Pricing Cloud Bandwidth Reservations under Demand Uncertainty

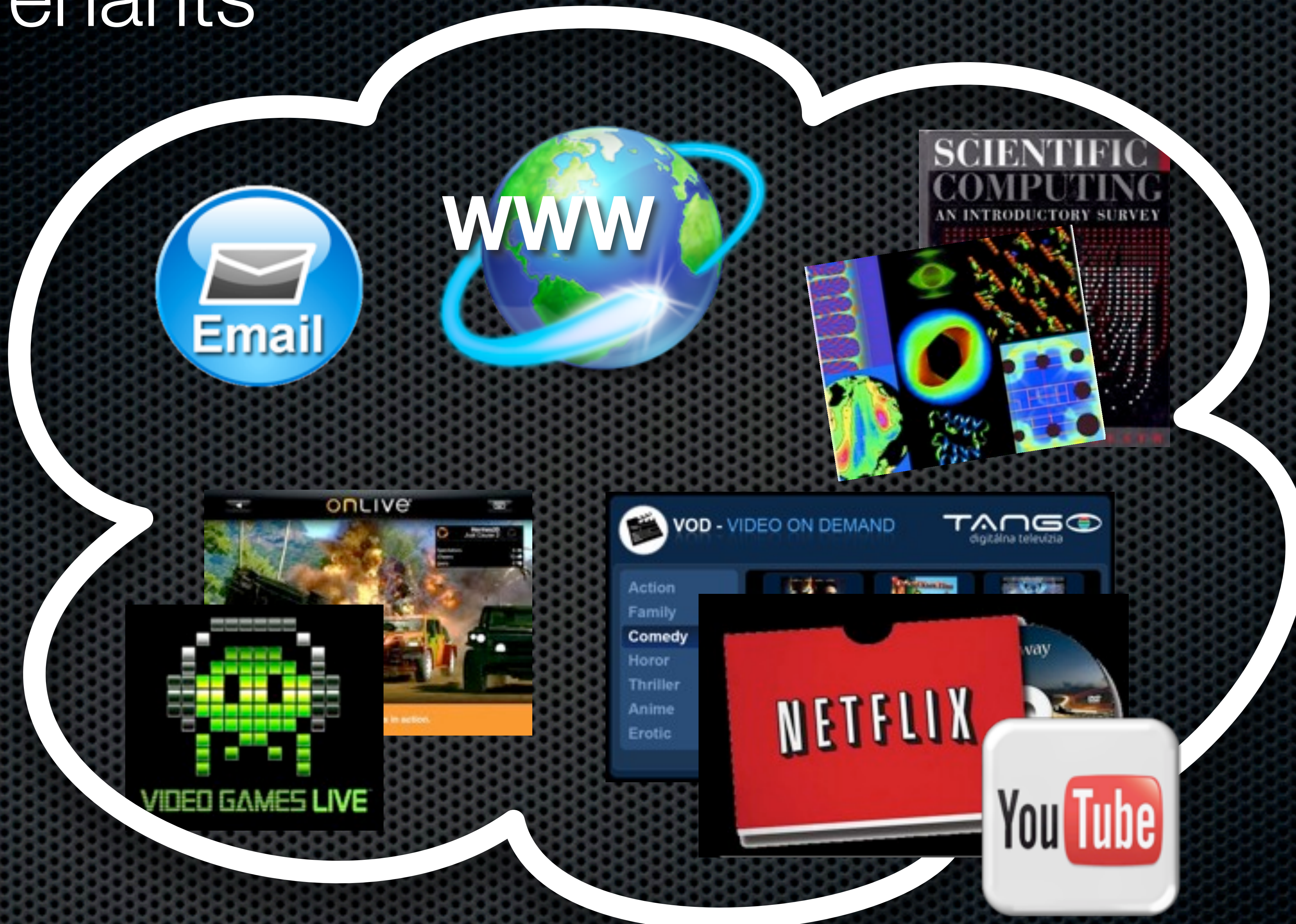
Di Niu, Chen Feng, Baochun Li

Department of Electrical and Computer Engineering
University of Toronto

Roadmap

- ***Part 1*** A cloud bandwidth reservation model
- ***Part 2*** Price such reservations
  - Large-scale distributed optimization
- ***Part 3*** Trace-driven simulations

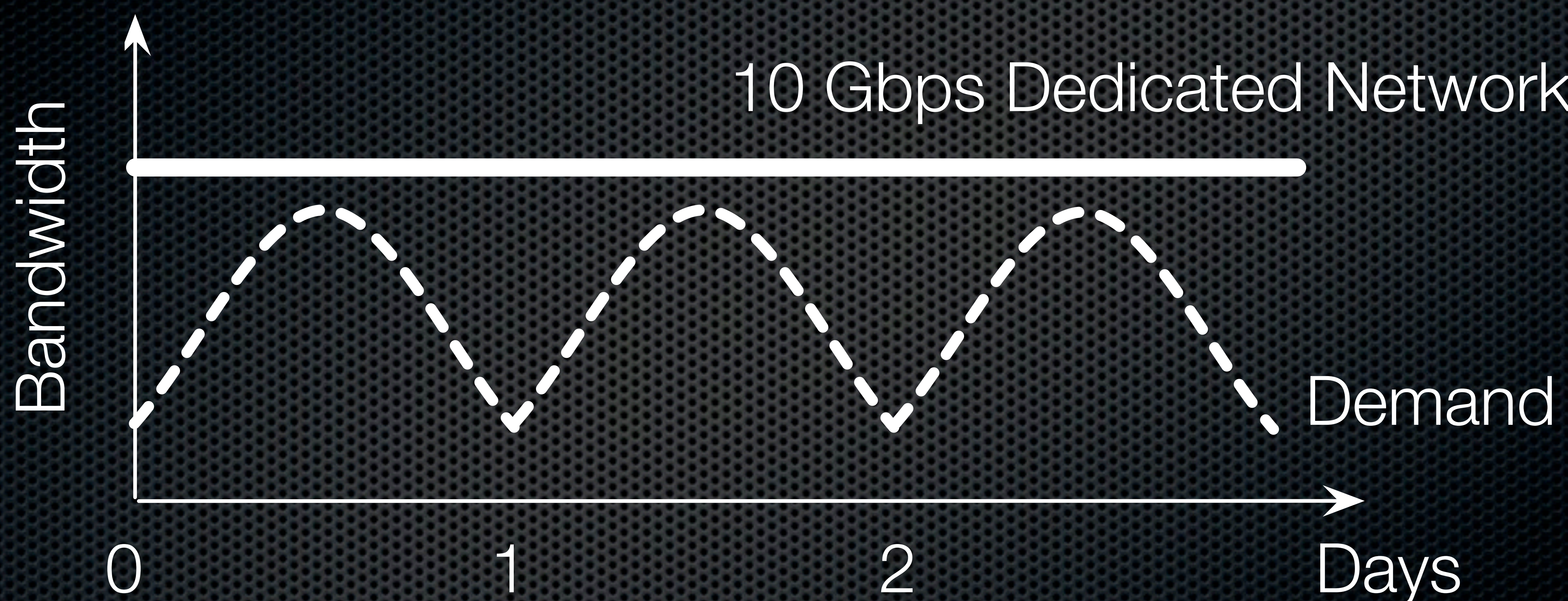
Cloud Tenants



Problem: No bandwidth guarantee

Not good for Video-on-Demand, transaction processing web applications, etc.

Amazon Cluster Compute



Over-provision

Good News:
Bandwidth reservations are becoming feasible between a VM and the Internet

H. Ballani, et al.

Towards Predictable Datacenter Networks

ACM **SIGCOMM '11**

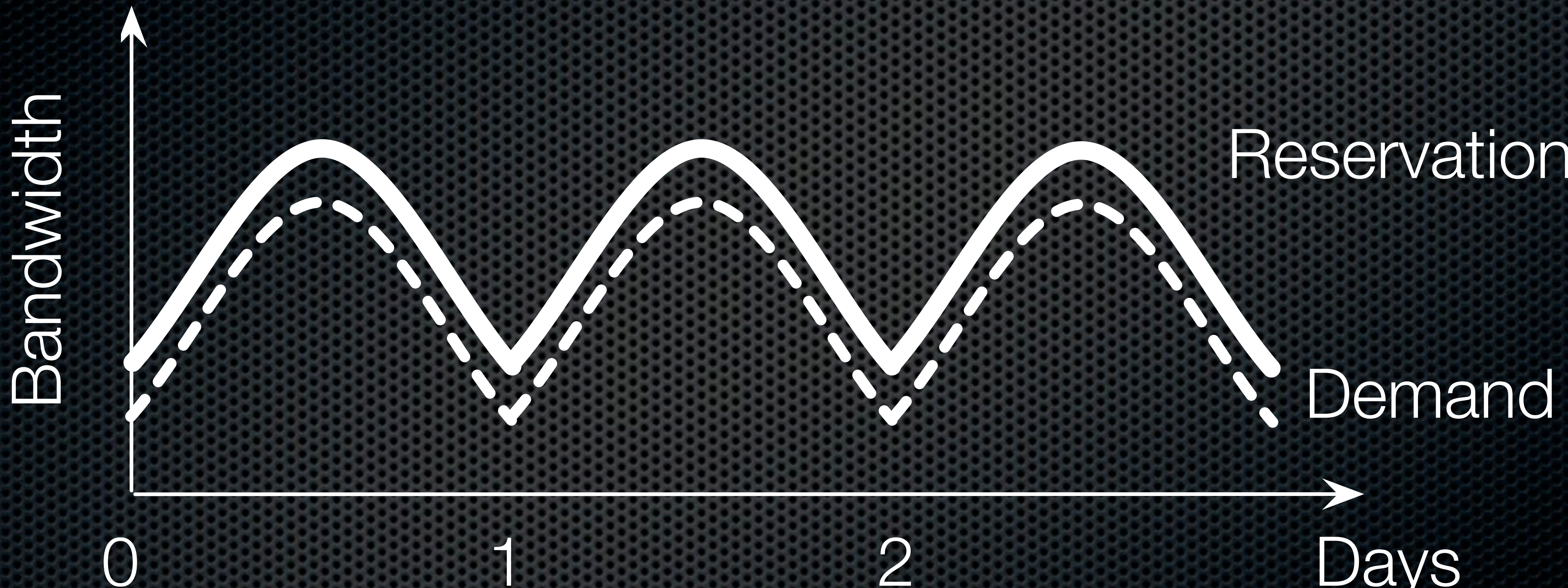
C. Guo, et al.

SecondNet: a Data Center Network Virtualization Architecture with Bandwidth Guarantees

ACM **CoNEXT '10**

Dynamic Bandwidth Reservation

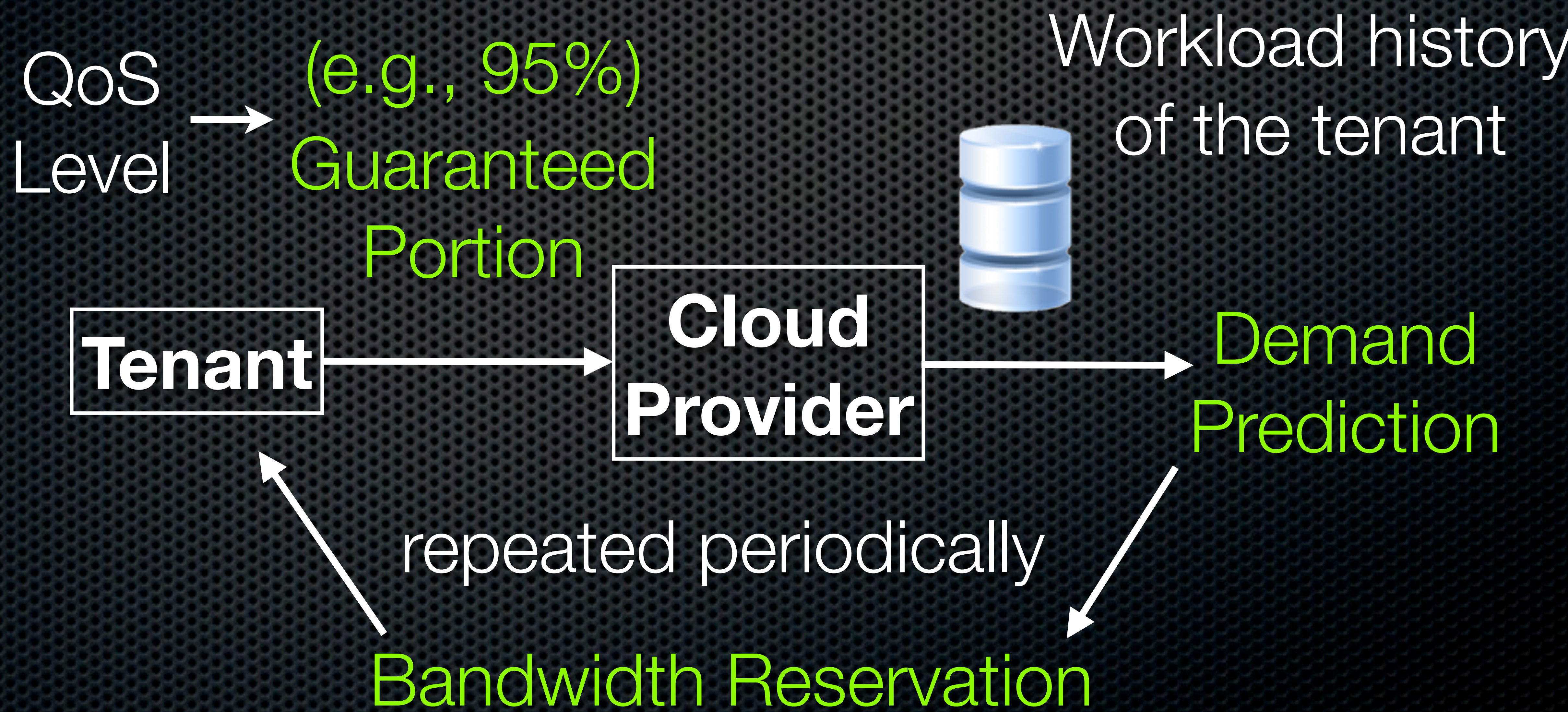
reduces cost due to better utilization



Difficulty: tenants don't really know their demand!

A New Bandwidth Reservation Service

A tenant specifies a *percentage of its bandwidth demand* to be served with guaranteed performance; The remaining demand will be served with best effort



Tenant Demand Model

- Each tenant i has a *random* demand D_i
- Assume D_i is Gaussian, with
  - mean $\mu_i = \mathrm{E}[D_i]$
  - variance $\sigma_i^2 = \mathrm{var}[D_i]$
  - covariance matrix $\Sigma = [\sigma_{ij}]$
- Service Level Agreement: Outage w.p. ϵ

Roadmap

- ***Part 1*** A cloud bandwidth reservation model
- ***Part 2*** Price such reservations
  - Large-scale distributed optimization
- ***Part 3*** Trace-driven simulations

Objectives

- Objective 1: Pricing the *reservations*
    - A reservation fee on top of the usage fee
- Objective 2: Resource Allocation
    - Price affects demand, which affects price in turn
    - Social Welfare Maximization

Tenant Utility (e.g., Netflix)

Tenant i can specify a guaranteed portion w_i

Tenant i's *expected* utility (revenue)

$$\mathrm{E}[u_i(w_i, D_i)] = U_i(w_i, \mu_i, \sigma_i, \ldots)$$

Concave, twice differentiable, increasing

Utility depends not only on demand, but also on the guaranteed portion!

Bandwidth Reservation

- Given submitted guaranteed portions

$$\mathbf{w} = [w_1, \ldots, w_N]^{\mathsf{T}}$$

the cloud will guarantee the demands

$$w_1 D_1, \ldots, w_N D_N$$

- It needs to reserve a total bandwidth capacity $K(\mathbf{w})$

Non-multiplexing: $\Pr(w_i D_i > R_i) < \epsilon,\ K = \sum_i R_i$

Multiplexing: $\Pr(\sum_i w_i D_i > K) < \epsilon$

Service cost $\mathrm{cost}(\mathbf{w}) = \mathrm{cost}(K(\mathbf{w})) \overset{\text{e.g.}}{=} \beta K(\mathbf{w})$

Cloud Objective: Social Welfare Maximization

$$\underbrace{\max_{w_1,\ldots,w_N} \sum_i U_i(w_i) - \mathrm{cost}(w_1,\ldots,w_N)}_{\textit{Social Welfare}}$$

Price

$$= \sum_i \underbrace{\left(U_i(w_i) - P_i\right)}_{\text{Surplus of tenant } i} + \underbrace{\sum_i P_i - \mathrm{cost}(\mathbf{w})}_{\text{Profit of the Cloud}}$$

Impossible: the cloud does not know U_i

Pricing Function

Pricing function $P_i(w_i)$

Price guaranteed portion,

not absolute bandwidth!

Example: Linear pricing $P_i(w_i) = k_i w_i$

Under $P_i(\cdot)$, tenant i will choose

$$\tilde{w}_i = \arg\max_{w_i} \underbrace{\boxed{U_i(w_i) - P_i(w_i)}}_{\text{Surplus (Profit)}}$$

Pricing as a Distributed Solution

Determine pricing policy $\{P_i(\cdot)\}$ to

$$\max \underbrace{\textstyle\sum_i U_i(\tilde{w}_i) - \mathrm{cost}(\tilde{w}_1, \ldots, \tilde{w}_N)}_{\textit{Social Welfare}}$$

where $\tilde{w}_i = \arg\max_{w_i} \underbrace{U_i(w_i) - P_i(w_i)}_{\textit{Surplus}}$

Challenge:

Cost not decomposable for multiplexing

A Simple Case: Non-Multiplexing

- Determine pricing policy $\{P_i(\cdot)\}$ to

$$\max_{\{P_i(\cdot)\}} \sum_i \left(U_i(\tilde{w}_i) - \text{cost}_i(\tilde{w}_i)\right)$$

where $\tilde{w}_i = \arg\max_{w_i} U_i(w_i) - P_i(w_i)$

$$P_i(w_i) = \text{cost}_i(w_i)$$

Since $\Pr(w_i D_i > R_i) = \epsilon$, for Gaussian D_i

$$\text{cost}_i(w_i) \sim R_i = (\mu_i + \theta(\epsilon)\sigma_i)w_i$$

Mean Std

The General Case: Lagrange Dual Decomposition

M. Chiang, S. Low, A. Calderbank, J. Doyle. Layering as optimization decomposition: A mathematical theory of network architectures. **Proc. of IEEE 2007**

Original problem $\max_{\mathbf{w}} \sum_i U_i(w_i) - \mathrm{cost}(\mathbf{w})$

$$\max_{\mathbf{w},\mathbf{v}} \sum_i U_i(w_i) - \mathrm{cost}(\mathbf{v}) \quad \text{s.t.} \quad \mathbf{w} = \mathbf{v}$$

$$L(\mathbf{w},\mathbf{v},\mathbf{k}) = \sum_i U_i(w_i) - \mathrm{cost}(\mathbf{v}) + \mathbf{k}^{\mathsf{T}}(\mathbf{v} - \mathbf{w})$$

$$= \sum_i (U_i(w_i) - k_i w_i) + \mathbf{k}^{\mathsf{T}}\mathbf{v} - \mathrm{cost}(\mathbf{v})$$

Lagrange dual $q(\mathbf{k}) = \sup_{\mathbf{w},\mathbf{v}} L(\mathbf{w},\mathbf{v},\mathbf{k})$

Dual problem $\min_{\mathbf{k}} q(\mathbf{k})$

$$L(\mathbf{w}, \mathbf{v}, \mathbf{k}) = \sum_i U_i(w_i) - \text{cost}(\mathbf{v}) + \mathbf{k}^\mathsf{T}(\mathbf{v} - \mathbf{w})$$
$$= \sum_i (U_i(w_i) - k_i w_i) + \mathbf{k}^\mathsf{T}\mathbf{v} - \text{cost}(\mathbf{v})$$

Lagrange dual $q(\mathbf{k}) = \sup_{\mathbf{w},\mathbf{v}} L(\mathbf{w}, \mathbf{v}, \mathbf{k})$

Dual problem $\min_{\mathbf{k}} q(\mathbf{k})$

Lagrange multiplier k_i as price: $P_i(w_i) := k_i w_i$

decompose

$$\tilde{w}_i = \arg\max_{w_i} U_i(w_i) - k_i w_i$$
$$\tilde{\mathbf{v}} = \arg\max_{\mathbf{v}} \mathbf{k}^\mathsf{T}\mathbf{v} - \text{cost}(\mathbf{v})$$

Subgradient Algorithm:

For dual minimization, update price:

$$\mathbf{k} = \mathbf{k} + \text{step} \times \underline{(\tilde{\mathbf{v}} - \tilde{\mathbf{w}})}$$

a subgradient of $q(\mathbf{k})$

Weakness of the Subgradient Method

4 Update $P_i(\cdot)$ to increase $\boxed{\sum_i U_i(\tilde{w}_i) - \mathrm{cost}(\tilde{w}_1, \ldots, \tilde{w}_N)}$

Social Welfare (SW)

Cloud Provider

3 Guaranteed Portion $\tilde{w}_i$

Price $P_i(w_i)$ **1**

Tenant 1 $\cdots$ **Tenant i** $\cdots$ **Tenant N**

2 $\tilde{w}_i = \arg\max_{w_i} \boxed{U_i(w_i) - P_i(w_i)}$

Surplus

Step size is a issue! Convergence is slow.

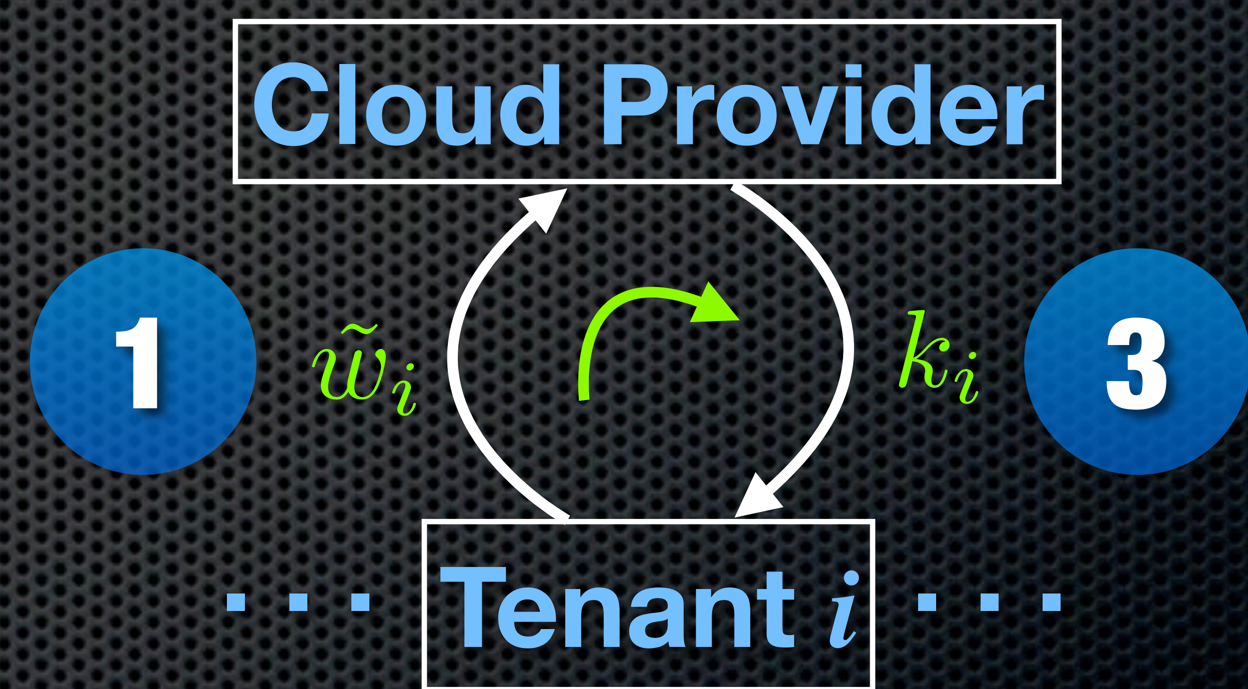
Our Algorithm: Equation Updates

KKT Conditions of

$\max \sum_i U_i(w_i) - \text{cost}:$

$$\begin{cases} U_i'(w_1) = \dfrac{\partial \text{cost}(\mathbf{w})}{\partial w_1}, \\ \ldots, \\ U_i'(w_N) = \dfrac{\partial \text{cost}(\mathbf{w})}{\partial w_N} \end{cases}$$

2 Set $k_i = \left. \dfrac{\partial \text{cost}(\mathbf{w})}{\partial w_i} \right|_{\mathbf{w}=\tilde{\mathbf{w}}}$



4 Solve $U_i'(\tilde{w}_i) = k_i$

$\tilde{w}_i = \arg\max_{w_i} U_i(w_i) - k_i w_i$

Linear pricing $P_i(w_i) = k_i w_i$ *suffices!*

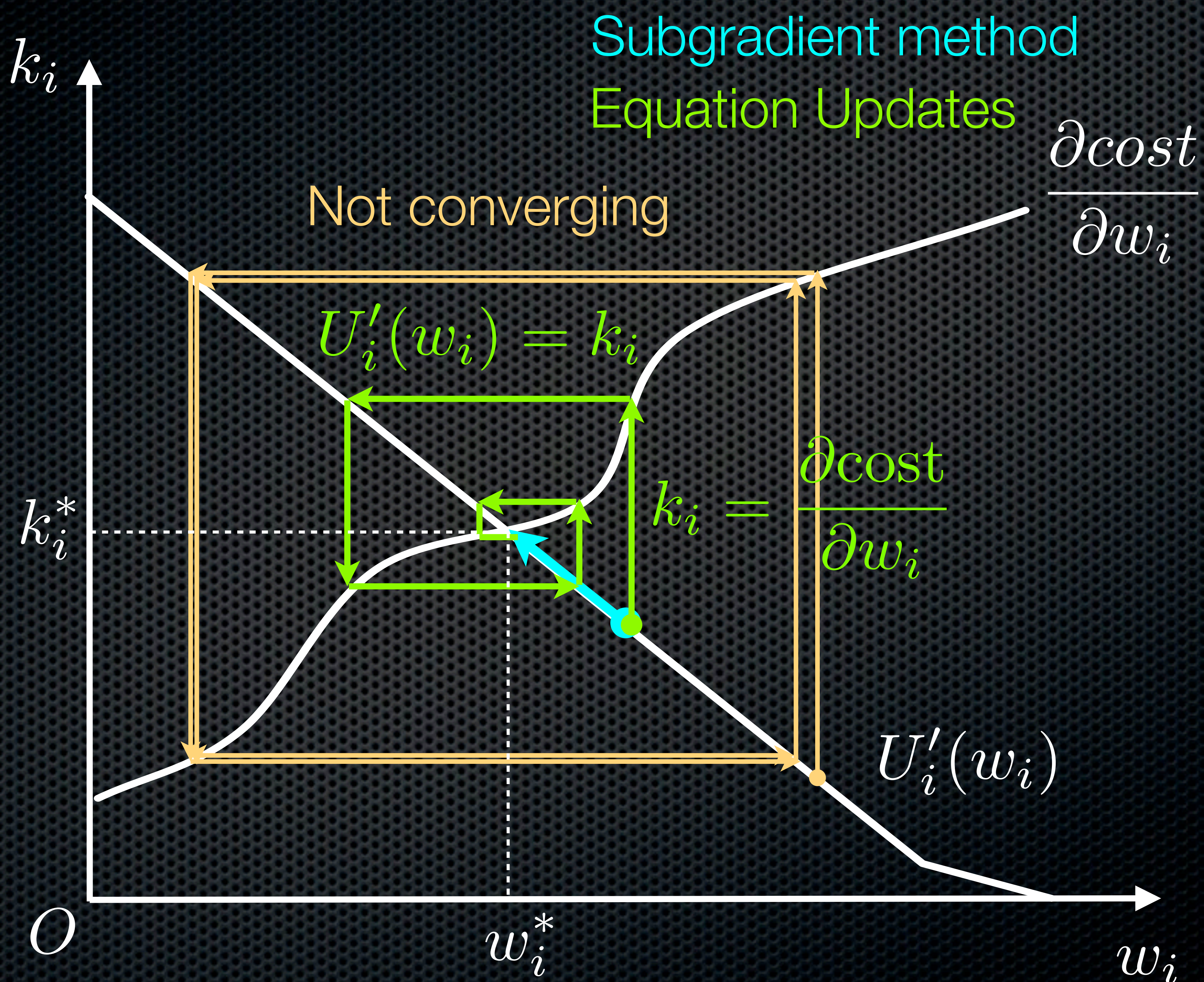
Theorem 1 (Convergence)

Equation updates converge if for all i

$$\min_{x_i} \left| U_i''(x_i) \right| > \sum_{j=1}^{N} \left| \frac{\partial^2 \mathrm{cost}(\mathbf{w})}{\partial w_i \partial w_j} \right|$$

for all $\mathbf{w}$ between $\mathbf{w}^{(0)} = \mathbf{1}$ and $\mathbf{w}^{(1)}$

Convergence: A Single Tenant (1-D)



The Case of Multiplexing

$\Pr(\sum_i w_i D_i > K) = \epsilon$

$$K(\mathbf{w}) = \mathbf{E}[\sum_i w_i D_i] + \theta(\epsilon)\sqrt{\mathbf{Var}[\sum_i w_i D_i]}$$
$$= \boldsymbol{\mu}^\mathsf{T}\mathbf{w} + \theta(\epsilon)\sqrt{\mathbf{w}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{w}}$$
$$= \boldsymbol{\mu}^\mathsf{T}\mathbf{w} + \theta(\epsilon)||\boldsymbol{\Sigma}^{1/2}\mathbf{w}||_2$$

Covariance matrix: symmetric, positive semi-definite

$\mathrm{cost}(\mathbf{w}) = \beta K(\mathbf{w})$ is a cone centered at $\mathbf{0}$

$\dfrac{\partial^2 \mathrm{cost}(\mathbf{w})}{\partial w_i \partial w_j} \approx 0$ if $\mathbf{w}$ is not zero and β is small

Satisfies Theorem 1, algorithm converges.

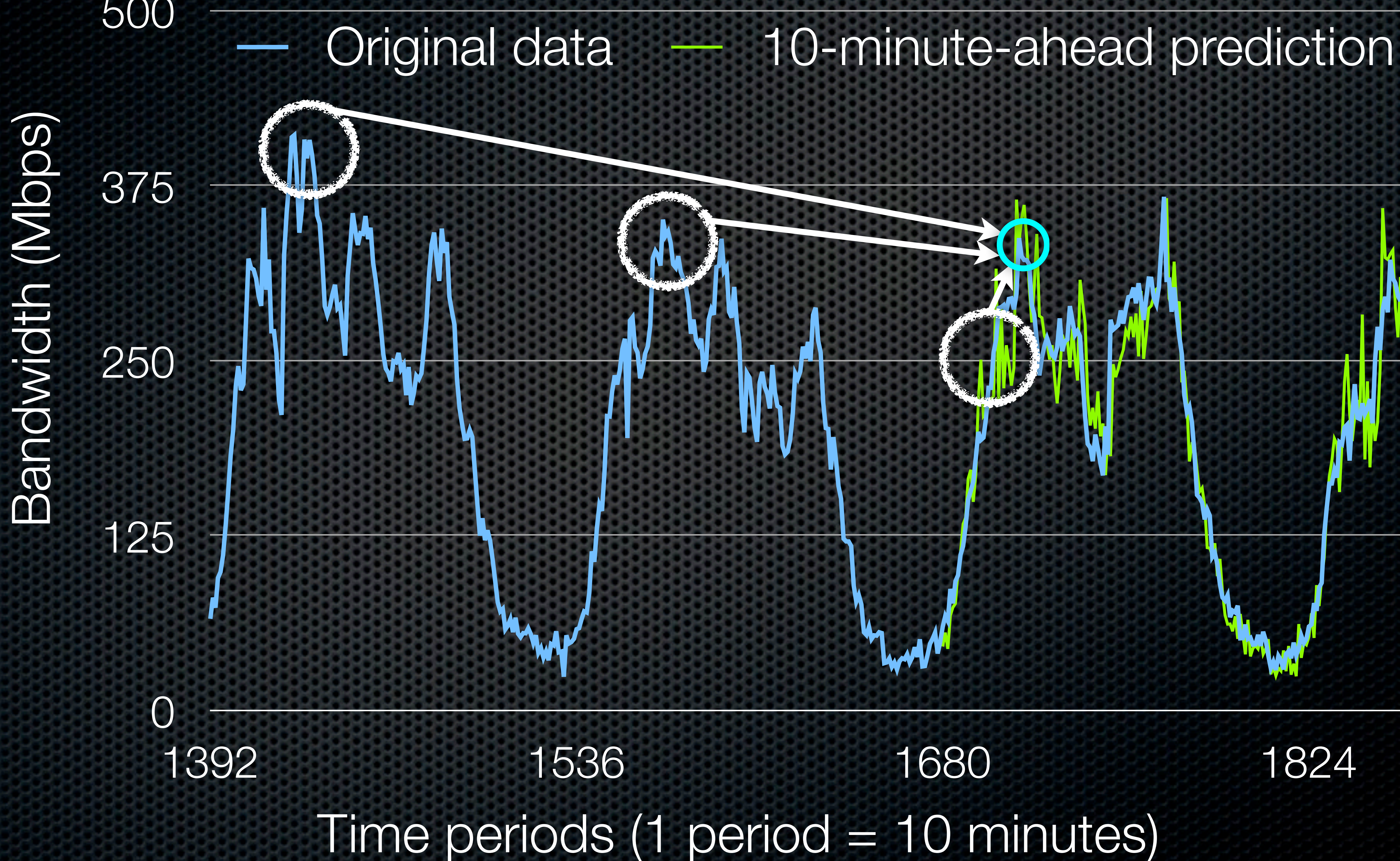
Roadmap

- ***Part 1*** A cloud bandwidth reservation model
- ***Part 2*** Price such reservations
  - Large-scale distributed optimization
- ***Part 3*** Trace-driven simulations

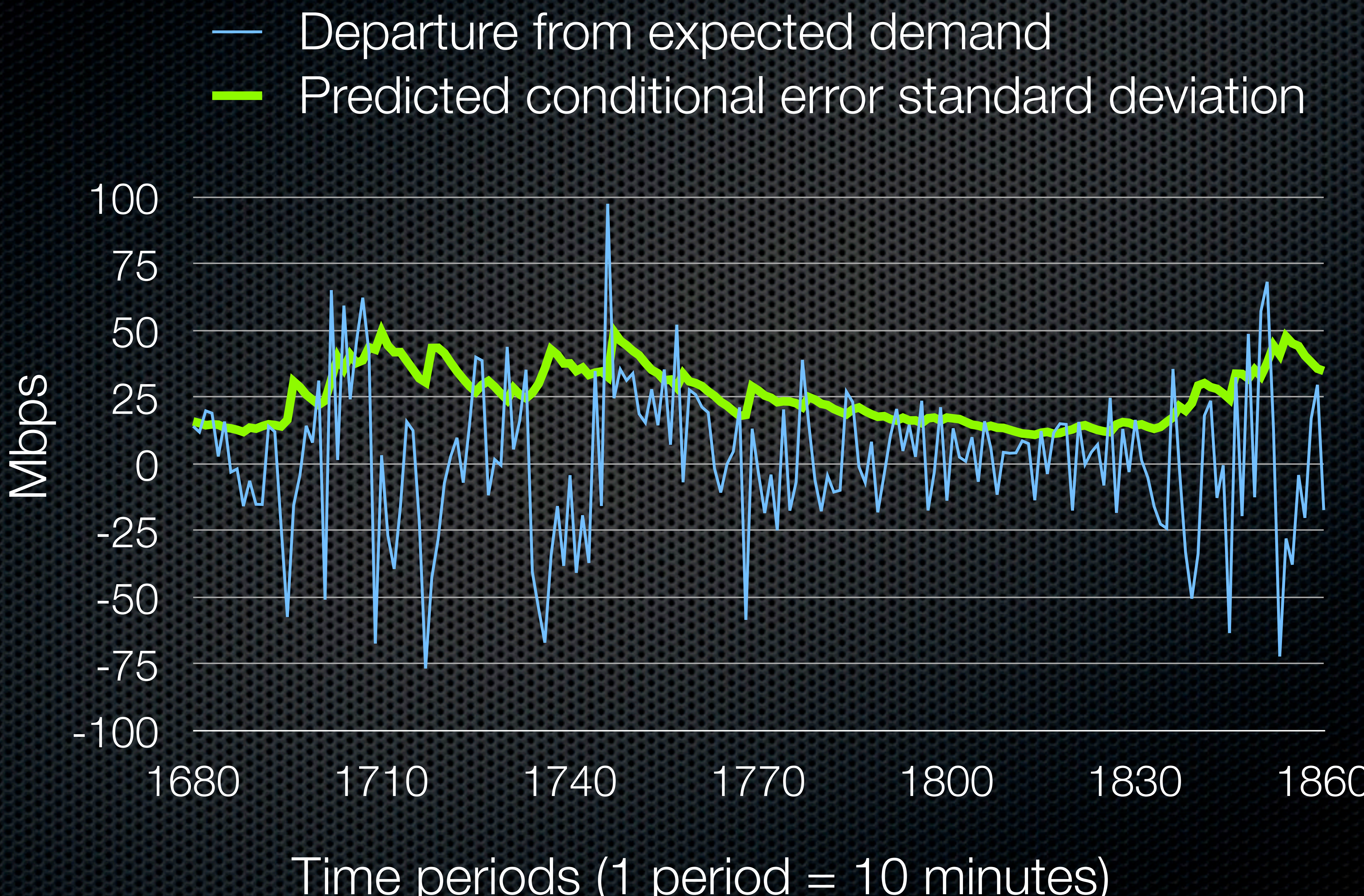
Data Mining: VoD Demand Traces

- 200+ GB traces (binary) from *UUSee Inc.*
  - reports from online users every 10 minutes
- Aggregate into video channels

Predict Expected Demand via *Seasonal ARIMA*



Predict Demand Variation via GARCH



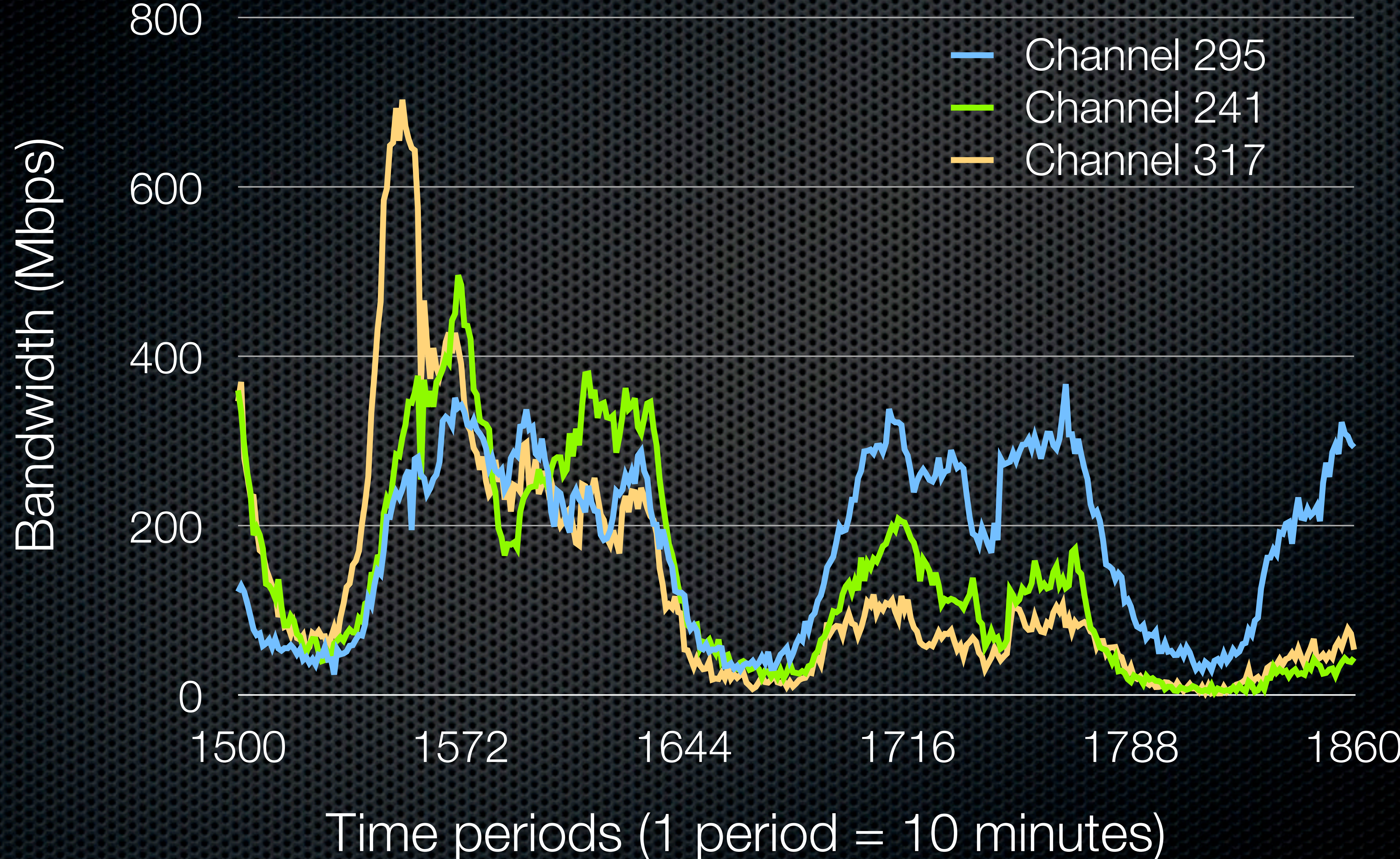
Prediction Results

- Each tenant i has a *random* demand D_i in each "10 minutes"
- D_i is Gaussian, with
  - mean $\mu_i = \mathrm{E}[D_i]$
  - variance $\sigma_i^2 = \mathrm{var}[D_i]$
  - covariance matrix $\Sigma = [\sigma_{ij}]$

Dimension Reduction via PCA

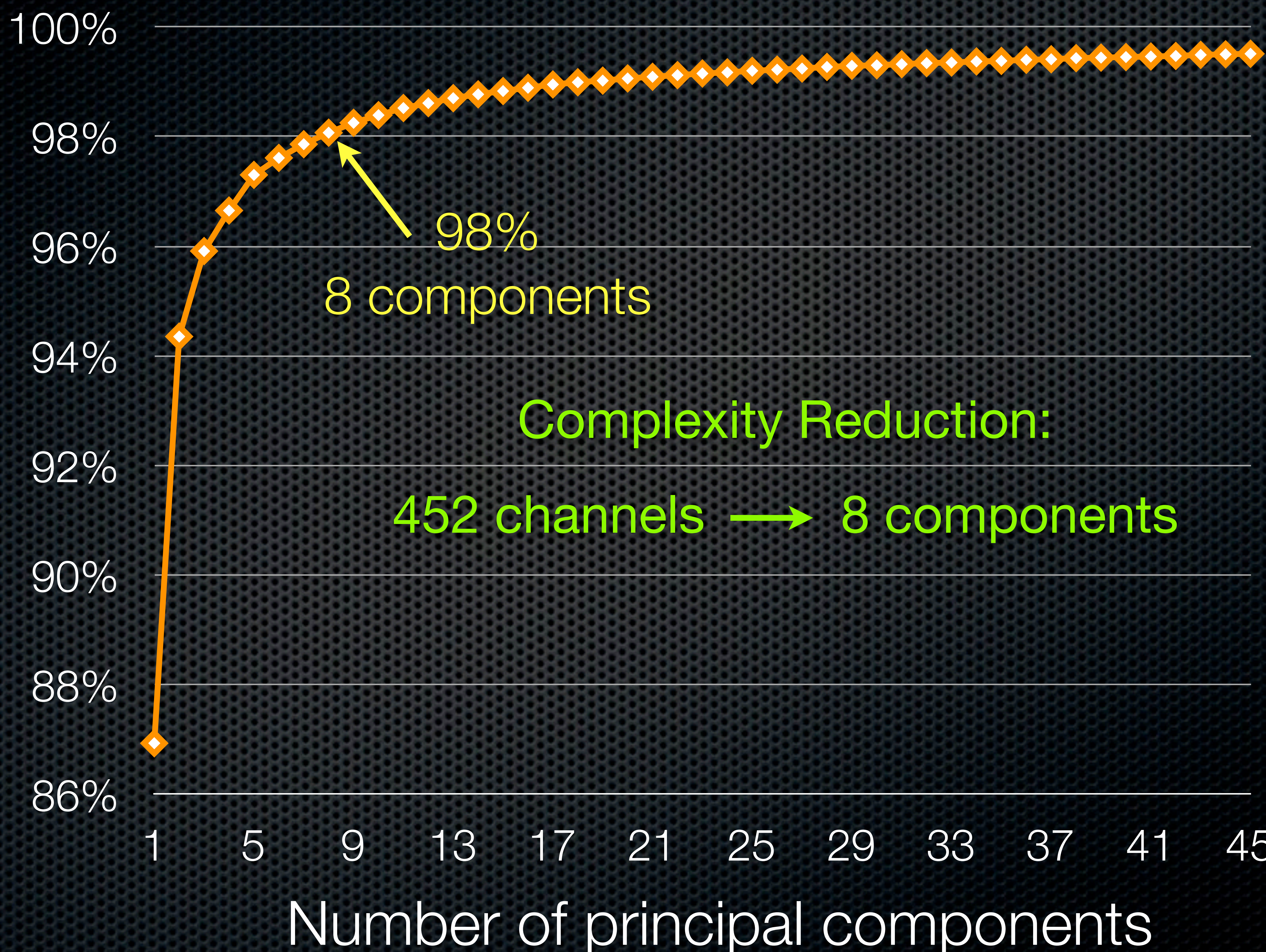
- A channel’s demand =
  weighted sum of factors
- Find factors using Principal Component Analysis (PCA)
- Predict factors first, then each channel

3 Biggest Channels of 452 Channels



The First 3 Principal Components

Data Variance Explained



Pricing: Parameter Settings

Usage of tenant i:

$$q_i(w_i) = w_i D_i \text{ w.h.p.}$$

Utility of tenant i (conservative estimate)

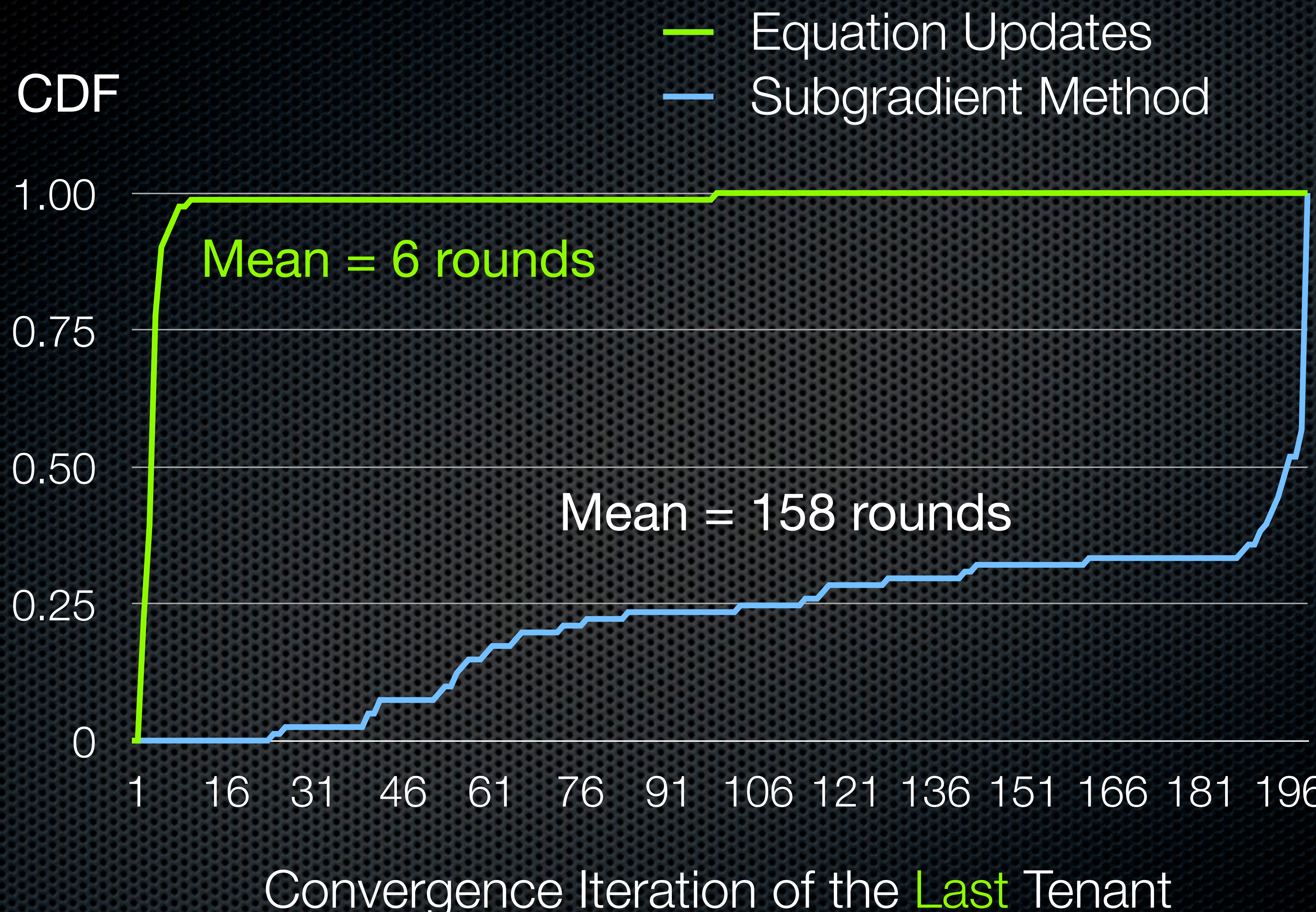
$$u_i\big(q_i(w_i), D_i\big) = \alpha_i q_i(w_i) - e^{A_i(D_i - q_i(w_i))},$$

Linear revenue

Reputation loss for demand not guaranteed

$$\mathbf{E}[u_i(w_i)] = \alpha_i w_i \mu_i - e^{A_i(1-w_i)\mu_i + \frac{1}{2}A_i^2(1-w_i)^2\sigma_i^2}.$$

$$\alpha_i = 1, A_i = 0.5, \beta = 0.5, \epsilon = 0.01$$



100 tenants (channels), 81 time periods (81 x 10 Minutes)

Related Work

- Primal/Dual Decomposition [Chiang *et al.* 07]
- Contraction Mapping $x := T(x)$
  - D. P. Bertsekas, J. Tsitsiklis, "Parallel and distributed computation: numerical methods"
- Game Theory [Kelly 97]
  - Each user submits a price (bid), expects a payoff
  - Equilibrium *may* or *may not* be social optimal
- Time Series Prediction
  - HMM [Silva 12], PCA [Gürsun 11], ARIMA [Niu 11]

Conclusions

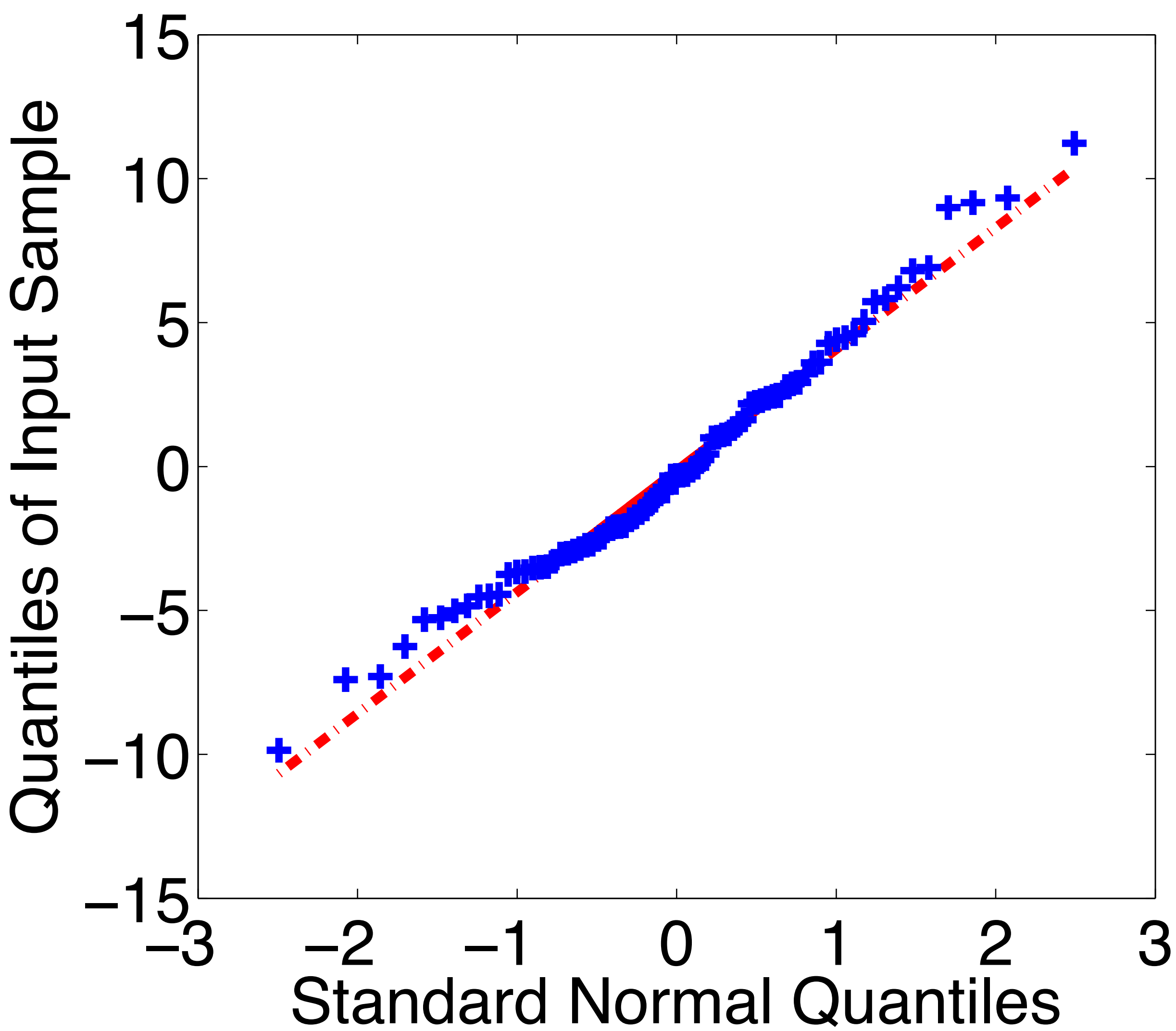
- A cloud bandwidth reservation model based on *guaranteed portions*
- Pricing for social welfare maximization
- Future work:
  - new decomposition and iterative methods for very large-scale distributed optimization
  - more general convergence conditions

Thank you

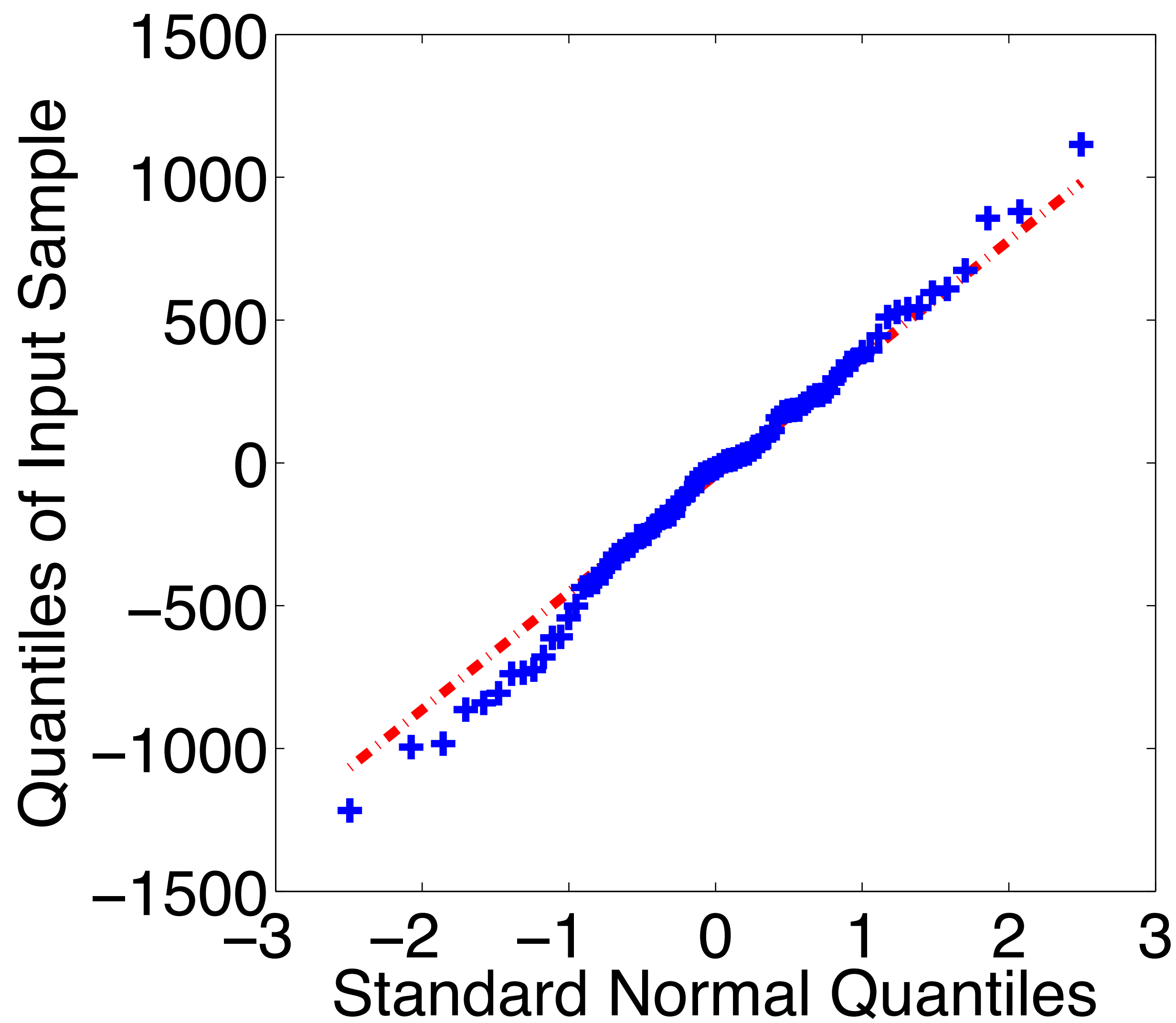
Di Niu

Department of Electrical and Computer Engineering
University of Toronto

http://iqua.ece.toronto.edu/~dniu

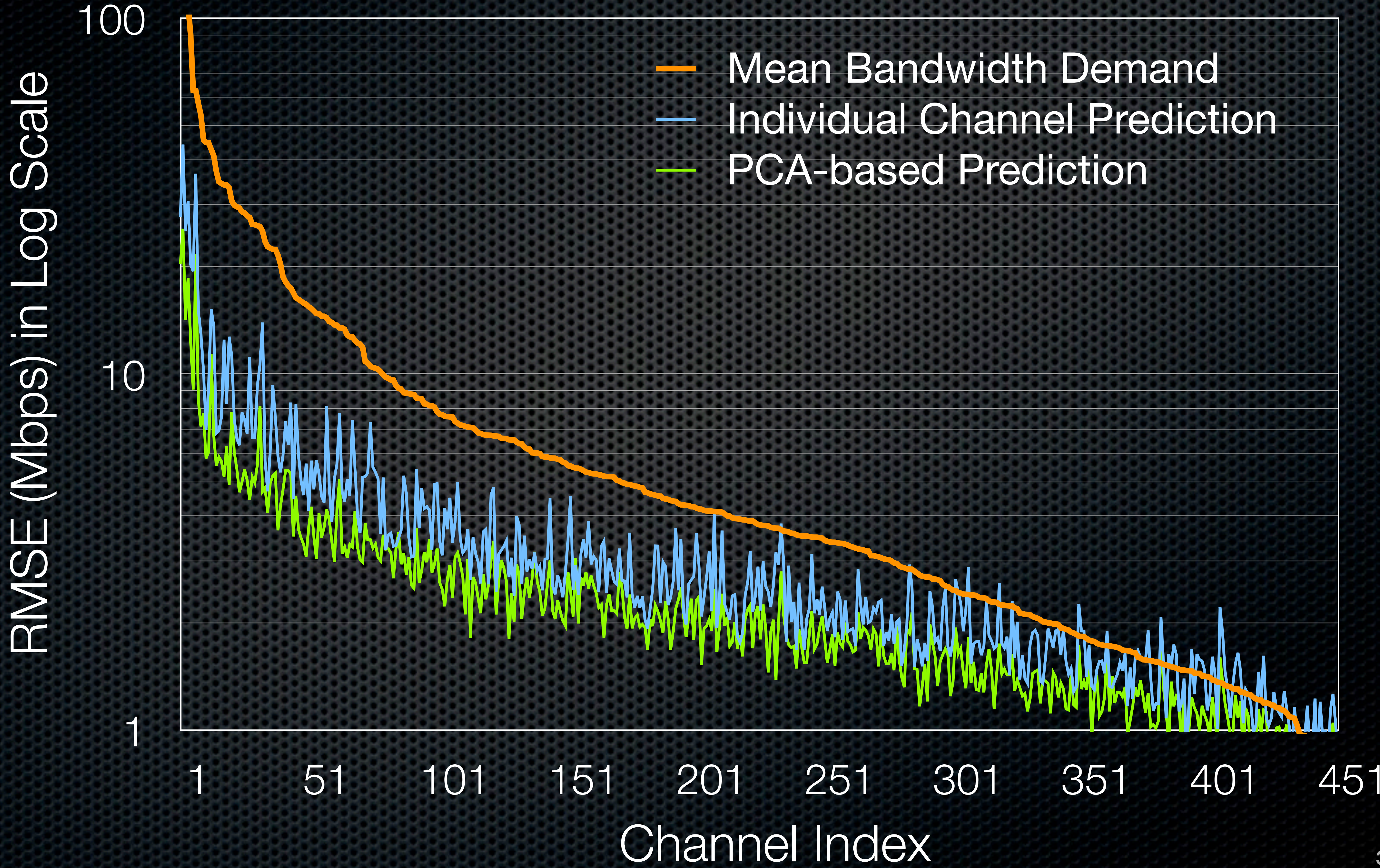


(a) Z_{it} of a typical channel



(b) $\sum_i Z_{it}$ of all channels combined

Root mean squared errors (RMSEs) over 1.25 days



Optimal Pricing when each tenant requires $w_i \equiv 1$

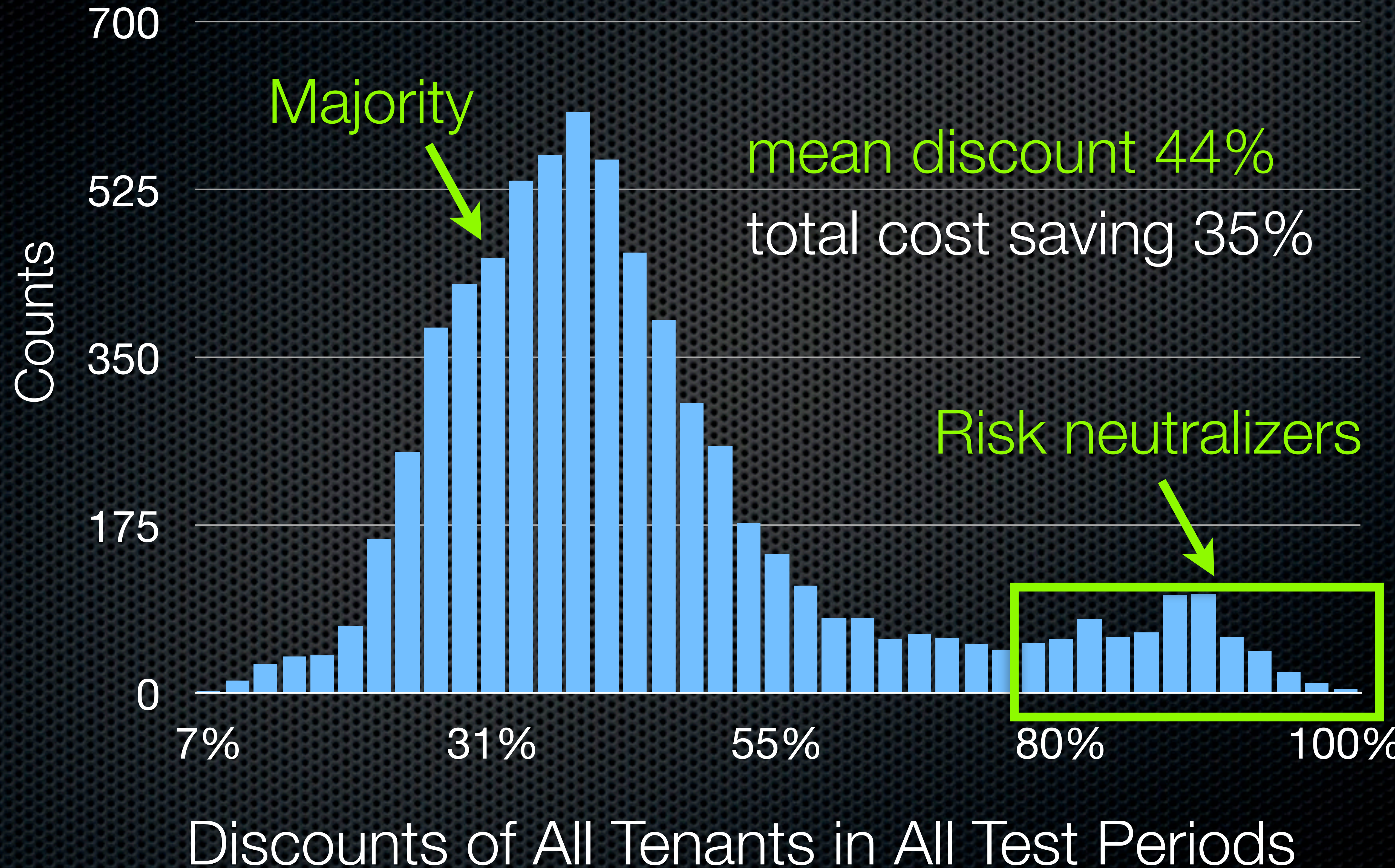
Without multiplexing,

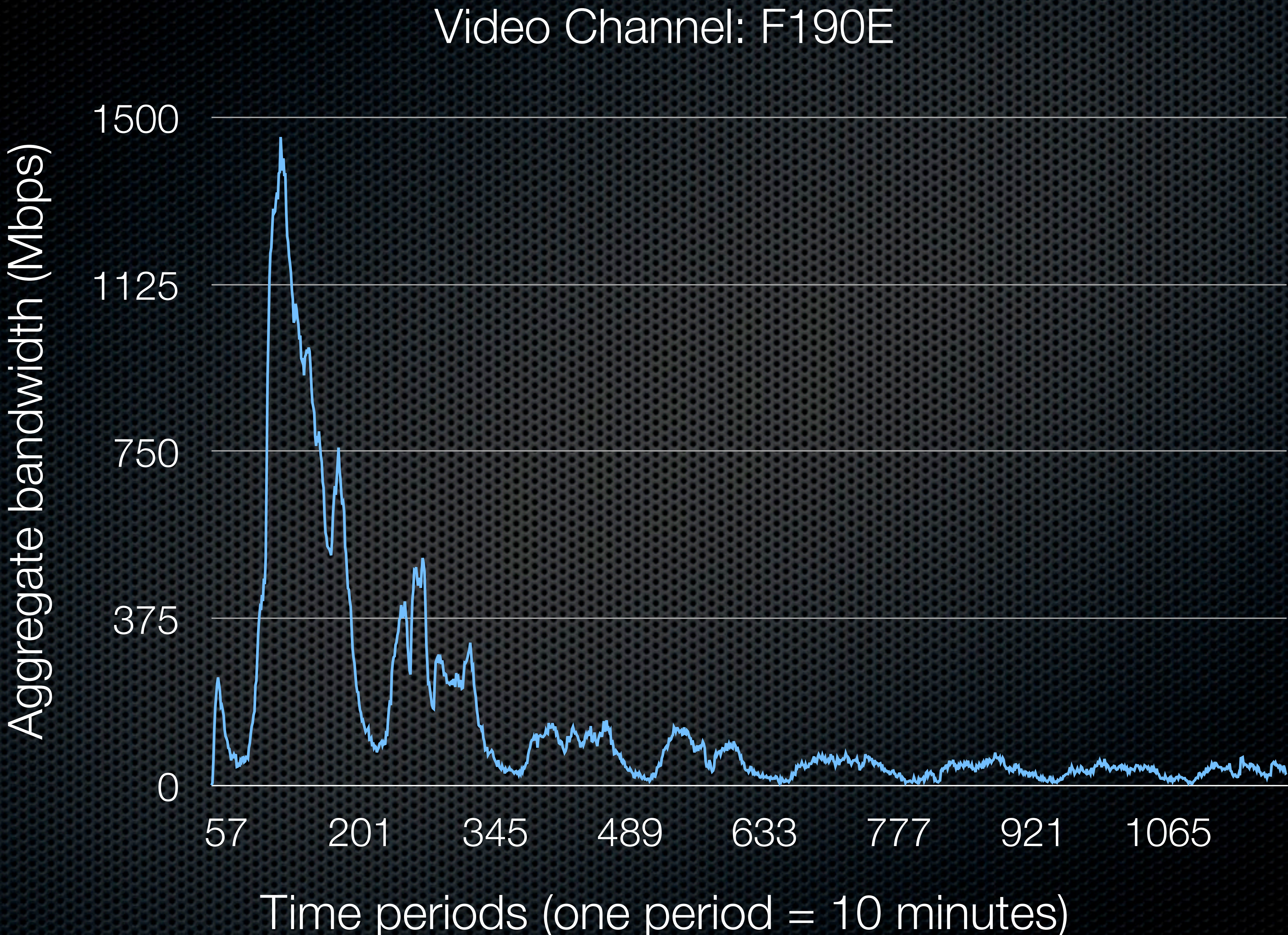
$$P_i^*(1) = \mu_i + \theta(\epsilon)\sigma_i$$

With multiplexing,

$$P_i^*(1) = \underbrace{\mu_i}_{\text{Expected Demand}} + \theta(\epsilon)\underbrace{\sigma_i}_{\text{Demand Standard Deviation}}\overbrace{\rho_{iM}}^{\text{Correlation to the market, in [-1, 1]}}$$

Histogram of Price Discounts due to Multiplexing




Video Channel: F190E
Aggregate bandwidth (Mbps)
1500
1125
750
375
0
57
201
345
489
633
777
921
1065
Time periods (one period = 10 minutes)