Online Resource Management for Data Center with *Energy Capping*

Hasan Mahmud and Shaolei Ren
Florida International University
A massive data center

- Facebook's data center in Prineville, OR
Three pieces of old news

• **2005**: EU introduced carbon emission caps to large energy consumers
  – “Cap and trade”: if cap exhausted, then buy more credits
• **2007**: eBay paid $79K fine to Sacramento, CA, for using generators and polluting air
• **2011**: Microsoft faced $210K penalty from Quincy, WA, utilities for overestimating its energy usage
  – Waived!
News!

• **2013:** China to impose carbon targets by 2016

Courtesy of The Independent
Energy cap!

• There is an *energy cap*
  – Penalty for exceeding the cap
  – Stricter energy caps are anticipated in light of the increasingly serious sustainability concerns

• In order to satisfy the cap, data centers need to carefully use their energy quota
Energy cap!

• There is an *energy cap*
  – Penalty for exceeding the cap
  – Stricter energy caps are anticipated in light of the increasingly serious sustainability concerns

• In order to satisfy the cap, data centers need to carefully use their energy quota

Energy budgeting
Power v.s. Energy

• Power budgeting
  – Peak power is costly to increase and hence often oversubscribed
  – Maximize performance given peak power constraint [1][2]

Power v.s. Energy

• Power budgeting
  – Peak power is costly to increase and hence often oversubscribed
  – Maximize performance given peak power constraint [1][2]

• Energy budgeting

Solution

• Turn on as few servers as possible to satisfy QoS
  – But, what should be the energy cap?
  – “Energy oversubscription”
    • Like what Microsoft did for its Quincy, WA, data center
    • Clearly, *not good* for power utilities
Another solution

- Plan everything ahead, assuming that we know everything about the future (e.g., workloads, renewables, etc.) [3]
  - How can we accurately predict the future?
  - Hour-ahead or day-ahead traffic/renewables prediction may be good, but month-ahead or even season-ahead predictions may NOT be!

Our proposal

• Realizing...
  – Long-term prediction may not be accurate

• Why not just give a rough estimate in advance and then try to follow your target online?

• **Challenge**
  – We have long-term target, but we only have short-term information
Our proposal

• Realizing...
  – Long-term prediction may not be accurate

• Why not just give a rough estimate in advance and then try to follow your target online?

• **Challenge**
  – We have long-term target, but we only have short-term information

Do it by tracking your energy usage online!
Model

• Time-slotted model
• Data center has $M$ homogeneous servers
  – On-site renewable energy available
  – Capacity provisioning decisions are made at the beginning of each time slot
  – Service process at each server is modeled by a FIFO queue
Objectives

• Electricity bill
  – Reduced by using fewer servers

• QoS
  – Response time
  – QoS can be increased by using more servers

Cost savings versus user experiences
Formulation

• Costs
  – Electricity cost: $e(\lambda, m)$
  – Delay cost: $d(\lambda, m)$

• Total cost is given by

$$g(\lambda(t), m(t)) = e(\lambda(t), m(t)) + \beta \cdot d(\lambda(t), m(t))$$

• Energy capping target

$$\frac{1}{K} \sum_{t=0}^{K-1} \left[ p(\lambda(t), m(t)) - r(t) \right]^+ \leq \frac{Z}{K}$$

– $r(t)$ is the available on-site renewables
Online resource management

• Construct an energy deficit queue

\[ q(t + 1) = \left\{ q(t) + \left[ p(\lambda(t), m(t)) - r(t) \right]^+ - z \right\}^+ \]

  – Queue length indicate the energy budget deficit

• Instead of minimizing the cost, minimize the following

\[ V \cdot g(\lambda(t), m(t)) + q(t) \cdot \left[ p(\lambda(t), m(t)) - r(t) \right]^+ \]

  – Queue length gives additional weight on electricity usage
  – Larger queue means: more energy is used than allowed budget
  – **Insight:** if exceeds, then reduce!
Algorithm analysis

• Prove the following two facts
  – *Good* cost compared to the optimal offline algorithm with future information
  – *Approximately* satisfy energy capping

• Proof technique
  – Recently-developed Lyapunov optimization
  – Relax i.i.d./Markovian assumptions to arbitrary dynamics
Algorithm analysis

• Prove the following two facts
  – *Good* cost compared to the optimal offline algorithm with future information
  – *Approximately* satisfy energy capping

• Proof technique
  – Recently-developed Lyapunov optimization
  – Relax i.i.d./Markovian assumptions to arbitrary dynamics
Case study
Simulation

- 50MW data center
- 6-month energy budgeting
- Hour-ahead prediction
Simulation

(a) Cost versus V.

(b) Budget deficit versus V.

Achieving low cost while satisfying budget!
Comparison

• **Prediction-based:**
  – Predict the next-day workload perfectly and allocate the daily energy budget in proportion to the hourly workloads

• **9% cost reduction only using hour-ahead prediction!!**
Impact of energy budget

• Average cost of ORM increases when the energy budget decreases
• With 90% energy budget, average cost ORM only exceeds by approximately 3%
Impact of energy budget

Increasing the operational cost **marginally** but reduce energy **significantly**
Conclusion

• ORM is a provably-efficient online energy budgeting algorithm using only short-term prediction (e.g., hour-ahead)
Conclusion

*Budgeting* energy for sustainability!
Thanks!