Machine Learning Meets Cellular Networks

Xin Liu

Computer Science Department

University of California, Davis

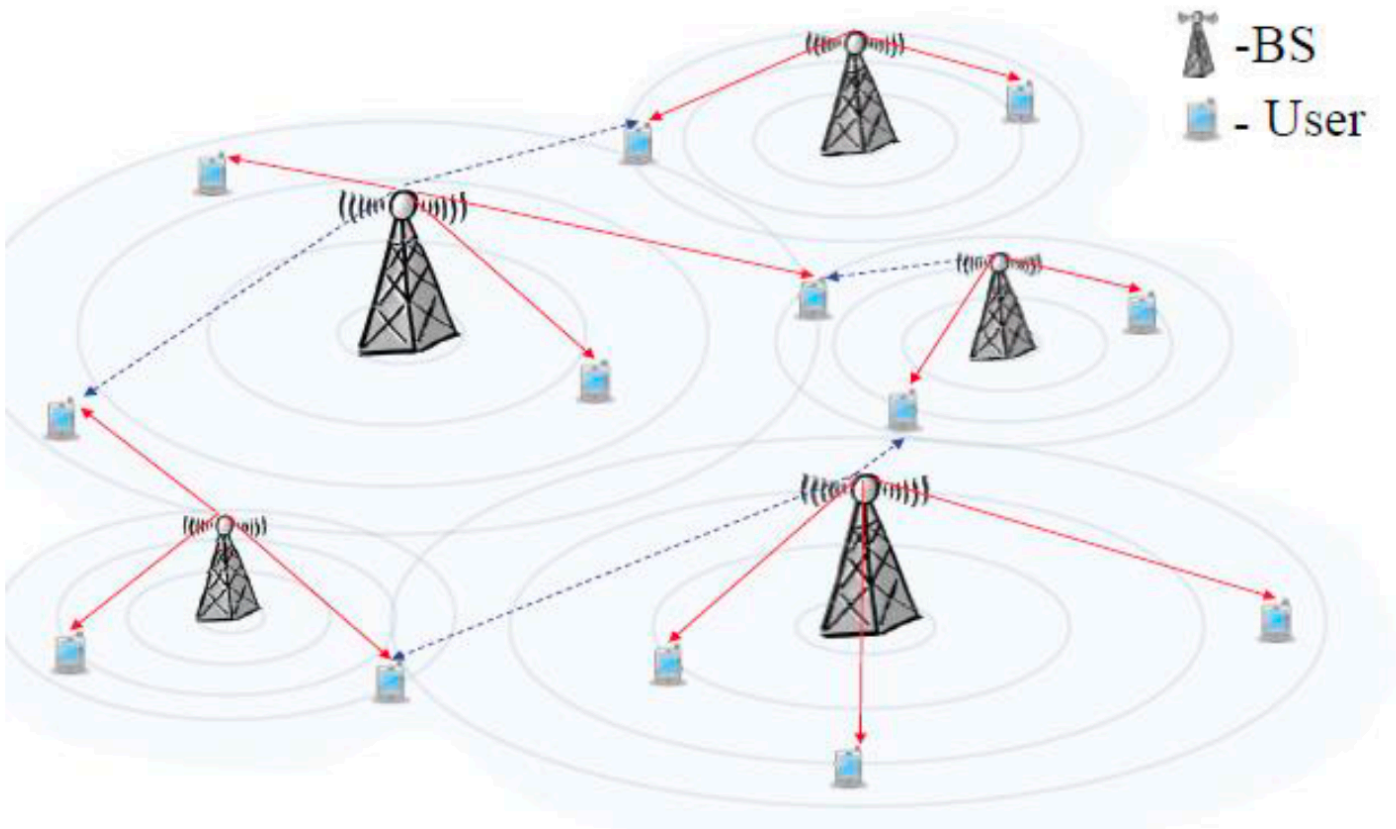
A Little Bit about Myself

- B.S, M.S., Xi'an Jiaotong University
- Ph.D., Dept. of EE, Purdue University
- Post-doc, UIUC
- Professor, CS, UC Davis
- MSRA, 2012- 2014

- Cellular resource management, opportunistic scheduling
- Cognitive radio networks
- Mobile resource management and personalization
- **Data-driven approaches in networking**

Cellular Network Configuration

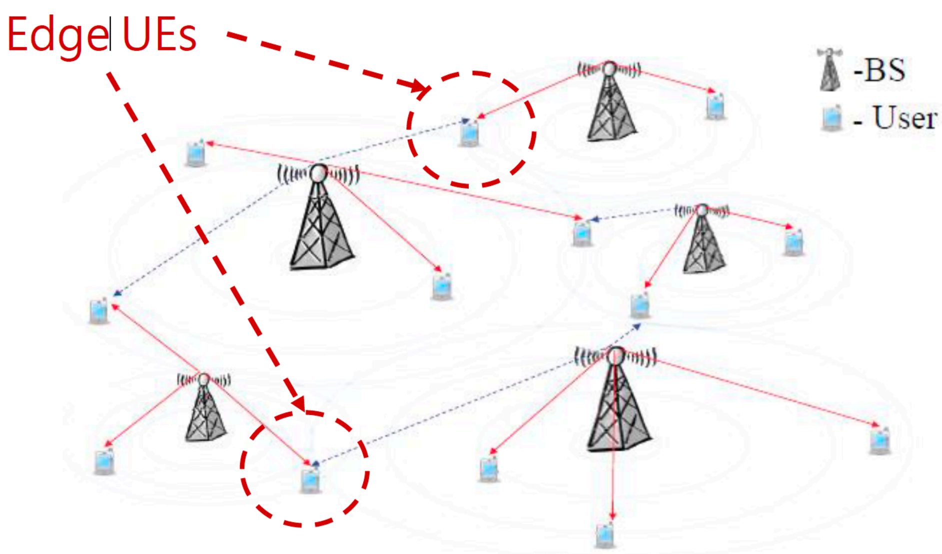
A large number of parameters to configure



Cellular Network Configuration

- A large # of parameters
  - BS transmission power
  - Thresholds for handover
  - Max. # of users
  - Antenna direction
  - Etc.
- Performance metrics
  - Total throughput
  - Average data rate
  - Edge user performance
  - Resource utilization rate
  - Etc.

Example: Handover



- Performance metric
  - Edge UE throughput
- Parameter conf.
  - A2-threshold-RSRP
  - Defined in LTE
- Impact on performance
  - Too small
  - Too large

Challenges

- Complex mapping from configurations (and network state) to network performance
- Traditional approach
  - By human expert
  - Labor intensive and suboptimal
    - Hours/days to tune on configuration based on experience
    - Tune one or two parameters each time
    - Hundreds of parameters
- As a result, most cells set the configurations to default values

Our Study

- Machine-learning-based approaches to automate this process
- Challenges:
  - Lots of data and still a cold start
  - Network performance is a highly complex (and noisy) function of configurations and cell states
  - Limited amount of exploration
    - Network operators are risk-averse
    - Limited configuration adjustment frequency (e.g., one adjustment a day)
    - Limited duration for adjustment (e.g., two weeks)

Our Approach

- Collaborative learning
- Formulate the problem as a transfer contextual multi-armed bandit problem
- Prove that the regret bound can be significantly reduced
- Develop a practical algorithm to decompose the policy of a cell into common and cell-specific components
  - the common component utilizes transfer learning and faster policy convergence;
  - the cell-specific component addresses dissimilarities amongst different cells
- A live field with 1700+ cells
  - 20% of performance improvement

Problem Formulation- Contextual Bandit

- A cellular network of N cells
- Adjust configurations in T time steps

s_t^i: state of Cell i at time t (e.g., number of users, channel quality)

a_t^i: chosen parameter configuration values of Cell i at time t (e.g., transmission power)

f_i: unknown true performance function of Cell i (depends on s_t^i and a_t^i)

y_t^i: noisy observation of network performance of Cell i at time t (e.g., cell throughput)

$$y_t^i = f_i(\mathbf{s}_t^i, \mathbf{a}_t^i) + \xi_t^i$$

- Objective: minimize the overall regret

$$\min_{\mathbf{a}_t^i:\forall i,t} \sum_{t=1}^{T} \sum_{i=1}^{N} \left\{ \max_{\mathbf{a}' \in \mathcal{A}} f_i(\mathbf{s}_t^i, \mathbf{a}') \right\} - f_i(\mathbf{s}_t^i, \mathbf{a}_t^i)$$

Contextual Bandits & Cell Configuration

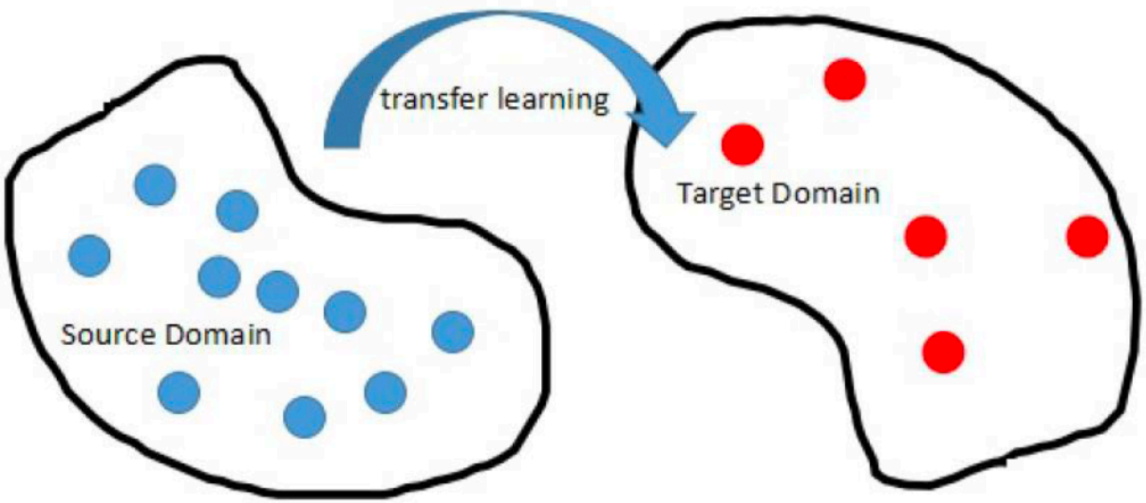
- Balance the tradeoff between exploitation and exploration
  - Existing algorithms: LinUCB, Thompson Sample, epsilon-greedy, etc.
- Generally, for a cell at a given time
  - Observe cell state; e.g., # of users, channel quality, traffic volume
  - Choose a configuration based on the learned performance model and action selection policy
  - Observe the noisy performance measurement
  - Update the learned model

Transfer Learning for Fast Convergence

- Key challenge: each cell has only limited chances for exploration
  - Ex: one configuration each day
  - Exploration limited to two weeks
- Learning bandit independently converges **slowly**
  - # of exploration limited per cell
  - Large # of parameters and their combinations

Transferable contextual bandit

- Share data among cells to accelerate the learning of the performance model



Fast Convergence by Transfer Learning

- The bound of the instantaneous regret at each step can be reduced by a discounting factor ***γ***<**1** leveraging data from other cells

$$\sigma_{\mathcal{T}+\mathcal{S},t-1}(\mathbf{x}_t) \leq \underbrace{\sqrt{\frac{1+\sigma^{-2}N_{\mathcal{T}}\beta}{1+\sigma^{-2}N_{\mathcal{S}}\alpha+\sigma^{-2}N_{\mathcal{T}}\beta}}}_{\gamma}\sigma_{\mathcal{T},t-1}(\mathbf{x}_t),$$

Scaled regret bound at slot t with transfer learning

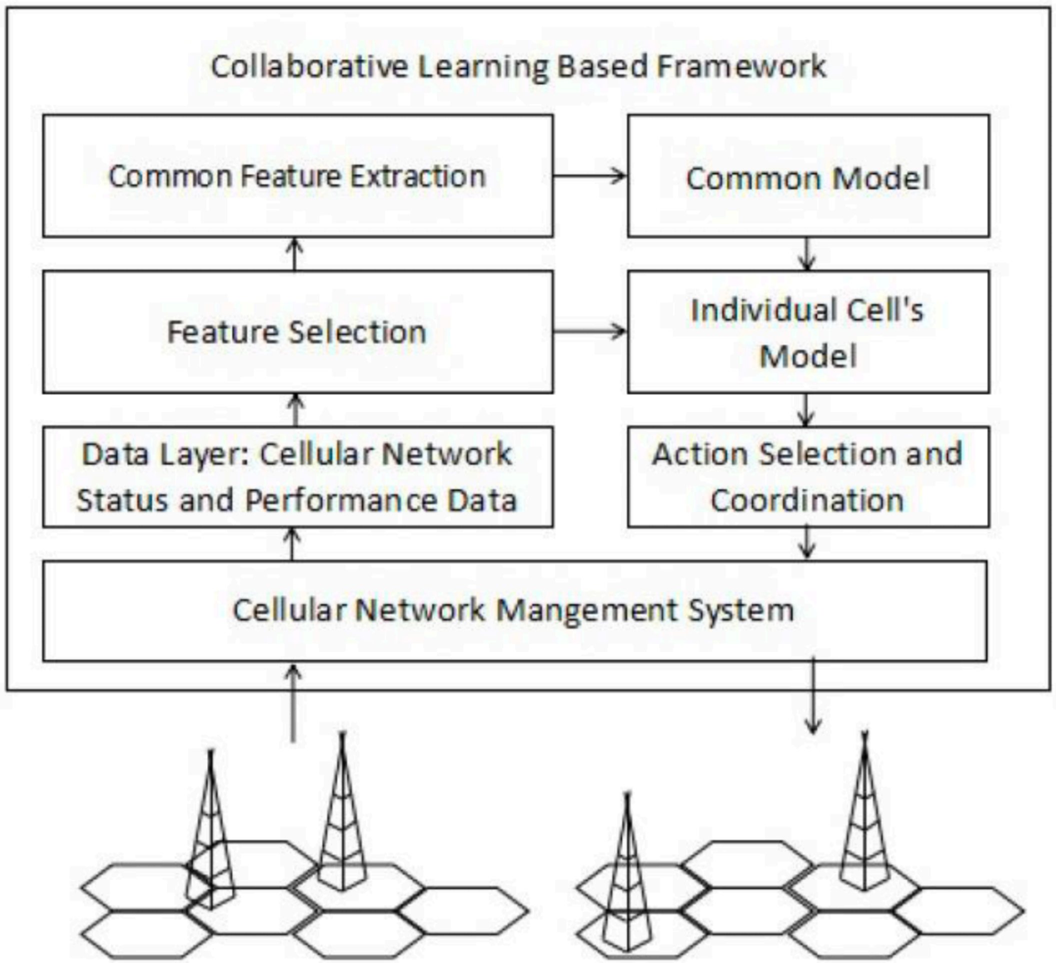
Scaled regret bound at slot t without transfer learning

- The sped-up is more significant when
  - # of transferred samples >> # of original samples in the target cell

From Theory to Practice

- Collaborative learning: utilize data from all cells to learn the performance model for each cell
- Key idea: consider both common behavior and cell-specific behavior
  - Decompose the model into a common part and a cell-specific part
  - User data from all cells to learn the common part
  - Use each cell's own data to learn the cell-specific part

Collaborative Contextual MAB Framework



Model Decomposition

State decomposition: $\mathcal{S} = \tilde{\mathcal{S}} \times \hat{\mathcal{S}}$.

$$\mathbf{s}_t^i = (\tilde{\mathbf{s}}_t^i, \hat{\mathbf{s}}_t^i), \text{ where } \tilde{\mathbf{s}}_t^i \in \tilde{\mathcal{S}} \text{ and } \hat{\mathbf{s}}_t^i \in \hat{\mathcal{S}}.$$

- Common features: $\tilde{\mathbf{s}}_t^i = W^T \mathbf{s}_t^i$
- Cell-specific features: $\hat{\mathbf{s}}_t^i = (I - WW^T)\mathbf{s}_t^i$

Performance Decomposition

$$y_t^i = h(\tilde{\mathbf{s}}_t^i, \mathbf{a}_t^i) + g_i(\mathbf{s}_t^i, \mathbf{a}_t^i) + \epsilon_t^i$$

Common behaviors

Cell-specific behaviors

Model Learning

Extract common features (data from all cells):

$$W^* = \arg\max_W \sum_i \sum_t \|cov(W^T \mathbf{s}_t^i, y_t)\|_F^2$$

Learn the common model (data from all cells):

$$h_{t-1}^* = \arg\min_h \sum_{i=1}^{N} \sum_{t'=1}^{t-1} \|y_{t'}^i - h(W^T \mathbf{s}_{t'}^i, \mathbf{a}_{t'}^i)\|_2^2 + \lambda C_h$$

Learn the cell-specific model:

- Calculate prediction residual of the common model: $\tilde{y}_{t'}^i = y_{t'}^i - h_{t-1}^*(W^T \mathbf{s}_{t'}^i, \mathbf{a}_{t'}^i)$
- Learn regression model of the residual: $g_{i,t-1}^* = \arg\min_{g_i} \sum_{t'=1}^{t-1} \|\tilde{y}_{t'}^i - g_i\left(\mathbf{s}_{t'}^i, \mathbf{a}_{t'}^i\right)\|_2^2 + \lambda C_{g_i}$

Action Selection

$$\mathbf{a}_t^{i,*} = \arg\max_{\mathbf{a}^i} h_{t-1}^*(\tilde{\mathbf{s}}_t^i, \mathbf{a}^i) + g_{i,t-1}^*(\mathbf{s}_t^i, \mathbf{a}^i), \text{ w.p. } 1 - \epsilon,$$

$$\mathbf{a}_t^{i,*} \sim U(|\mathcal{A}|) \text{ w.p. } \epsilon,$$

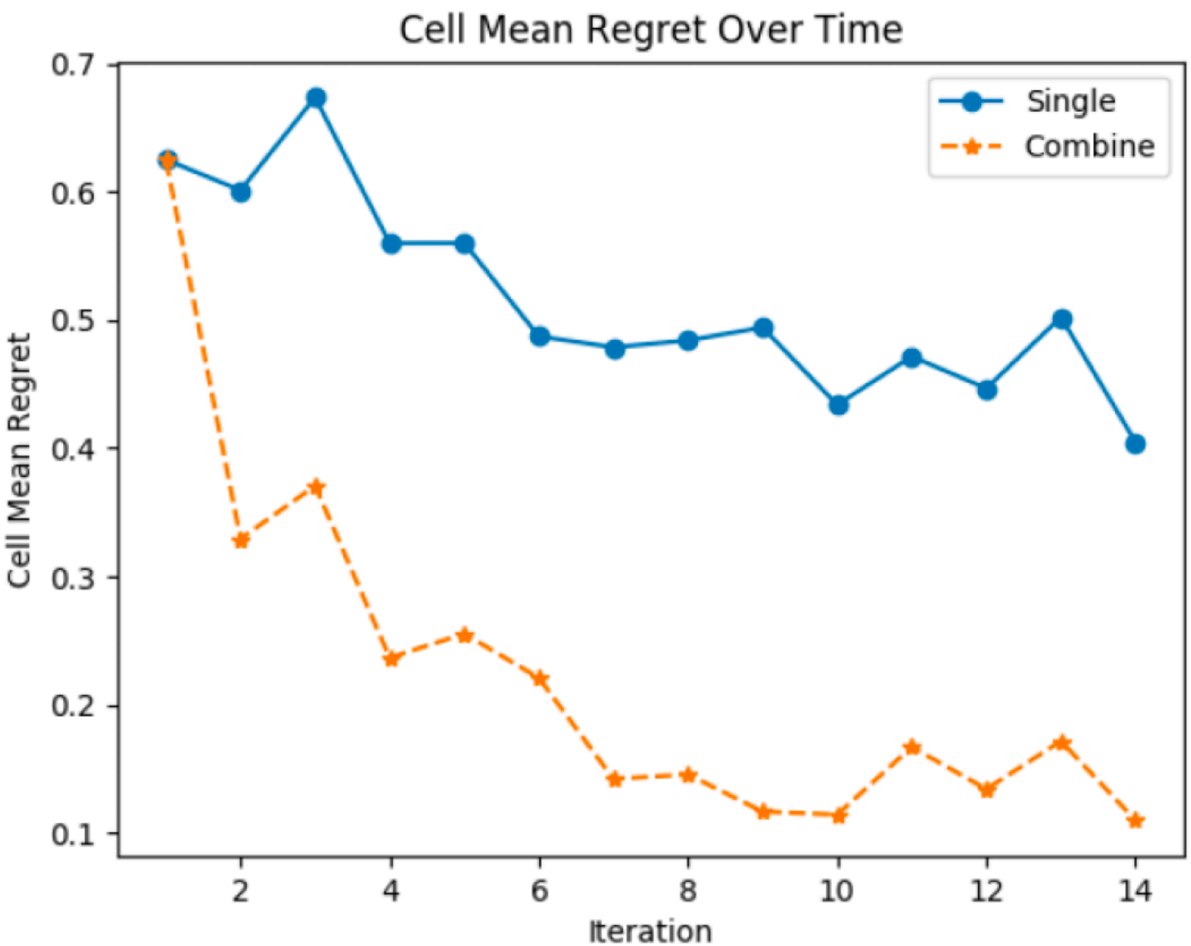
Evaluation – Trace-driven Simulation

Datasets collected from cellular networks of a metropolitan city:

- Single-parameter dataset:
  - data from 297 cells over 17 days
  - one parameter configuration related to handover is adjusted once each day for each cell
  - performance metric is edge UE ratio (less-than-5M-ratio)
  - one sample is collected each hour for each cell, including cell state measurements, configured parameter, value of the performance metric

- Multi-parameter dataset:
  - data from 185 cells over 14 days
  - two parameter configurations related to uplink power control are adjusted once each day for each cell
  - performance metric is edge UE ratio
  - one sample collected each hour for each cell (including cell states measurements, configured parameter, value of the performance metric)

Multi-Parameter Simulation Results



Evaluation- Live Field Test

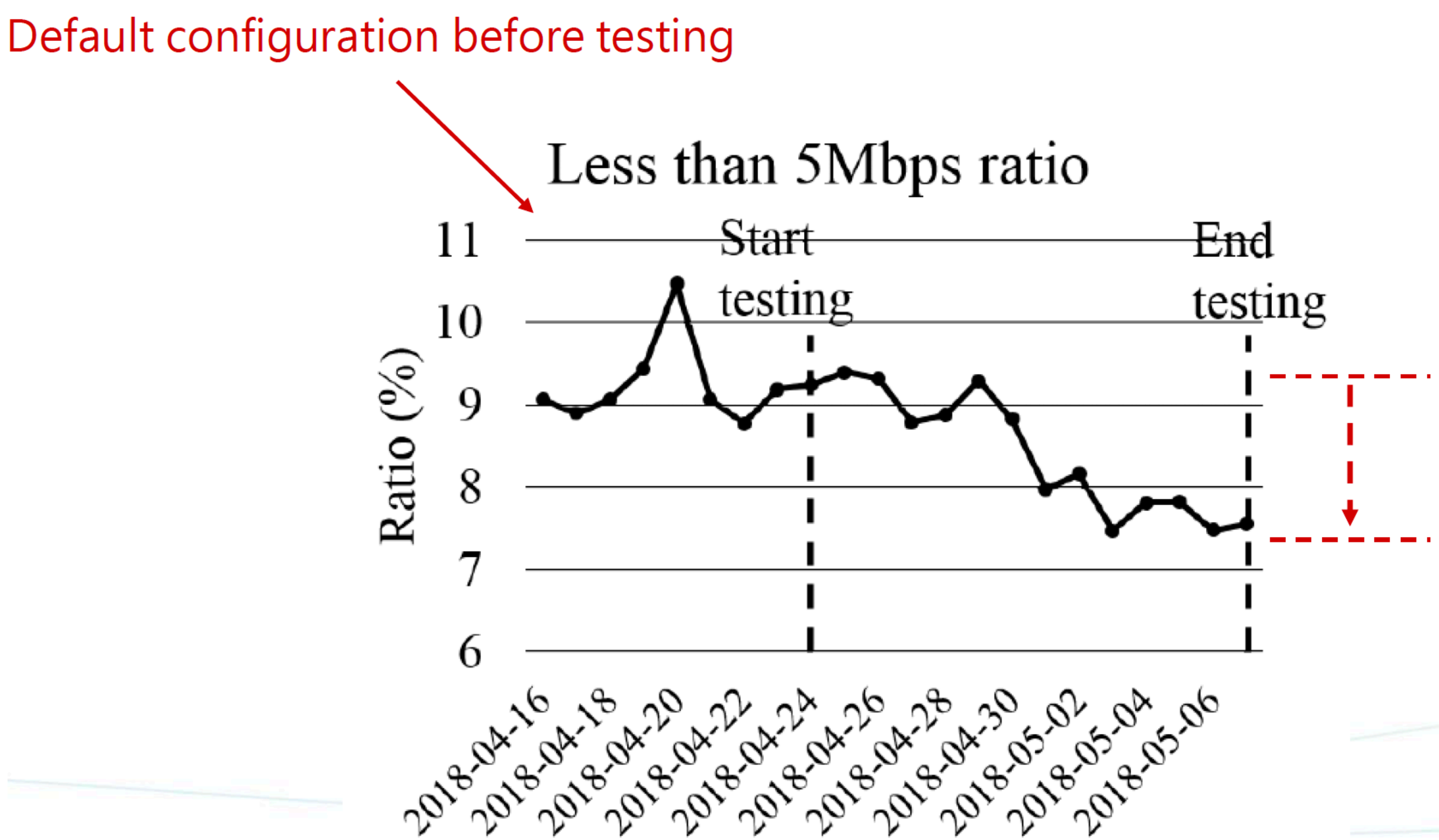
- Adjusted 5 parameter configurations within two weeks in April 2018
- Each configuration has around 10 possible values
- One adjustment for each cell for each day is allowed
- 1700+ cells in a metropolitan city tested

PARAMETERS OPTIMIZED IN REAL NETWORK TEST

Parameter	Meaning
A	An upper bound on the uplink reception power; used for uplink power control
B	Target initial downlink BLER; used for deciding downlink modulation and coding scheme (MCS)
C	Controls how MCS is adjusted to utilize unoccupied resource blocks (RBs)
D	Controls the initial MCS of users
E	Controls the MCS adjustment speed

Results

- Metric: edge UE ratio (the smaller the better)



Summary

- Collaborative-learning-based approach for cellular network configuration
- Contextual bandits with transfer learning for better data efficiency
- Models decomposed to accommodate common and cell-specific behavior
- Significant performance improvement

Our Related Work

- Cellular network configuration based on MAB and Gibbs-sampling
- Mobile prefetching based on user-profiling
- Data-driven resource allocation for cellular user experience improvement
- Prediction-based 360-video transmission
- Opportunistic bandits for efficient learning
- Constrained contextual bandits and dueling bandits
- Deep-learning-based RF fingerprinting
- Encrypted traffic classification
- **Security issues of ML algorithms**
- **Network slicing and NOMA pairing**

Challenges and Opportunities

- Opportunities:
  - Networks are large engineered complex systems – beyond white-box modeling
  - Environment: highly dynamic and partially observable
  - All layers and all players
- Challenges:
  - Data, data, data
  - Evaluation/experimentation
  - Interpretability, safety, security, privacy
  - Does not always outperform existing algorithms


Thank you