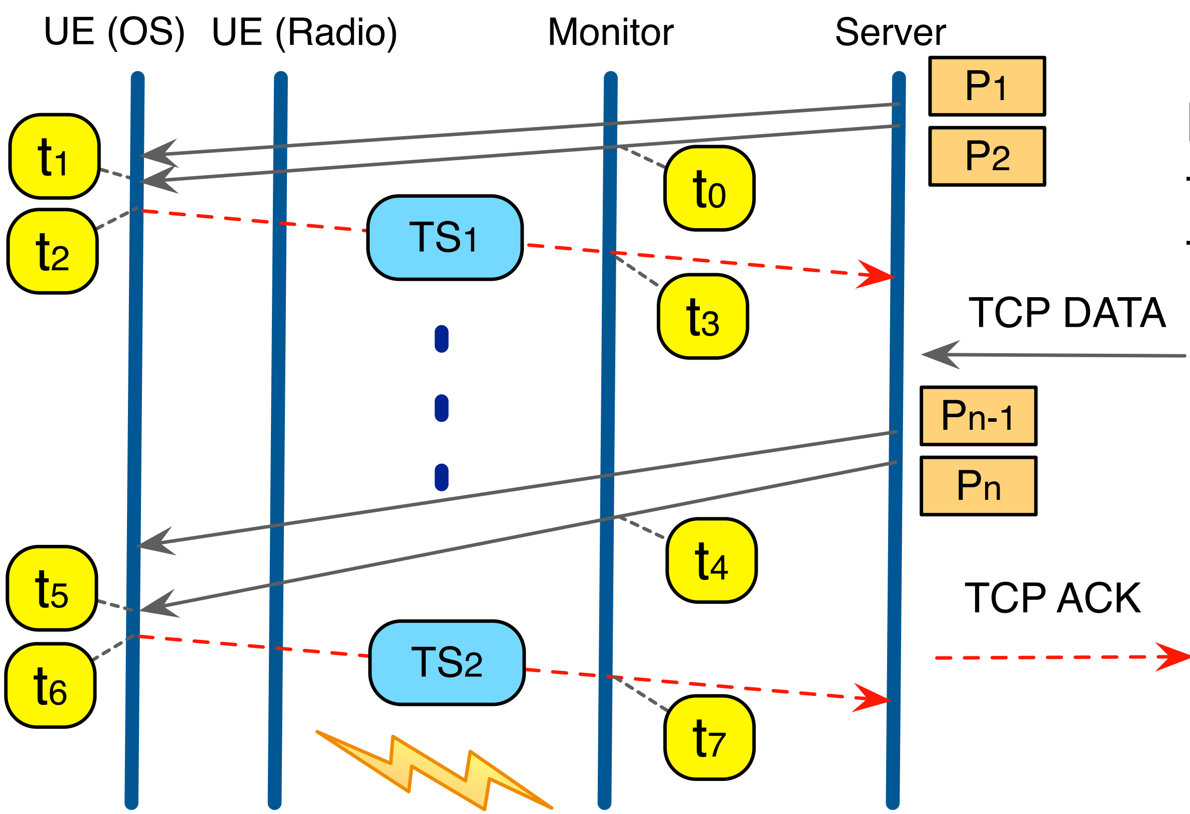
# Bandwidth estimation algorithm




Assume packet size is S, sending rate between $t_0$ and $t_4$ is

$$R_{snd} = \frac{S(n-2)}{t_4 - t_0}$$

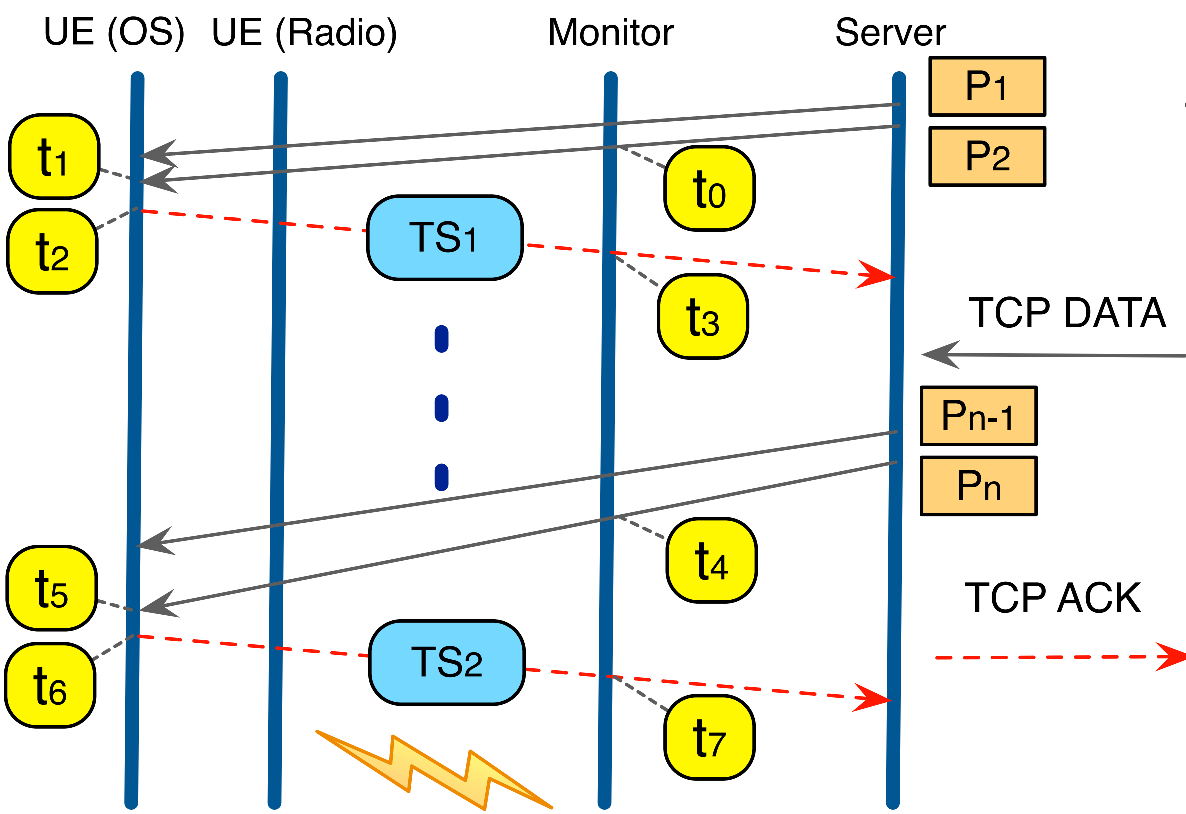
# Bandwidth estimation algorithm




From UE's perspective, the receiving rate for these n – 2 packets is

$$R_{rcv} = \frac{S(n-2)}{t_5 - t_1}$$

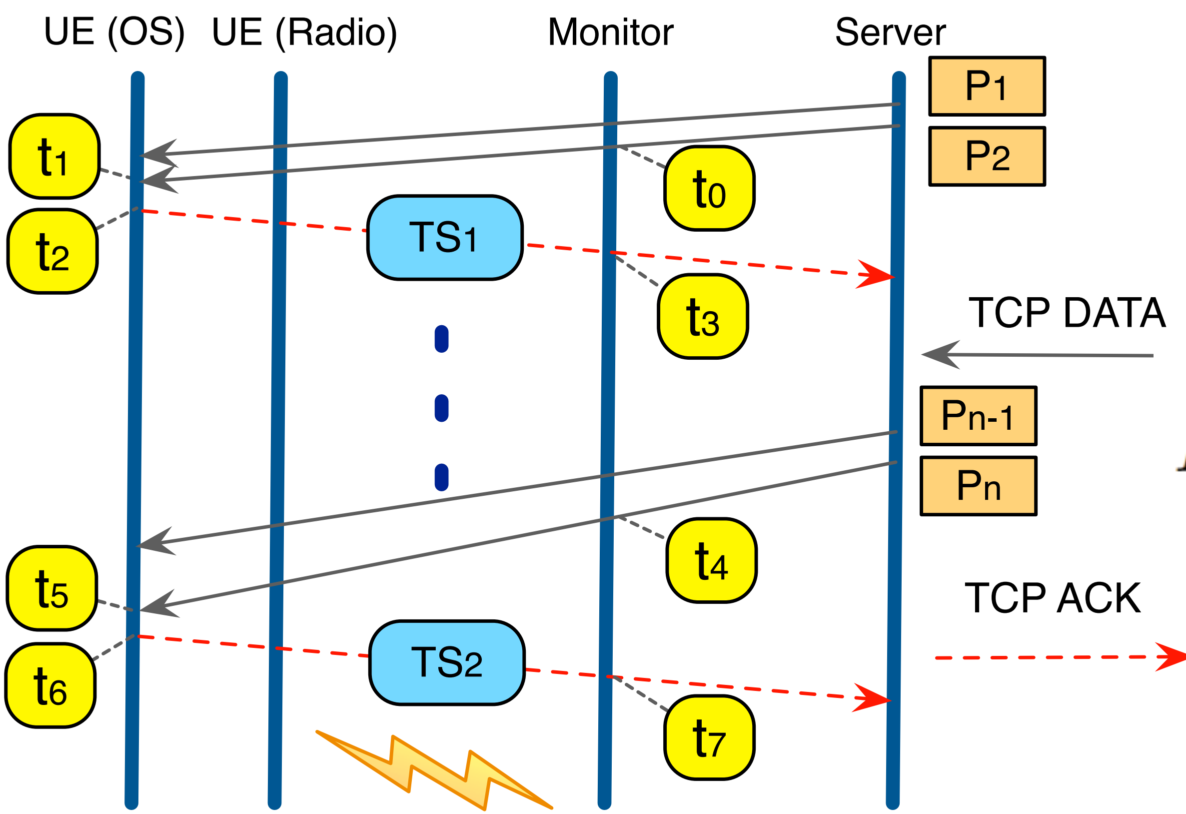
# Bandwidth estimation algorithm




Typically, $t_2$ is very close to $t_1$ and similarly for $t_5$ and $t_6$

$$R_{rcv} \approx \frac{S(n-2)}{t_6 - t_2}$$

# Bandwidth estimation algorithm




We use TCP Timestamps option to calculate $t_6 - t_2$ (G is a constant that can be inferred for each client)

$$R_{rcv} \approx \frac{S(n-2)}{G(TS_2 - TS_1)}$$

93% of the TCP flows have TCP Timestamps option enabled

# Bandwidth estimation algorithm

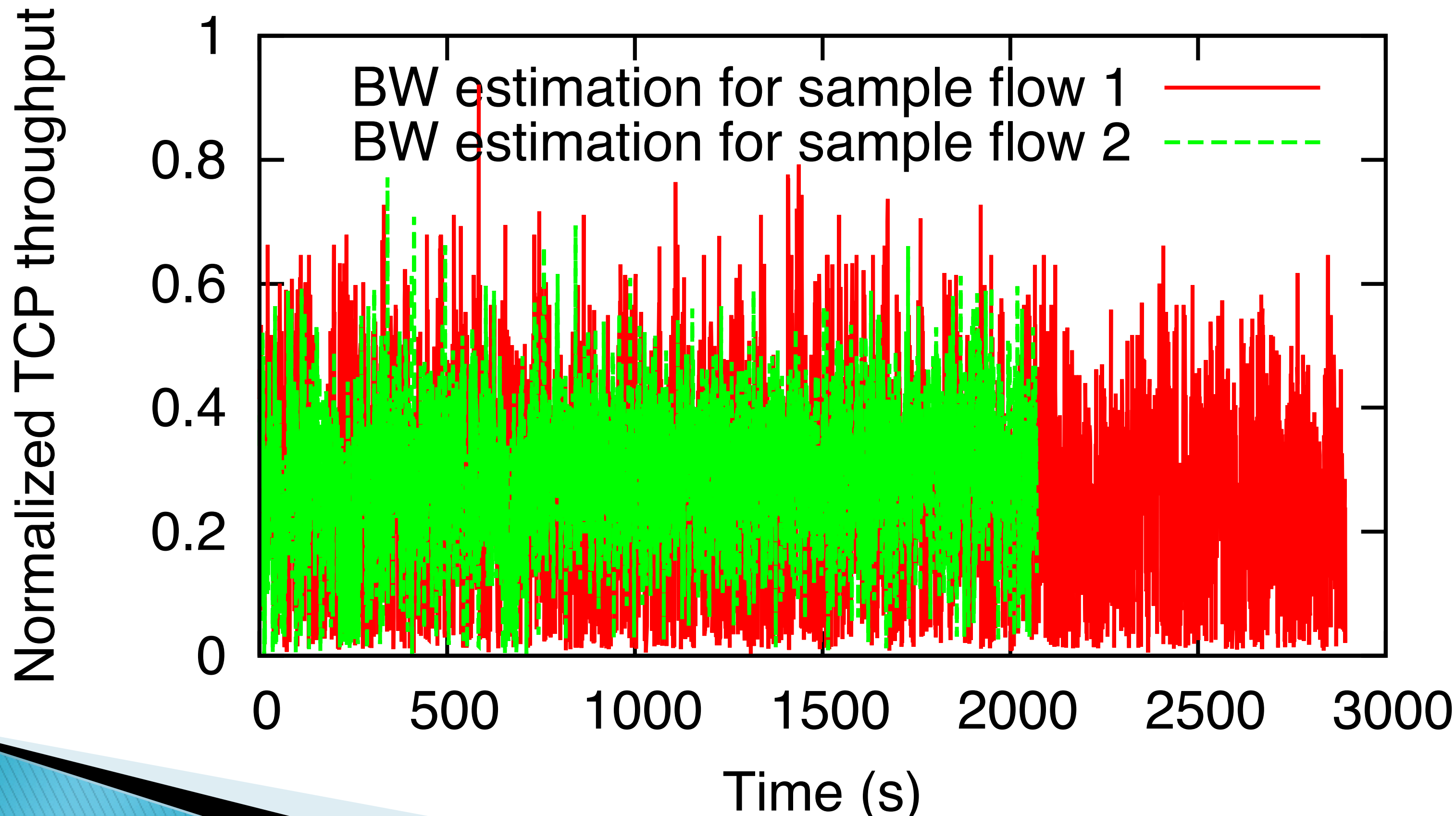
- Using a sliding window, we obtain pairs of ($R_{snd}$ , $R_{rcv}$ )
- If a list of conditions about $R_{snd}$ and TCP flow characteristics are met, $R_{rcv}$ is the estimated bandwidth
- We use local controlled experiments to validate, the error rate is < 8%
- Using this algorithm, we are able to estimate the available bandwidth for over 90% of the large downlink flows

# Bandwidth utilization by real applications in LTE

- Median of bandwidth utilization ratio is 20%, and on average, the utilization ratio is 35%
- For 71% of the large flows, the bandwidth utilization ratio is below 50%
- Reasons for underutilization
  - Small object size
  - Insufficient receiver buffer
  - Inefficient TCP behaviors

# Bandwidth estimation timeline for two sample large TCP flows

LTE network has highly varying available bandwidth



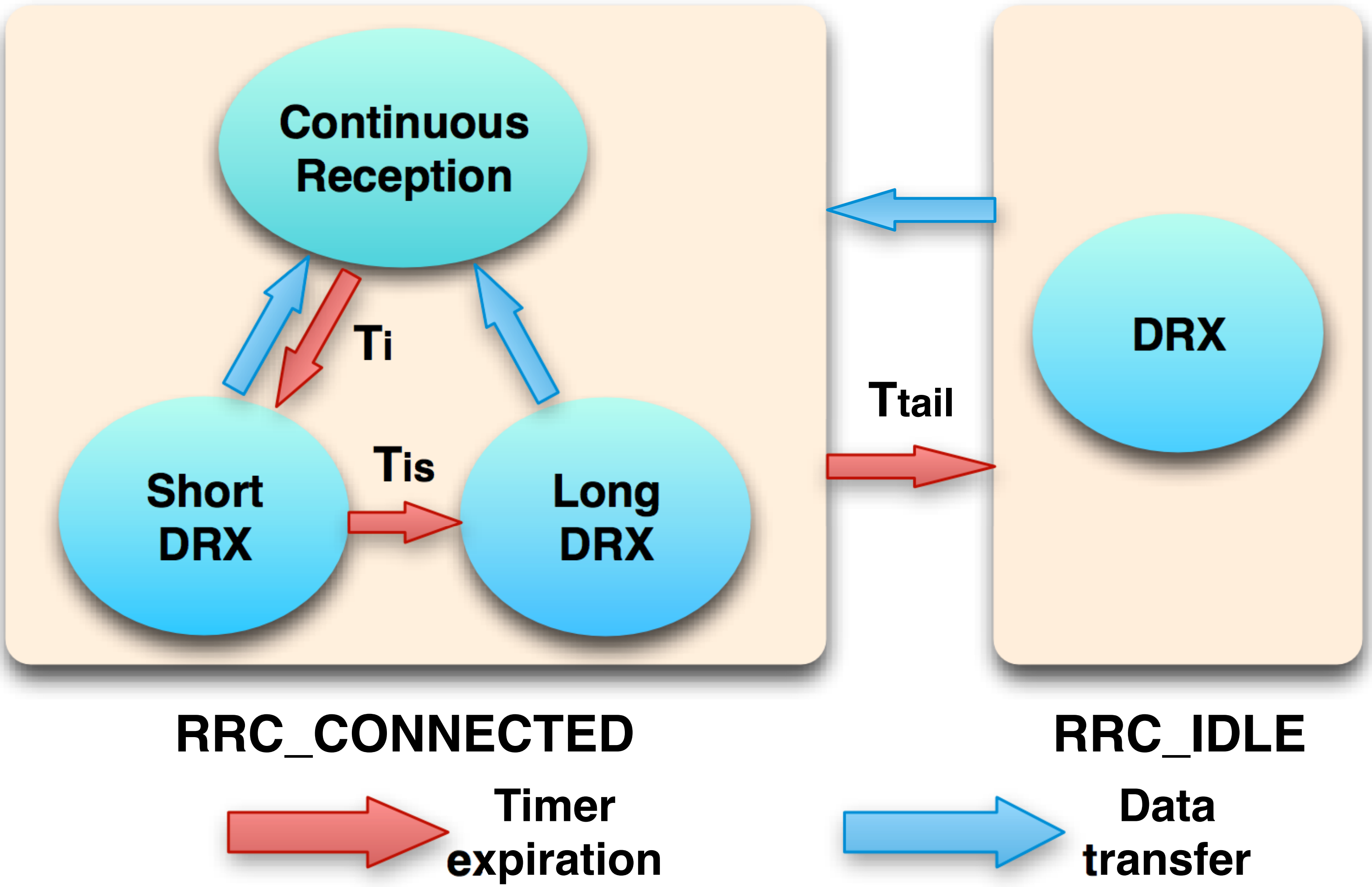
# LTE bandwidth variability, RTT and TCP performance

- Under small RTTs, TCP can utilize over 95% of the varying available bandwidth
- When RTT exceeds 400~600ms, the utilization ratio drops to below 50%
- For the same RTT, higher variation leads to lower utilization
- Long RTTs can degrade TCP performance in the LTE networks

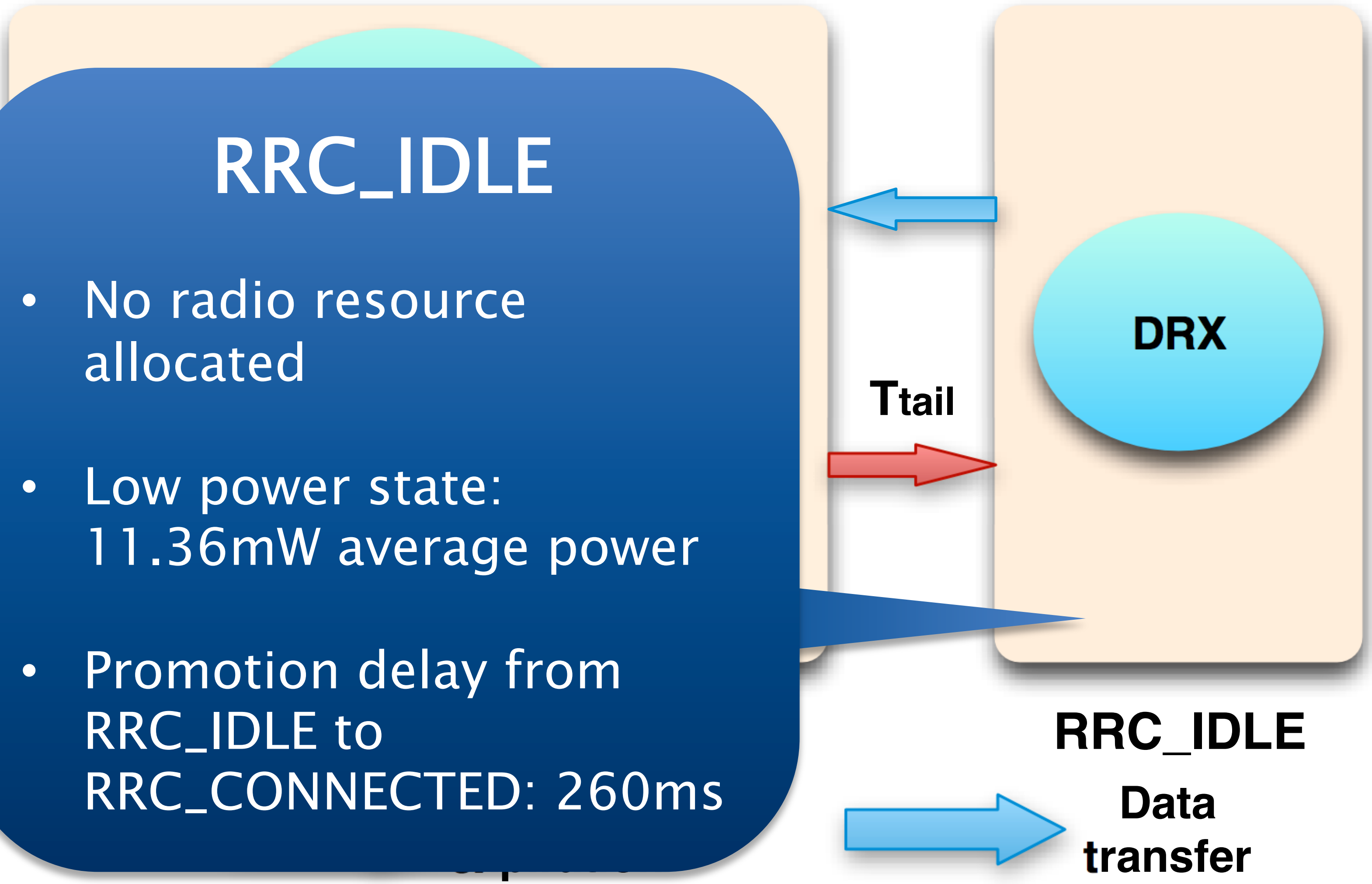
# Roadmap

- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

# Radio Resource Control (RRC) state transitions in LTE



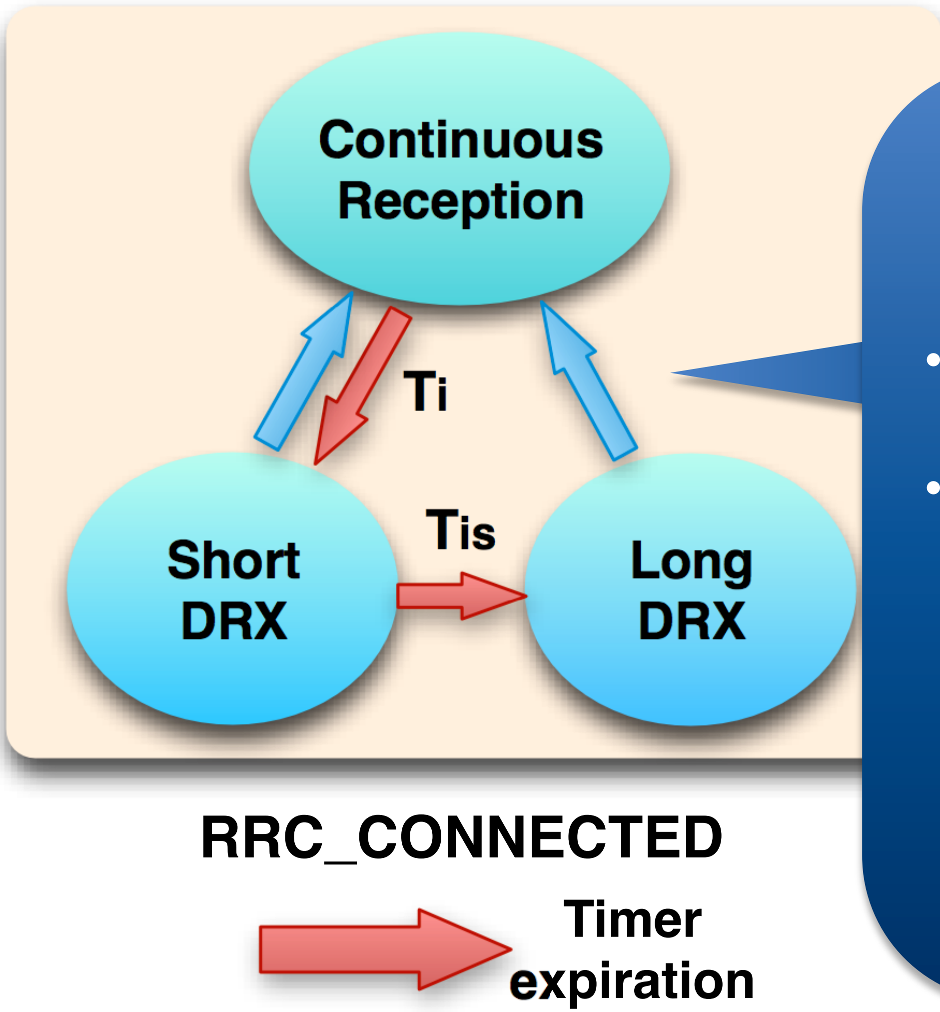
# RRC state transitions in LTE




## RRC_IDLE

- No radio resource allocated
- Low power state: 11.36mW average power
- Promotion delay from RRC_IDLE to RRC_CONNECTED: 260ms

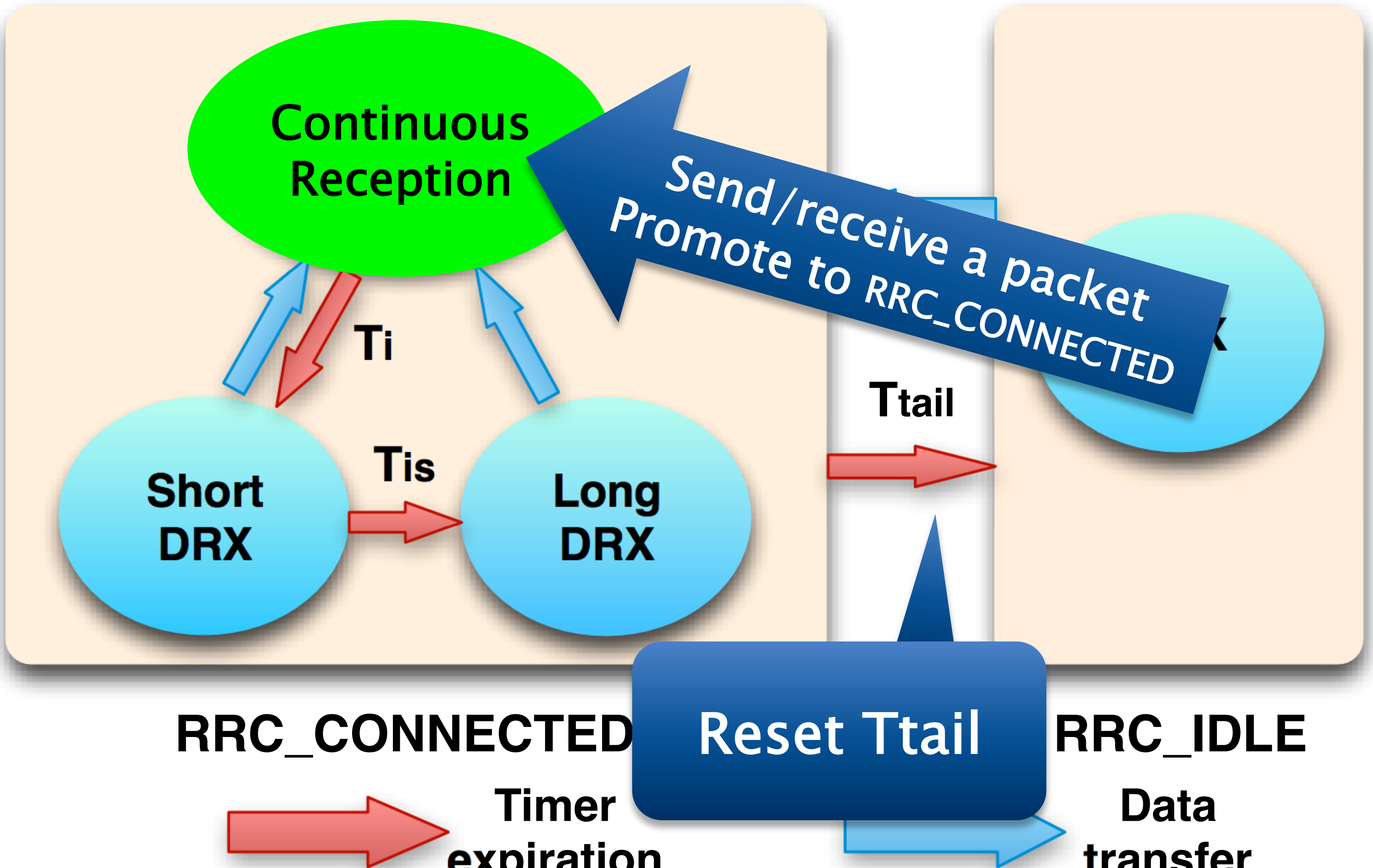
# RRC state transitions in LTE



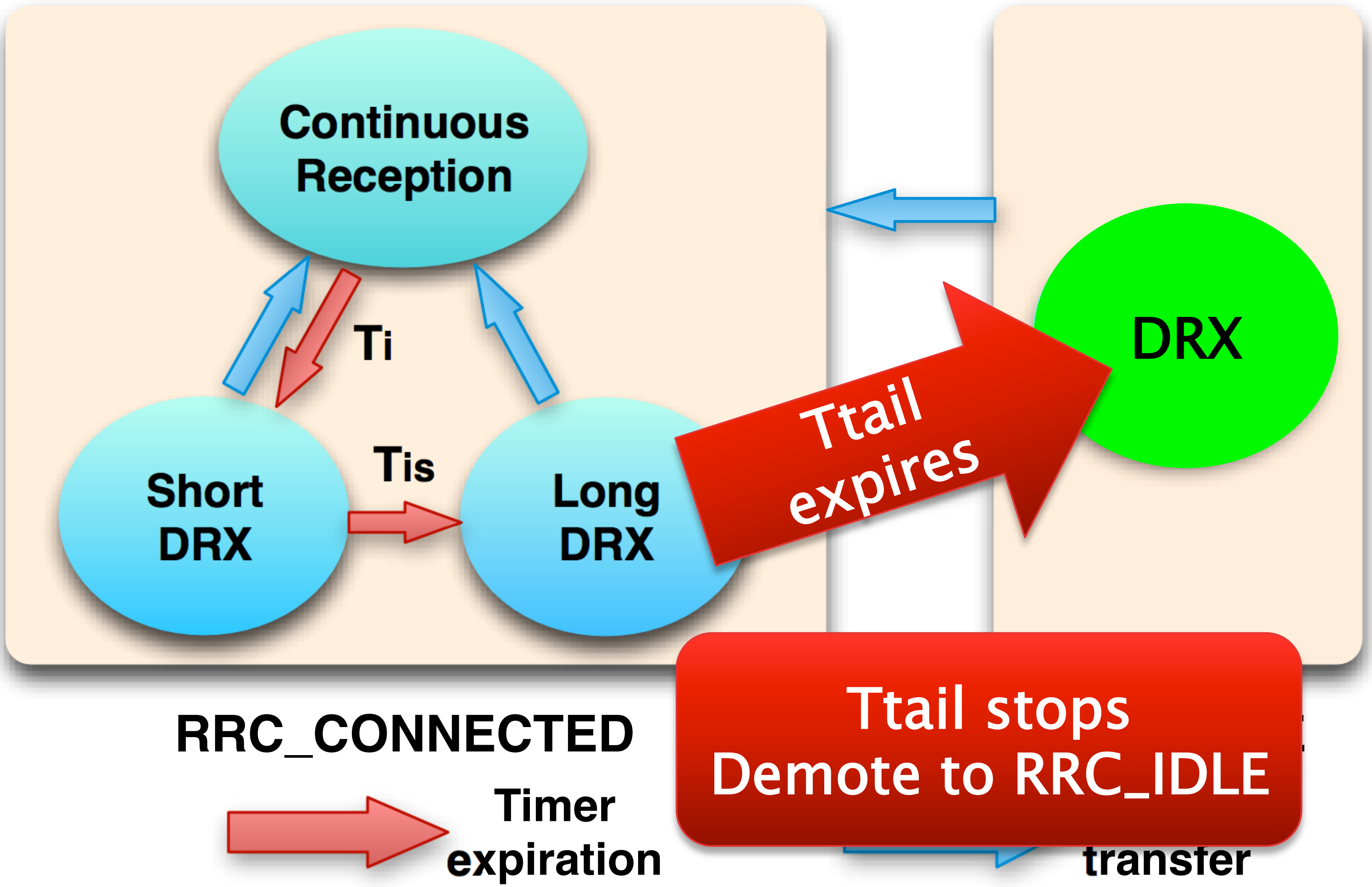

## RRC_CONNECTED

- Radio resource allocated
- Power state is a function of data rate:
  - 1060mW is the base power consumption
  - Up to 3300mW transmitting at full speed

# RRC state transitions in LTE

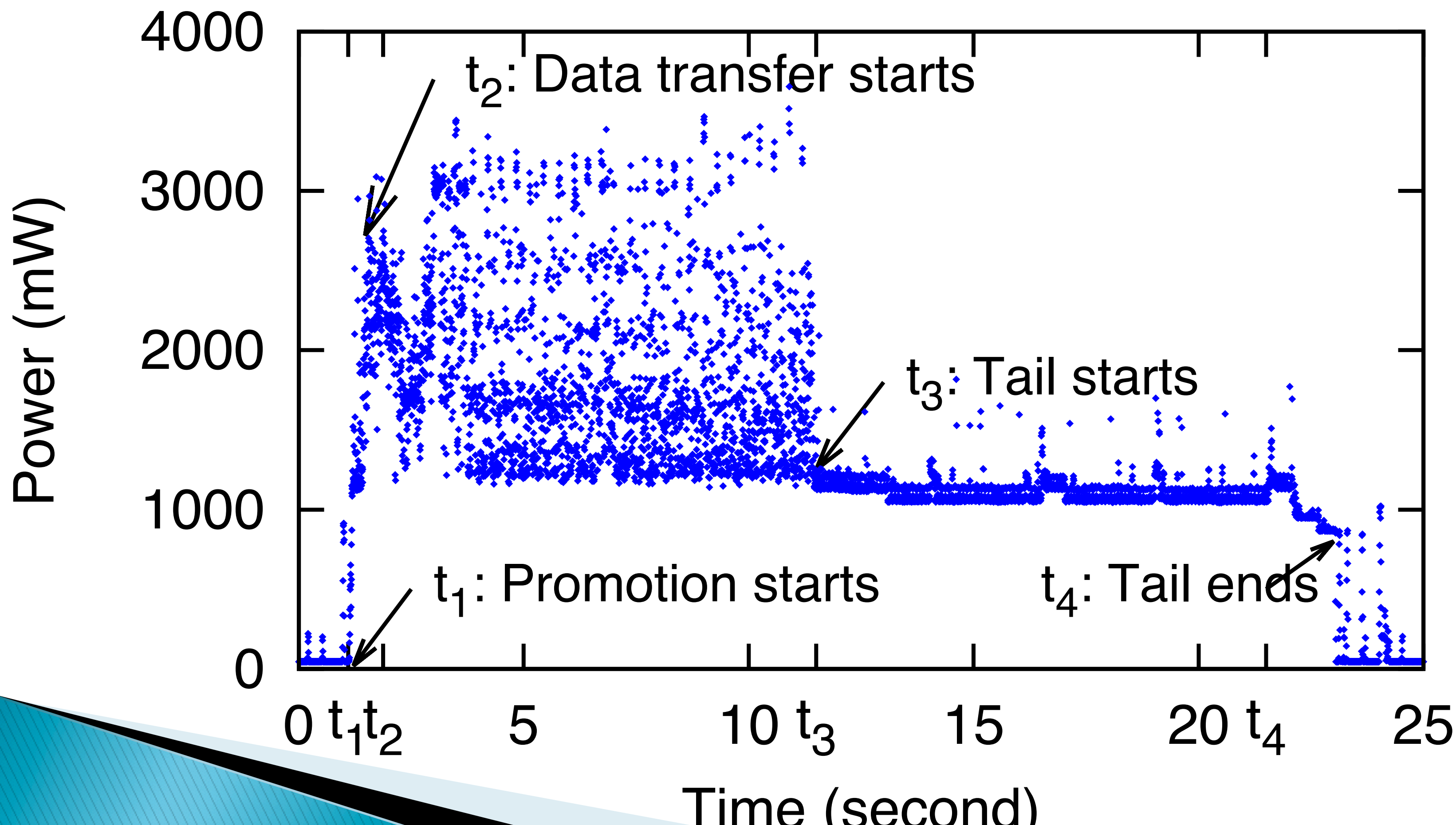


# RRC state transitions in LTE



# Power trace of RRC state transitions

The data points are sampled and DRX in RRC_CONNECTED tail is not obvious due to the low sampling rate



# LTE power model (Verizon)

- Measured with a LTE phone and Monsoon power meter, averaged with repeated samples

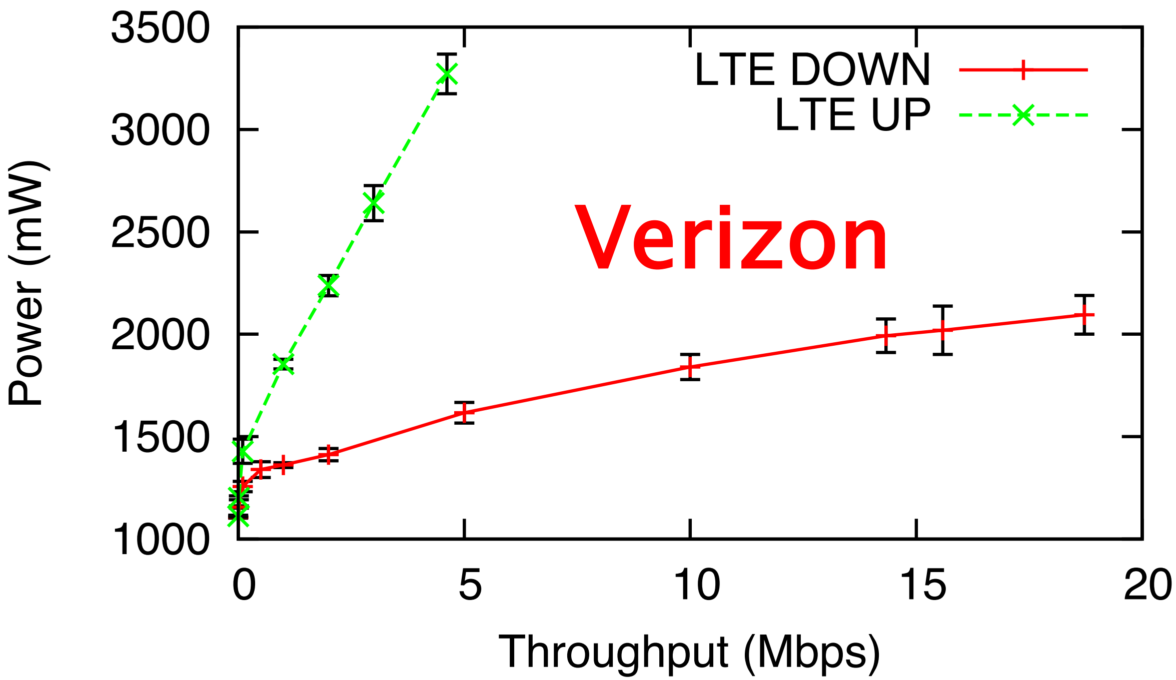
| | Power* (mW) | Duration (ms) | Periodicity (ms) |
|---|---|---|---|
| Screen off (base) | 11.4±0.4 | N/A | N/A |
| Screen 100% on | 847.2±2.7 | N/A | N/A |
| LTE promotion | 1210.7±85.6 | $T_{pro}$: 260.1±15.8 | N/A |
| LTE Short DRX On in **RRC_CONNECTED** | 1680.2±15.7 | $T_{on}$: 1.0±0.1 | $T_{ps}$: 20.0±0.1 |
| LTE Long DRX On in **RRC_CONNECTED** | 1680.1±14.3 | $T_{on}$: 1.0±0.1 | $T_{pl}$: 40.1±0.1 |
| LTE Off Duration in **RRC_CONNECTED** | $\boxed{1060.0±3.3}$ | $\boxed{T_{tail}: 11576.0±26.1}$ | N/A |
| LTE DRX On in **RRC_IDLE** | 594.3±8.7 | $T_{oni}$: 43.2±1.5 | $T_{pi}$: 1280.2±7.1 |

# Power model for data transfer

- A linear model is used to quantify instant power level:
  - Downlink throughput $t_d$ Mbps
  - Uplink throughput $t_u$ Mbps

$$P = \alpha_u t_u + \alpha_d t_d + \beta$$

**Data transfer power model**

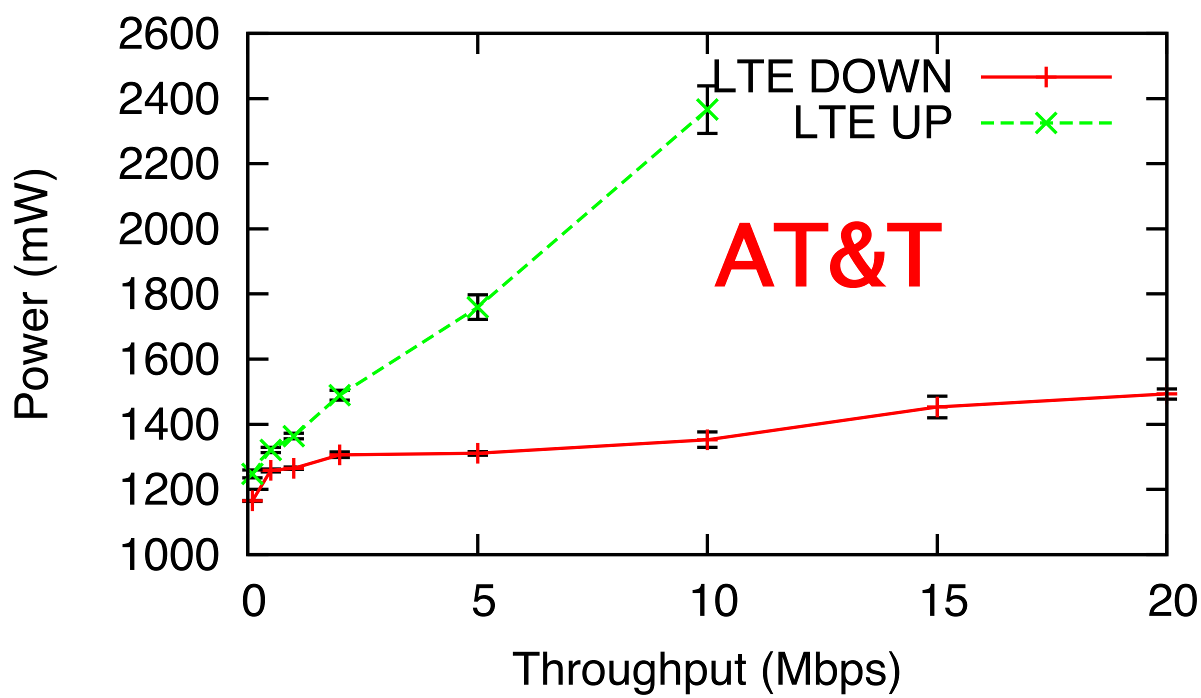



## transfer

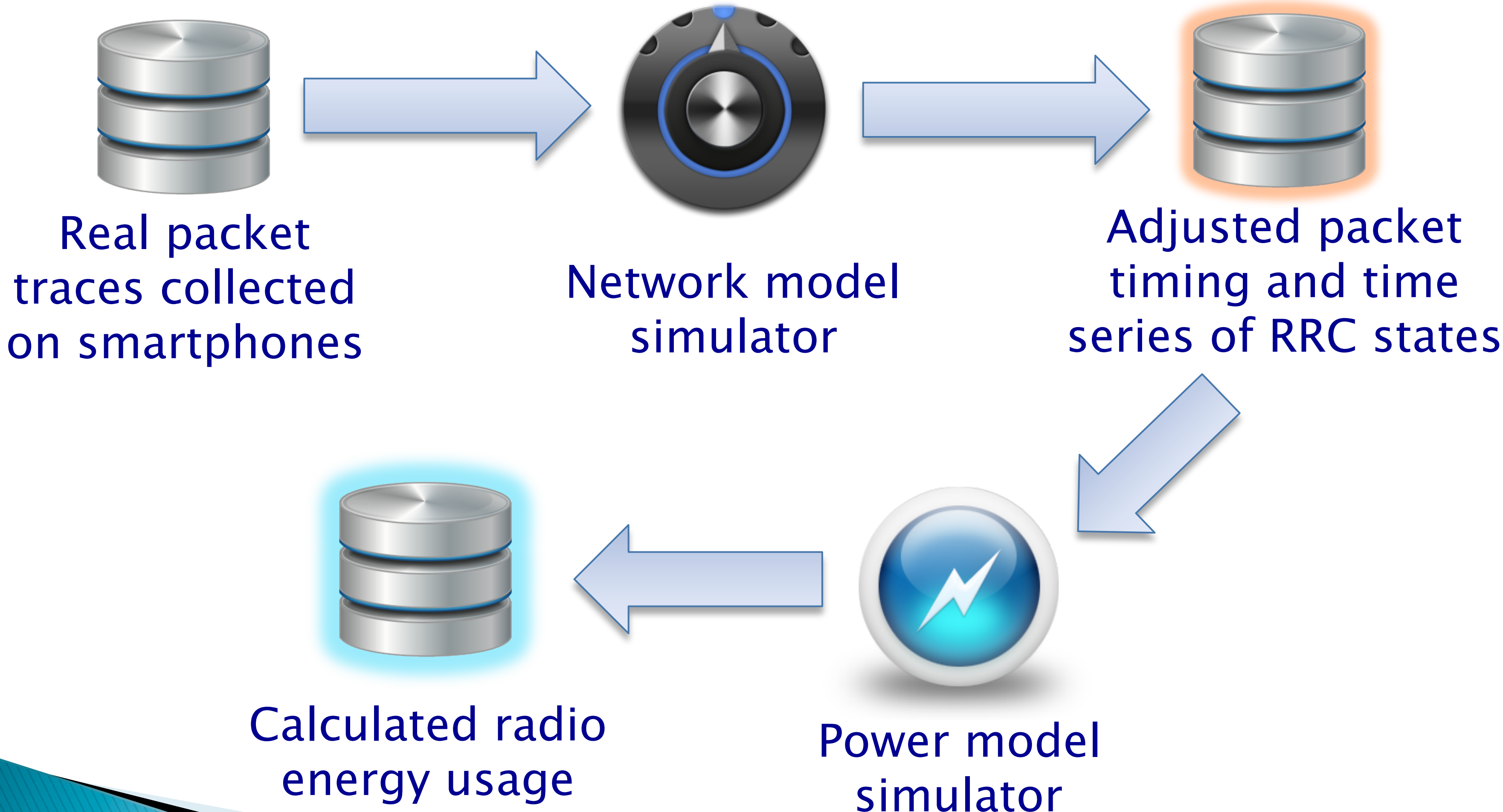
tify instant

$P =$

**Data**



# Trace-driven modeling methodology



# Power model validation


Real applications running

Power meter directly measure

Packet traces + Trace-driven analysis

< 6% error rate in evaluations with real applications

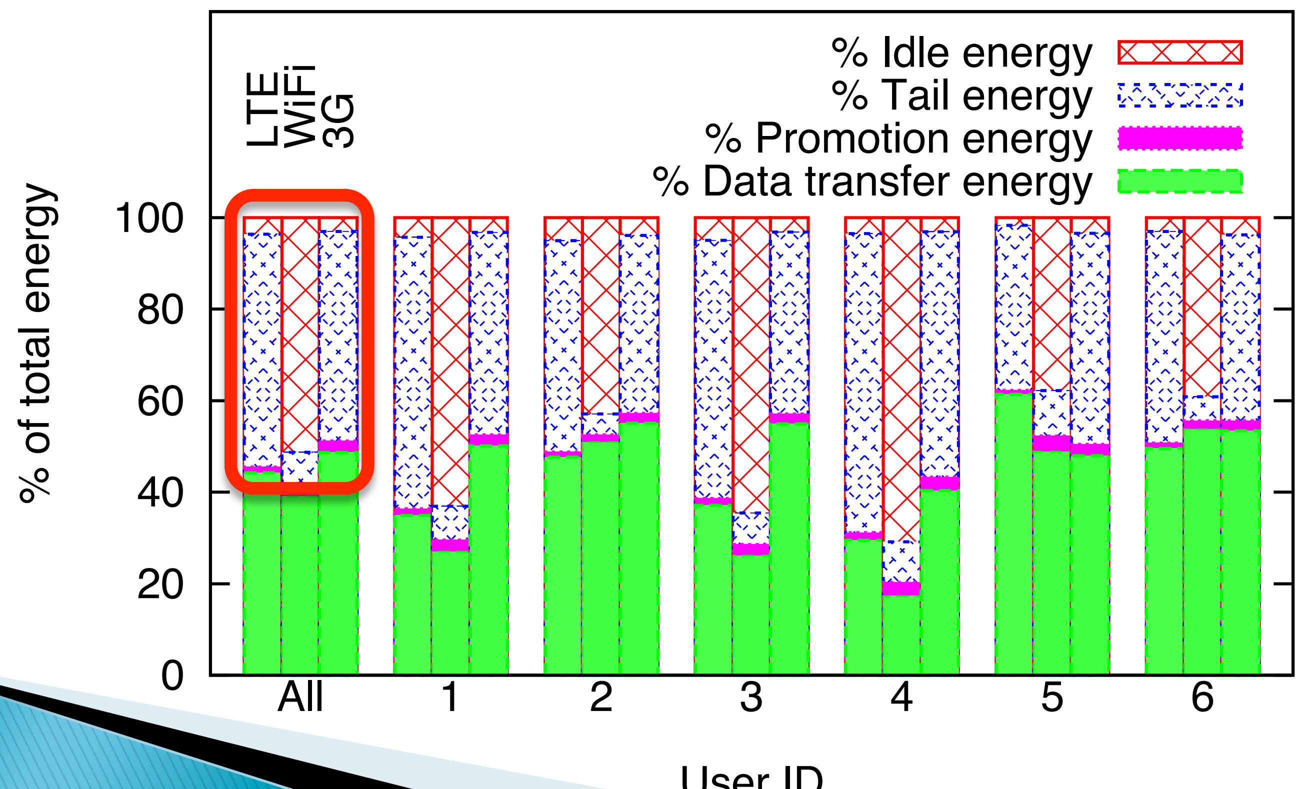
# UMICH data set

- 20 voluntary smartphone users for seven months
  - Undergraduate and graduate students from 8 departments at University of Michigan
- Data collected (152GB in total)
  - Full *tcpdump* packet traces
  - Process name for sending/receiving each packet
  - Screen status logger (1Hz)

# Energy consumption break down

- Tail energy dominates LTE energy consumption, similar to 3G

The total energy for different networks and users is normalized to be 100%

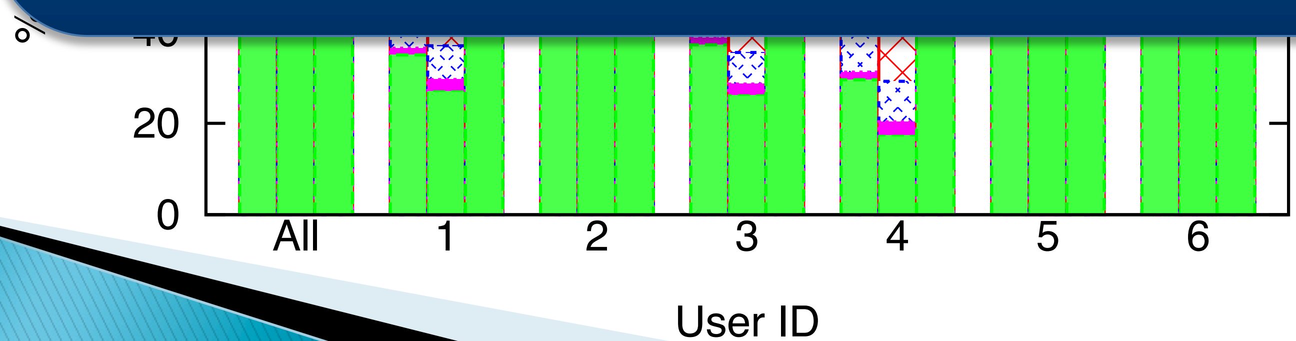


# Energy consumption break down

- Tail energy dominates LTE energy consumption, similar to 3G

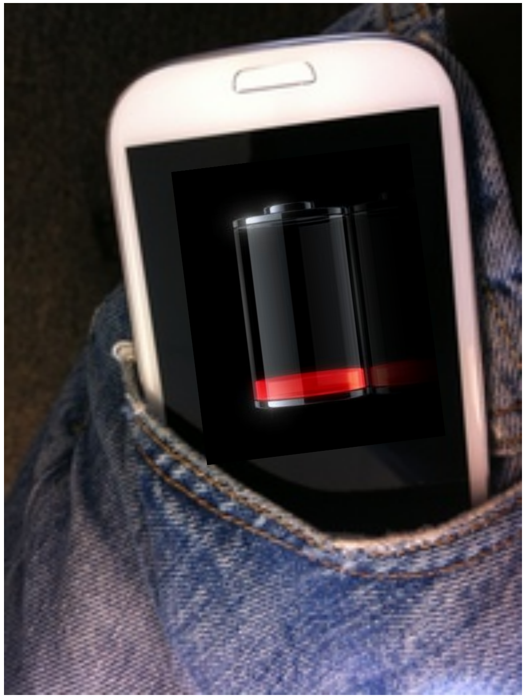
The total energy for different networks and users is normalized to be 100%

The **tail problem** is the key factor for LTE's high energy consumption, similar to 3G networks



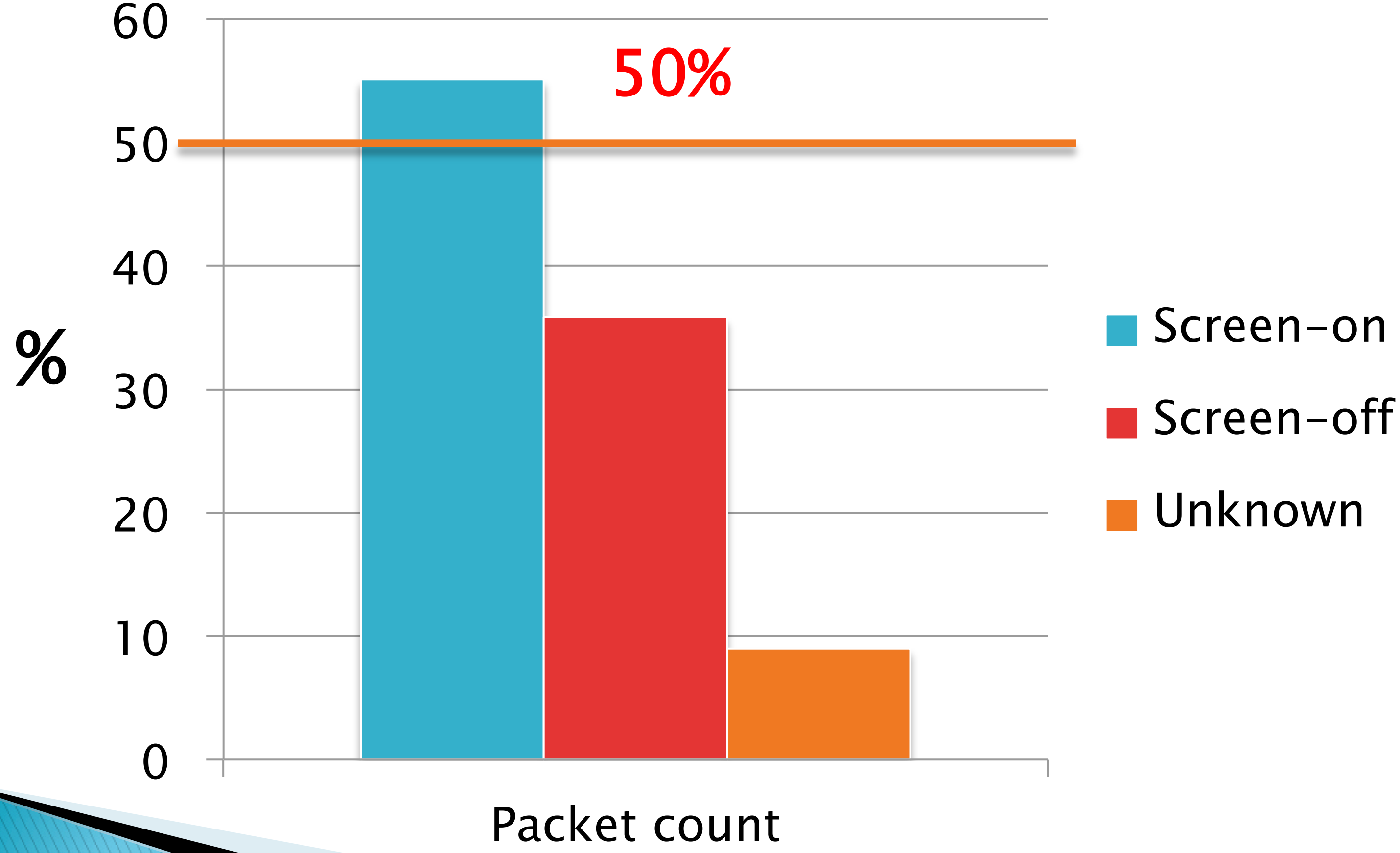
# Screen-off traffic characterization

- Smartphone screen is switched on and off often (>50 times/ day/user)
- Screen status is a good heuristic for determining whether the user is actively interacting with the device



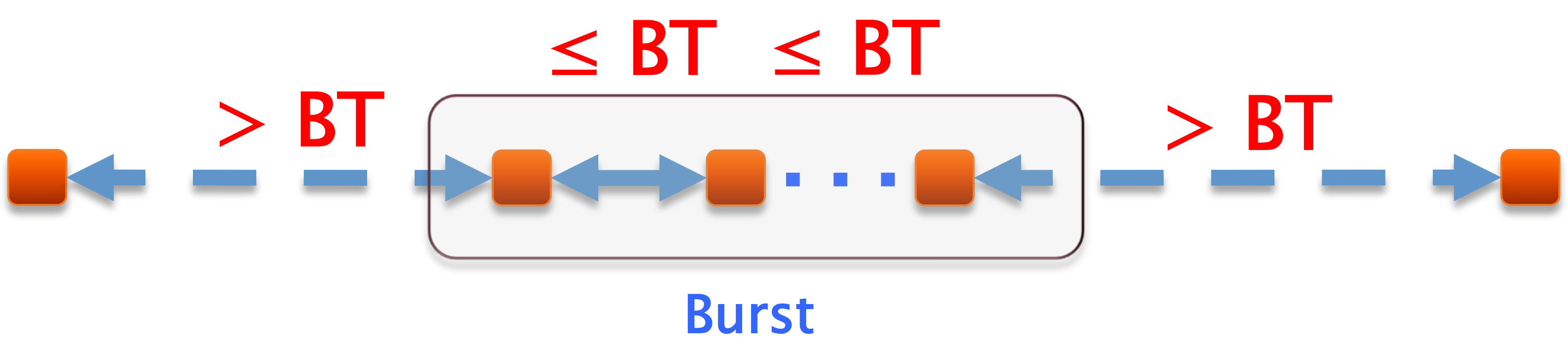
# Breakdown of packet count based on screen status

*Unknown group: 9% of all packets, due to users accidentally terminating screen status logger*



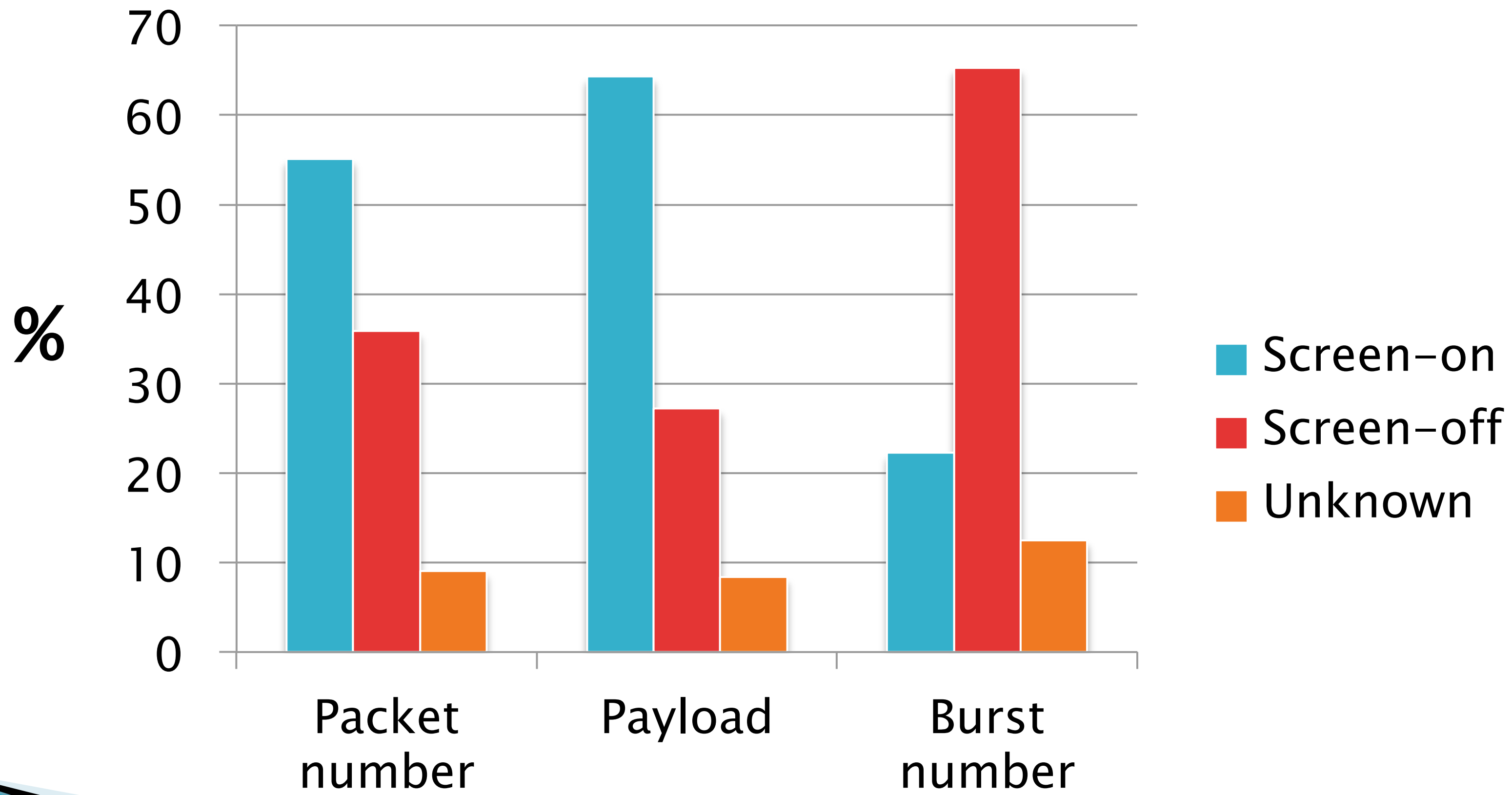
# Definition of a "*burst*"

- A burst is a sequence of packets with inter-packet time $\leq$ **BT**, and leading/trailing gap $>$ **BT**
- **BT** is burst threshold selected empirically based on network RTTs, e.g. BT = 2s



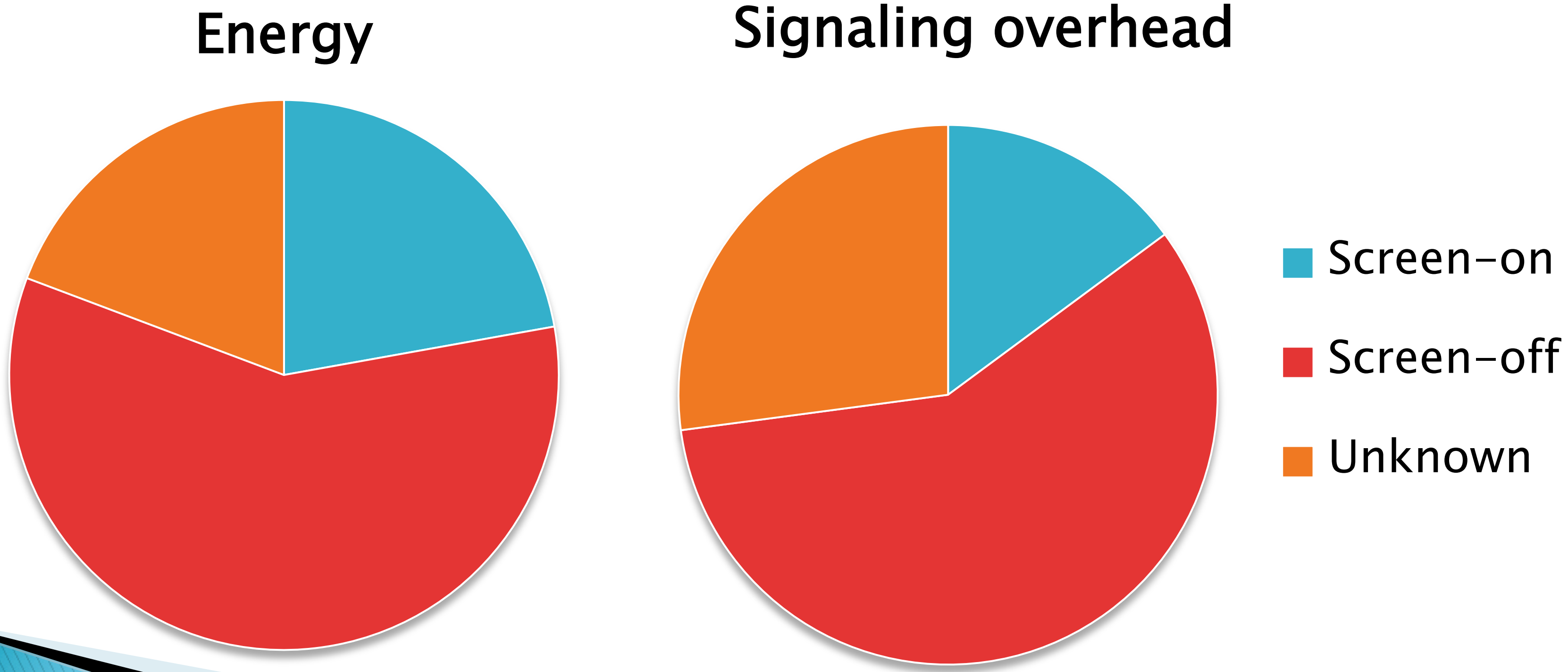
# Screen-on and screen-off traffic comparison

- Screen-off traffic has *less* packets/payload, but *more* bursts, which are *smaller* and *shorter*



# Does screen-off traffic matter for radio resource and energy?

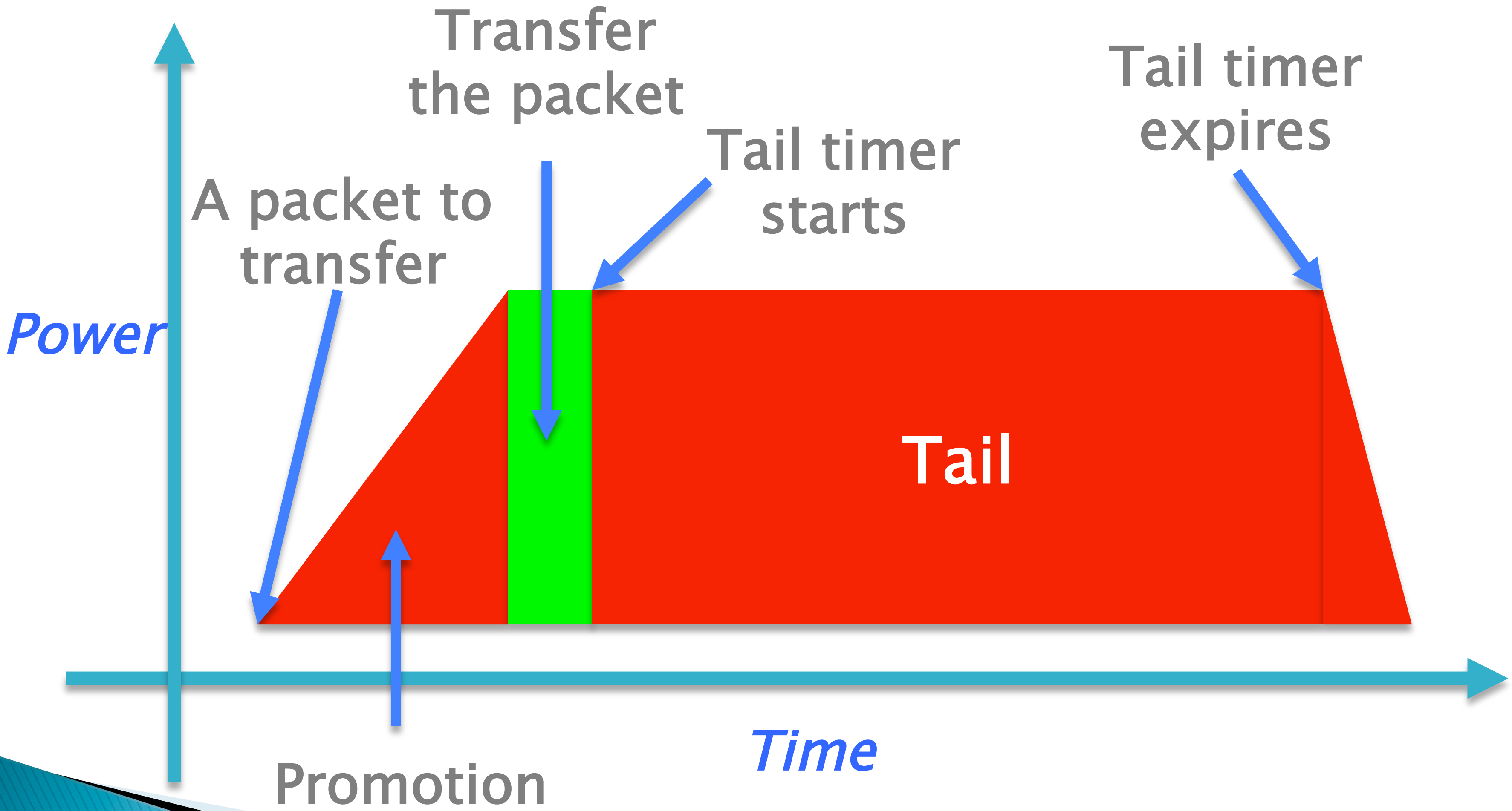
- Yes! Actually, screen-off traffic has higher impact than screen-on, though with less packets



# Roadmap

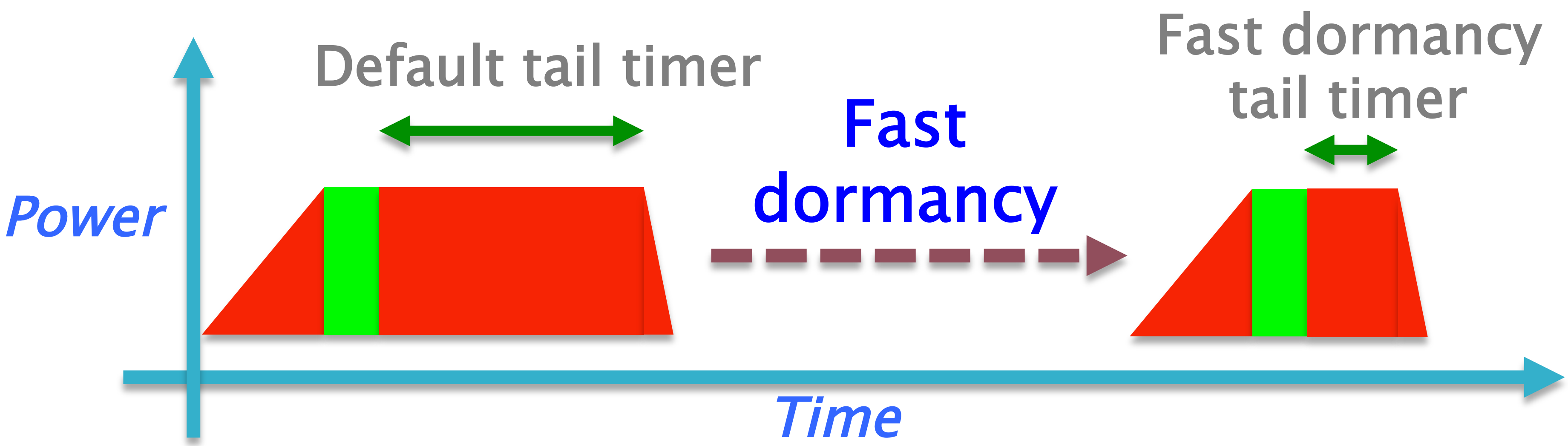
- MobiPerf: characterizing cellular network performance
- Anatomizing smartphone application performance
- An In-depth Study of LTE
- Characterizing Radio Energy Usage
- RadioProphet: Optimizing Smartphone Energy and Radio Resource

# Radio Resource Control (RRC) State Machine



# Fast Dormancy (FD)

- Fast dormancy reduces the tail length by actively notifying the network for early demotion

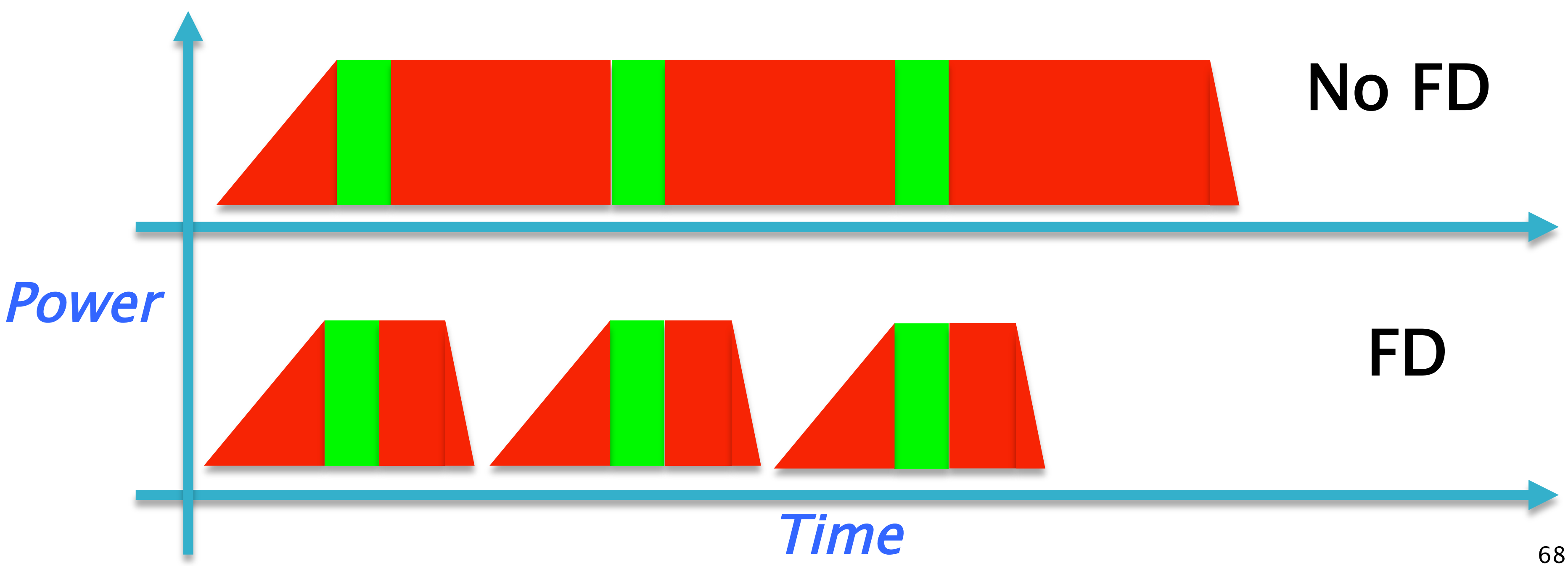




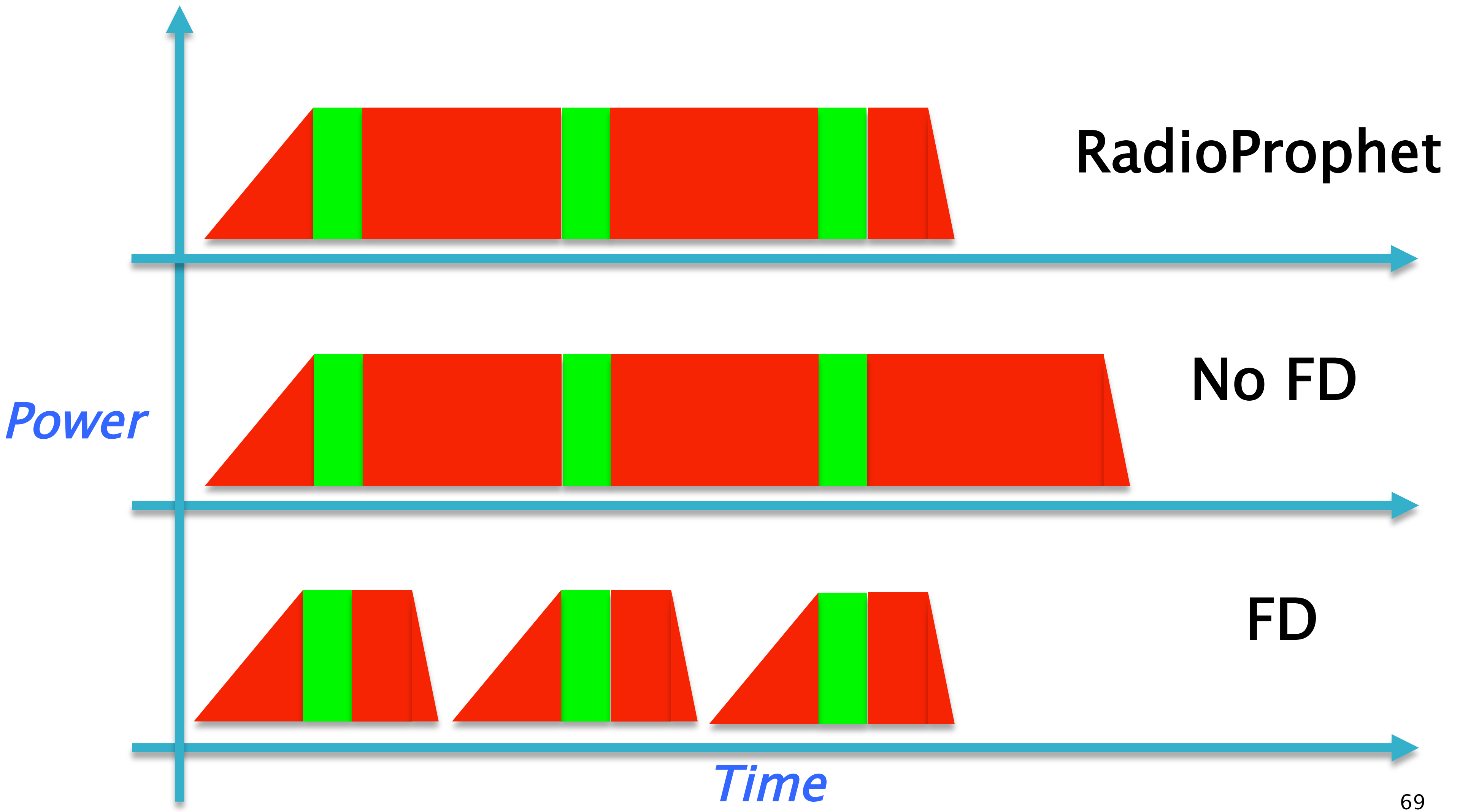
67

# Problem with the Naïve Fast Dormancy

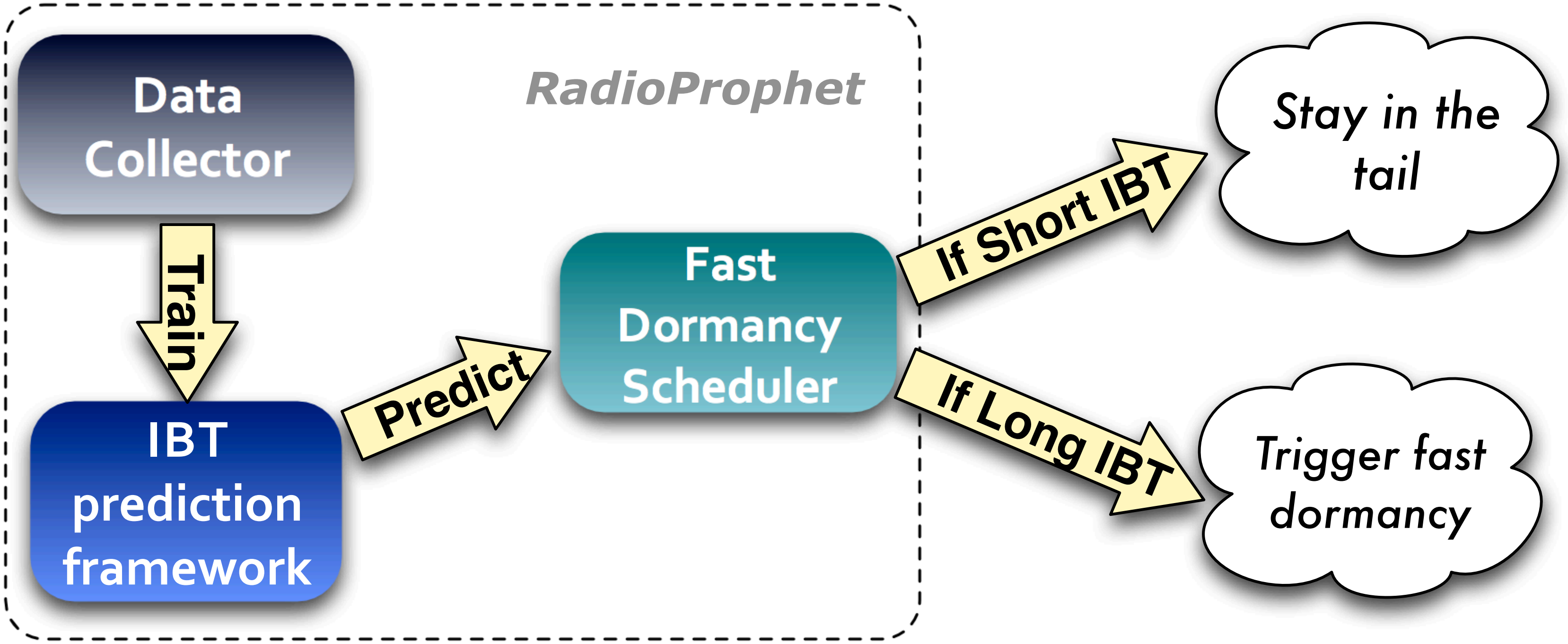
- Extra promotion overhead
  - UE energy and delay overhead
  - Signaling overhead
    - 32 radio messages exchanged between UE and cell tower for promotion



# RadioProphet: Intelligently Invoke FD



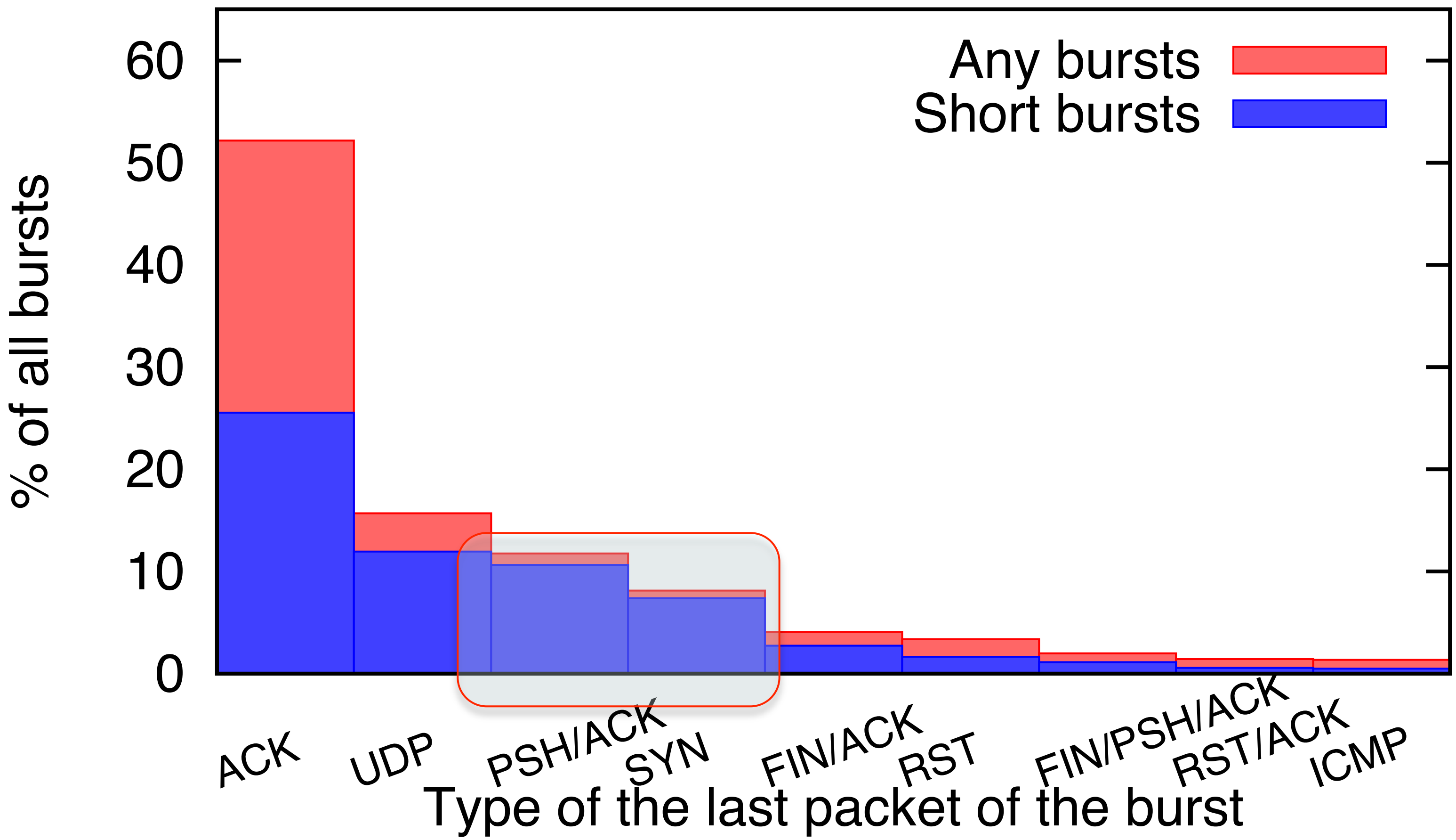
# RadioProphet: Intelligent Radio Resource Deallocation for Cellular Networks



# Inter-Burst Time (IBT) Prediction

- Traffic features
  - Packet direction, port, packet length, protocol, TCP flags, application ID
- Machine learning algorithms
  - Naïve Bayes
  - Regression tree
  - Classification tree
  - Ensemble boost
  - etc.

# Example Feature Selection – TCP Flags



# Implementation and Evaluation

- Data collector
  - Modified *tcpdump* cross-compiled for Android
  - Collect traffic for prediction
  - $<$ 2% CPU overhead (energy overhead $<$ 1% tail energy)
- IBT prediction framework
  - Weka machine learning library ported to Android
  - Average model training time 200ms, prediction time 0.1ms (energy overhead $<$ 1% tail energy)
  - Best accuracy for ***all users*** in the UMICH data set is 86%
- Overall performance
  - Energy saving: *59%* of radio energy (*20% ~ 30%* total energy)
  - Signaling overhead reduction: *57%* comparing with default FD

# Summary and future agenda

- Optimizing cellular network and power resource usage for mobile applications is important and requires further study
- MobiPerf
  - A more general mobile measurement platform to better benefit the research community in both the tool and the data
  - Longitudinal study of the evolution of cellular networks
- RadioProphet
  - To consider more TCP- and application-layer context to improve prediction accuracy
  - Public deployment and OS integration

# Thank you!