Data Driven Networking

Sachin Katti
Platform Lab Review, Feb 2017

Stanford University & Uhana
Can we learn the control plane of the network?

Control & Orchestration Plane automatically synthesized by learning from telemetry data

Real-time Network Telemetry

Operator Policies & Intents

You specify what you want, and the network figures out how to deliver it!
Data Driven Control Loop

Service KPI

Prediction

Will Current Controls Meet KPI?

Will Adjusted Controls Meet KPI?

Meets Operator Policies?

Operator Policies Intent & SLAs

Real-time Data

Network Telemetry

Apply Controls

Recommend Current Control Settings

Recommend Adjusted Control Settings

Apply Controls

Recommend Current Control Settings

Recommend Adjusted Control Settings

What-If Adjust Control Knobs

Y

N

Y

Y

N

N

N

N
Two learning approaches:

Classical Machine Learning

- Requires extensive domain knowledge

Deep Reinforcement Learning

- Largely blind, truly approximates intent based system design
This Talk: Applying this Framework to Commercial Mobile Networks

Operator policy: Optimize user throughput for the 20% worst affected users

Service KPI

Prediction

Will Current Controls Meet KPI?

Y

Apply Controls

Recommended Current Control Settings

N

What-If Adjust Control Knobs

Will Adjusted Controls Meet KPI?

Y

Meets Operator Policies?

Y

Recommended Adjusted Control Settings

N

N

Data: Per flow Records from the mobile network

Operator Policies Intent & SLAs

Real-time Data

Network Telemetry

Control plane application: Load Balancing in mobile networks

Operator policy:
Optimize user throughput for the 20% worst affected users
Mobile traffic demand is skyrocketing

<table>
<thead>
<tr>
<th>Year</th>
<th>Data Unit</th>
<th>GB/ month</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>0.4 EB/month</td>
<td></td>
</tr>
<tr>
<td>2015</td>
<td>3.7 EB/month</td>
<td></td>
</tr>
<tr>
<td>2020</td>
<td>30.6 EB/month</td>
<td></td>
</tr>
<tr>
<td>2025</td>
<td>300 EB/month</td>
<td></td>
</tr>
</tbody>
</table>

*Courtesy: Cisco VNI, 2015-2020*

New, smarter devices

Demanding content

Phones

Smartphones

Tablets

Wearables

Audio, Text

Web

Video

Virtual Reality
How will mobile networks meet this surging traffic demand?

Densification will bring the next wave of capacity scaling in mobile access networks.
Densification has challenges
Load (demand) and other factors (e.g. interference, mobility) become highly dynamic
→ need to be proactively managed

E.g: Cells can offload users to less-congested neighbors to balance load, need to know expected user experience

For densification to scale, networks need to proactively manage load, interference and mobility to optimize user experience
Why is this challenging?

**Challenge**

User performance is hard to predict
Complex function of many, potentially unknown and/or dynamic, network variables

Need to predict user performance every ~1s for the best load distribution

**Existing approaches**

Aggregate cell-level counters blind to dynamism across time and users; reactive

Need to **predict user experience** seconds in advance, hard in practical networks with many unknown/dynamic variables
What do we need today to enable wide-scale densification?

Predict user performance every second
to manage load dynamically

Coordinate across cells in realtime
to balance load

Can we use a data driven approach to tackle this challenge?
without making invasive changes to existing network infrastructure
without deploying expensive network infrastructure
without modifying user devices/cellular standards
ForeC
Data-driven control
in dense mobile networks
How to use realtime analytics & learning for network control?

MONITOR
Right now, what is the avg throughput of user/cell? resource utilization of user/cell? contention faced by user/cell?

NETWORK ACTIONS
handover user admit/reject user inc/dec tx power

NETWORK STATE
cell: bandwidth, #users, demand etc. user: serving cell, link quality etc.

NETWORK EFFECTS
throughput resource utilization contention

FORECAST
In the next 1s, what will be the avg throughput of user/cell? resource utilization of user/cell? contention faced by user/cell?

FORECAST IMPACT
In the next 1s, what if: handover users? admit/reject new users? increase/decrease tx power?

Networks require the ability to monitor and forecast network effects per user/cell, per second, potentially based on network actions.
ForeC: Analytics on ‘after-the-fact’ logs to monitor & forecast network effects

Cells already expose usage stats per user session (~10s)

One report per event of the session, logged after the event
setup report, release report, traffic report, radio measurements report, mobility report etc.
ForeC has been tested in diverse deployments
How does ForeC work?
The internals of ForeC

What can ForeC do?
and how well?

How can a network use ForeC?
Load management in a stadium
How does ForeC work?
The internals of ForeC

What can ForeC do? and how well?

How can a network use ForeC?
Load management in Levi’s Stadium
How does ForeC forecast network effects?

What will be throughput of user A if handed over from cell 1 to 2 over the next 10 seconds?

ForeC’s approach: Forecast the causes, predict the effect. Decoupling simplifies design and implementation, also works quite well!
The internals of ForeC

**QUERY LAYER:** Query parsing and response

**ANALYTICS LAYER**
- Proposed state change
- Future effects
- Current effects
- Current states

- PREDICTION ENGINE
- FORECAST ENGINE
- MONITORING ENGINE

**DATA LAYER:** Data sanitization and organization

- OFFLINE PATH: Past and current data
- ONLINE PATH: Current data

**queries**

**responses**

**user session logs**
How do we decompose an effect into its constituent causes?

Use domain knowledge to formulate/hypothesize causes, use learning tools to test significance and discover relationships.
How do we forecast the dynamic causes?

Test for seasonality and trend, use appropriate version of ARIMA time-series forecasting.
ForeC can forecast network effects per user, per second, aggregates

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Example query</th>
<th>Median error</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Basic primitives</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREDICT effect per user</td>
<td>Throughput of user A if it were on cell 2 instead of 1?</td>
<td>8.5 %</td>
</tr>
<tr>
<td>based on (changes in) network state</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FORECAST effect per user per second</td>
<td>Throughput of user A over the next 1s?</td>
<td>13.0 %</td>
</tr>
<tr>
<td><strong>Extensions to aggregates over space and time</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FORECAST effect per cell, sector, network...</td>
<td>Average throughput of cell 1 in the next 1s?</td>
<td>4.0 %</td>
</tr>
<tr>
<td>FORECAST effect per 10s of seconds, minutes...</td>
<td>Average throughput of cell 1 in the next 5s?</td>
<td>1.7 %</td>
</tr>
</tbody>
</table>

**Network effects**: throughput, resource utilization, contention...& any function of them

**Network actions**: handover user, admit new user, increase/decrease transmit power
How does ForeC work?
The internals of ForeC

What can ForeC do?
and how well?

How can a network use ForeC?
Load management in a stadium
ForeC lets operator monitor, forecast and optimize network KPIs

[SWITCH TO TABLEAU] MONITOR – FORECAST – OPTIMIZE

Dash 1: Operators can monitor network KPIs at a fine resolution in space and time (show “ACTUAL” KPI for Band 700, 1900, 2100) [can also define new KPIs to monitor]

Dash 2: Operators can forecast these KPIs to manage load proactively (show “ACTUAL” KPI and “FORECAST” KPI for a cell)

Dash 3: Operators can choose the best load management to optimize these KPIs (show “ACTUAL” and “OPTIMIZED” KPI)
Can we do this unsupervised?

Classical Machine Learning

- Key Challenge: Deep RL is very data hungry, you cannot run billions of A/B tests on a live network
- What we need: Network/System Simulators we trust!
  - Very hard given the complex interactions in these systems

Deep Reinforcement Learning

- Neural Network Reinforcement Learning (Prediction & What-if)
Summary & Takeaways

• Goal: Building a programmable connectivity layer for future applications
  – Applications programmatically specify their connectivity requirements
  – The network platform leverages realtime streaming telemetry and learning to automatically deliver that connectivity

• You specify what you want, and the network figures out how to deliver it!