Algorithms for sustainable data centers

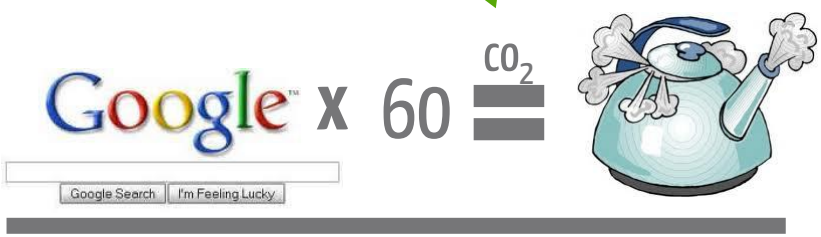
Adam Wierman (Caltech)
Minghong Lin (Caltech)
Zhenhua Liu (Caltech)
Lachlan Andrew (Swinburne)
and many others...

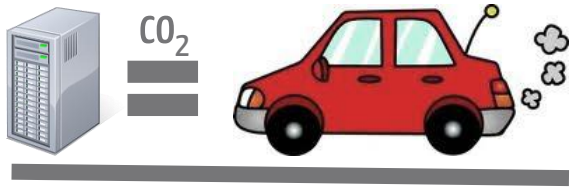
Microsoft Research

facebook

IT is an energy hog

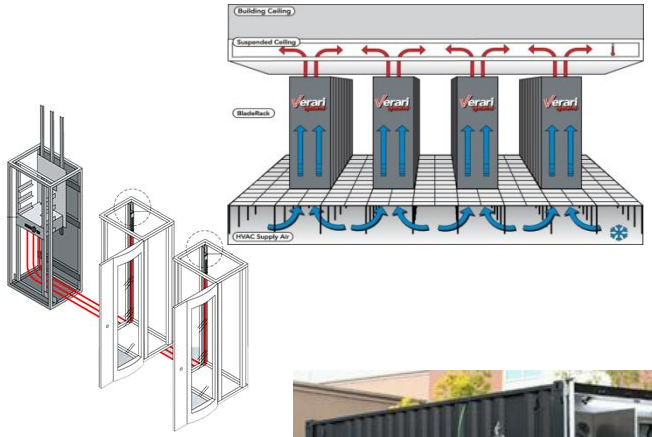
The electricity use of data centers is ≈2-3% of the US total,
and it is growing ≈12% a year! Total US use grows ~1% a year!

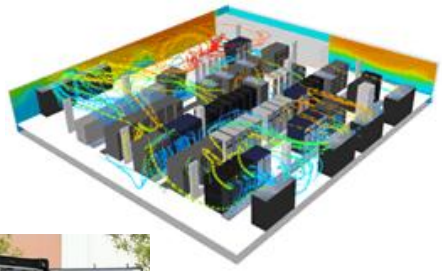




Huge push from academia & industry

Lots of engineering improvements





$$\text{Power Usage Effectiveness (PUE)} = \frac{\text{Power to building}}{\text{Power to computing}}$$

$$>3 \longrightarrow <1.1$$

lower PUE $\neq$ lower energy consumption
$\neq$ more sustainable

$$\text{Power Usage Effectiveness (PUE)} = \frac{\text{Power to building}}{\text{Power to computing}}$$

$>3 \longrightarrow <1.1$

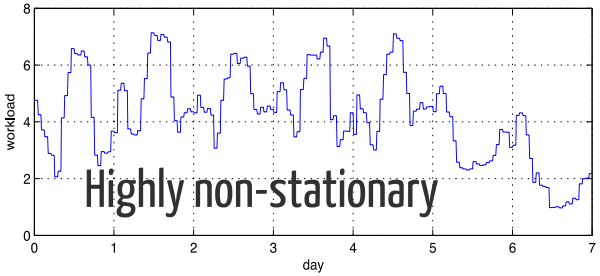
The building is efficient,
it's time to work on the algorithms...

This talk: An overview of algorithmic questions & challenges

1. Two examples
2. A general model & some results
3. A case study

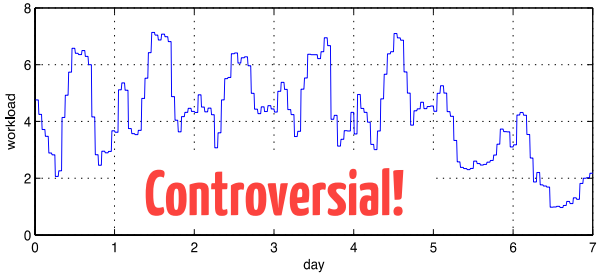
Example 1: Dynamic resizing in a data center

Workload



Idling servers use
30-50% of peak power

Active Servers

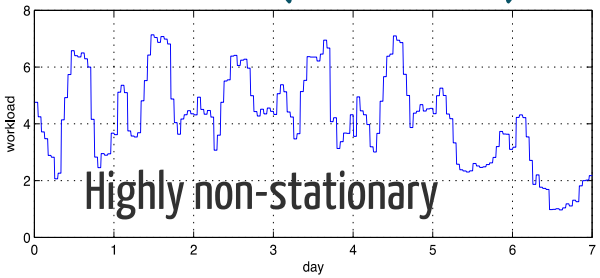


Goal: Adapt provisioning to minimize cost.

Challenges: Switching is costly & prediction is hard.

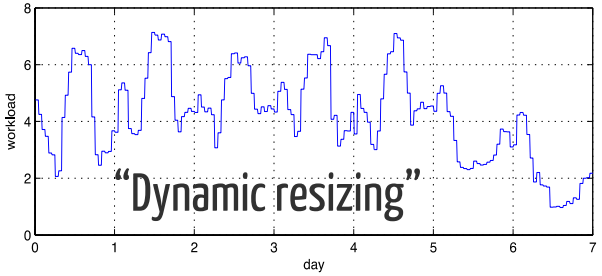
Example 1: Dynamic resizing in a data center

Workload (Interactive)



Idling servers use
30-50% of peak power

Active Servers

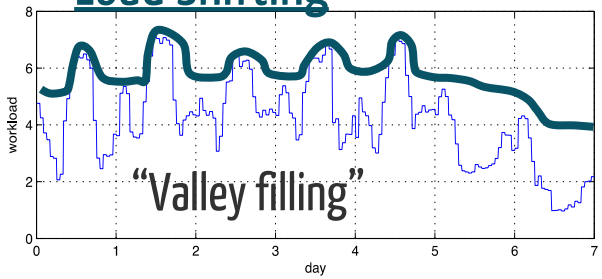


Workload (Batch)

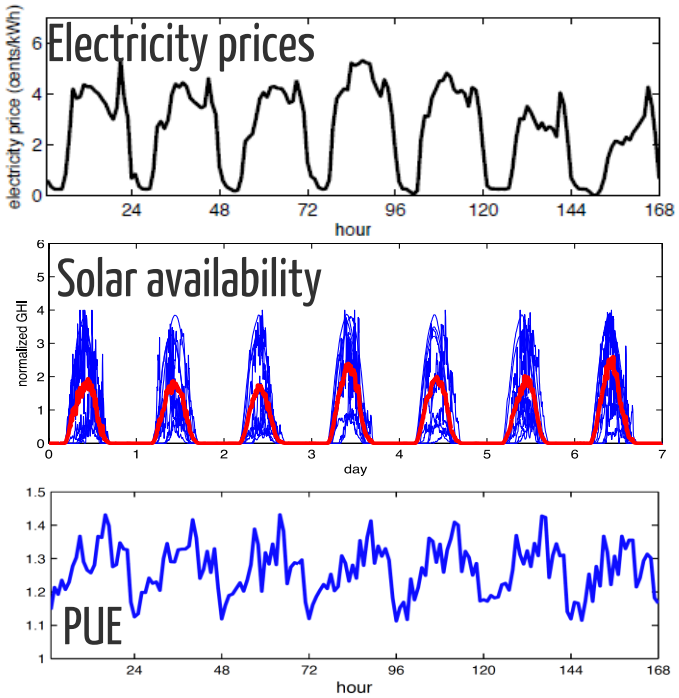
Large, delay tolerate jobs,
typically with deadlines

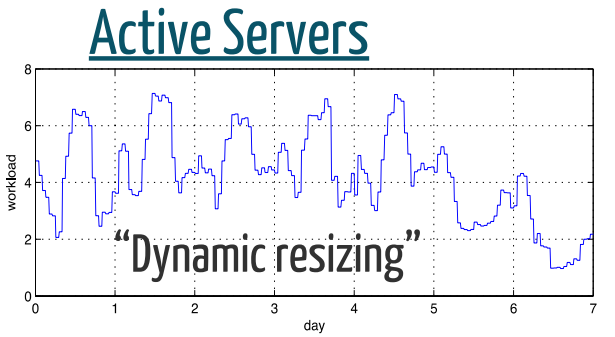
Deadlines are often
on the order of hours

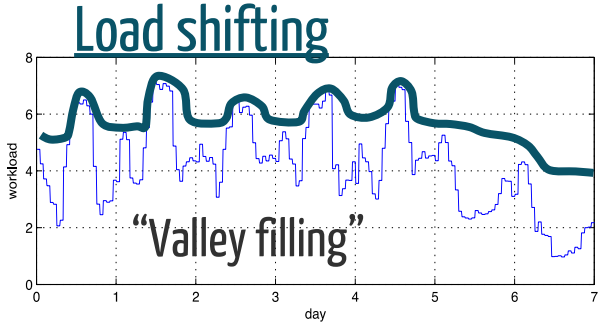
Load shifting



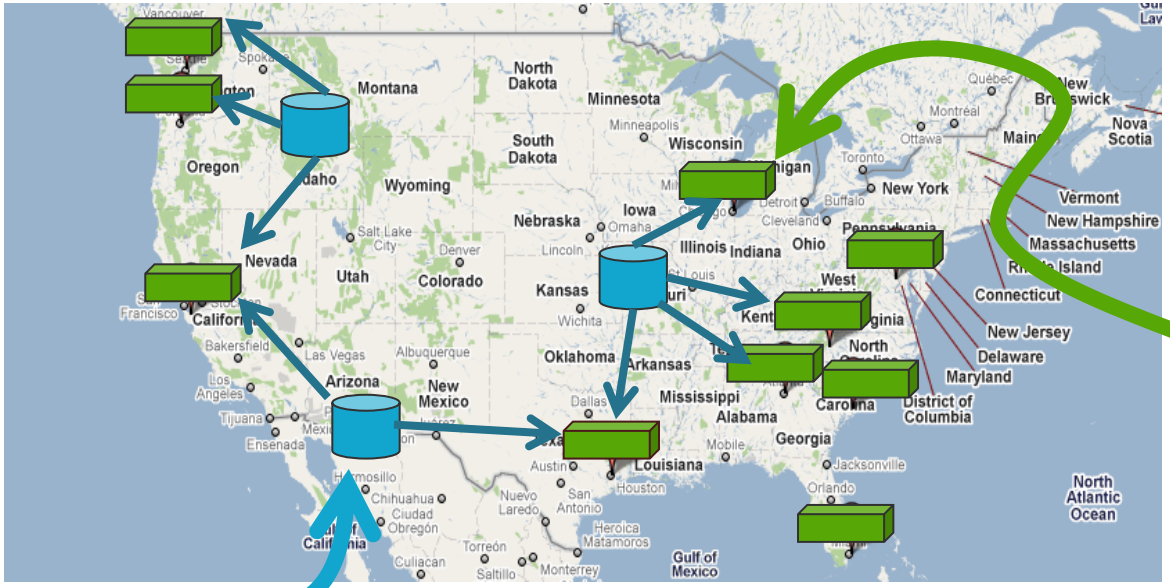
Example 1: Dynamic resizing in a data center





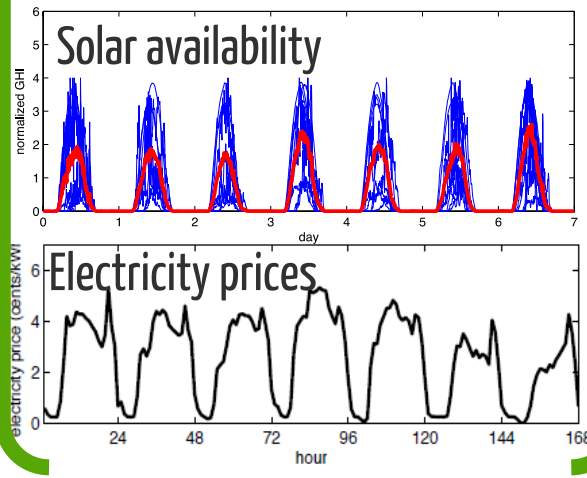


Example 2: Geographical load balancing

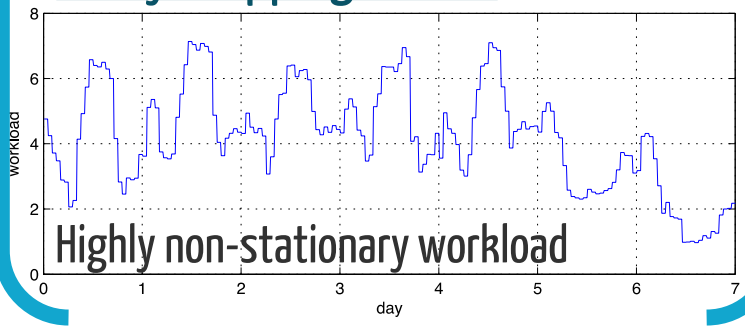


Data centers:

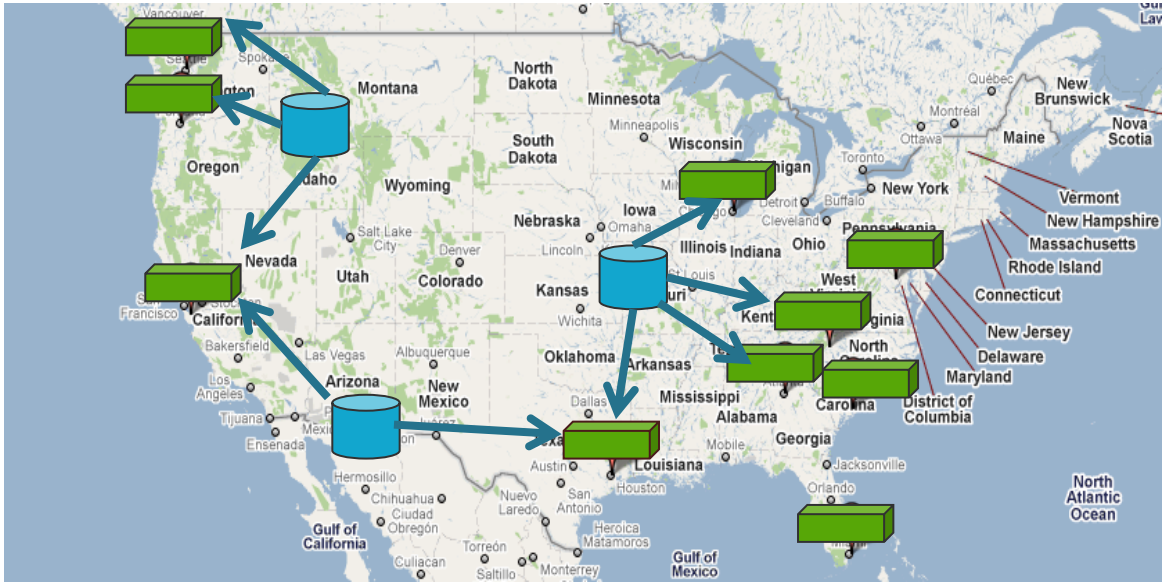
Diverse, time varying electricity prices, renewable availabilities, cooling efficiencies, etc.



Proxy/Mapping Nodes:



Example 2: Geographical load balancing



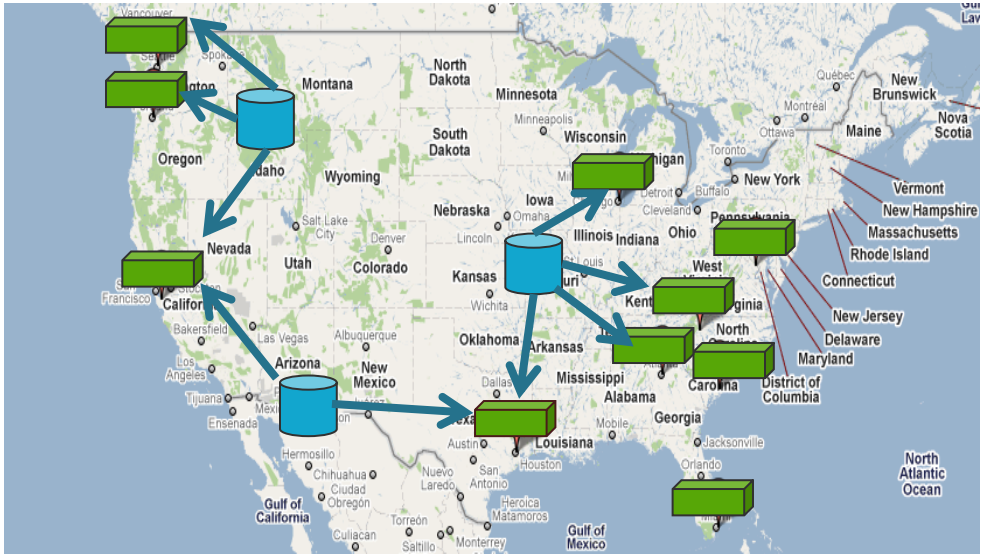
Goal: Adapt routing & provisioning to minimize cost.

Challenges: Switching is costly & prediction is hard.

Dynamic resizing

[Chase et al. 01] [Pinheiro et al. 01] …[Chen et al. 05] …[Ghandi et al. 09] [Khargharia et al. 10] [Kansal et al. 10] …[LWAT10] …

Geographical load balancing

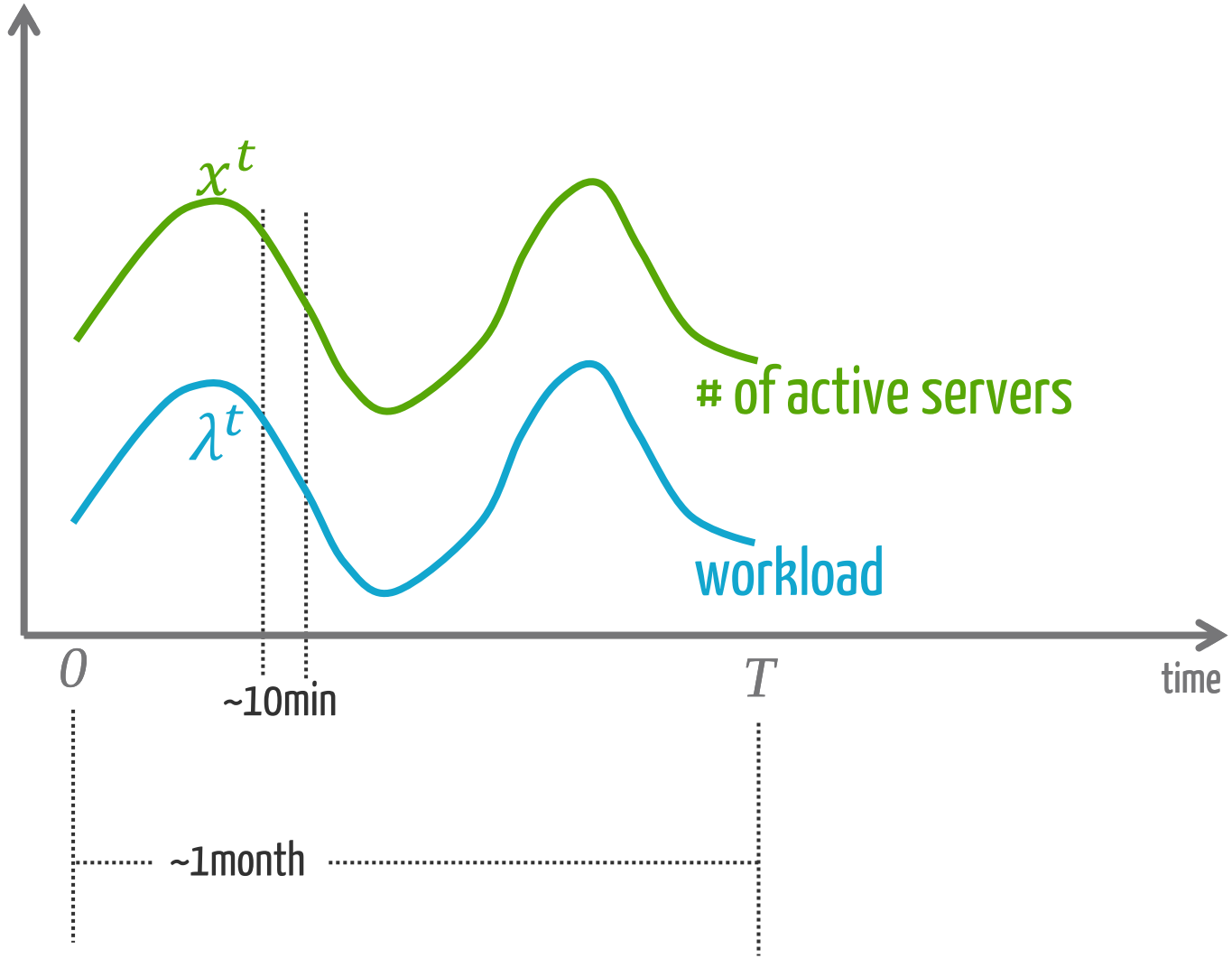


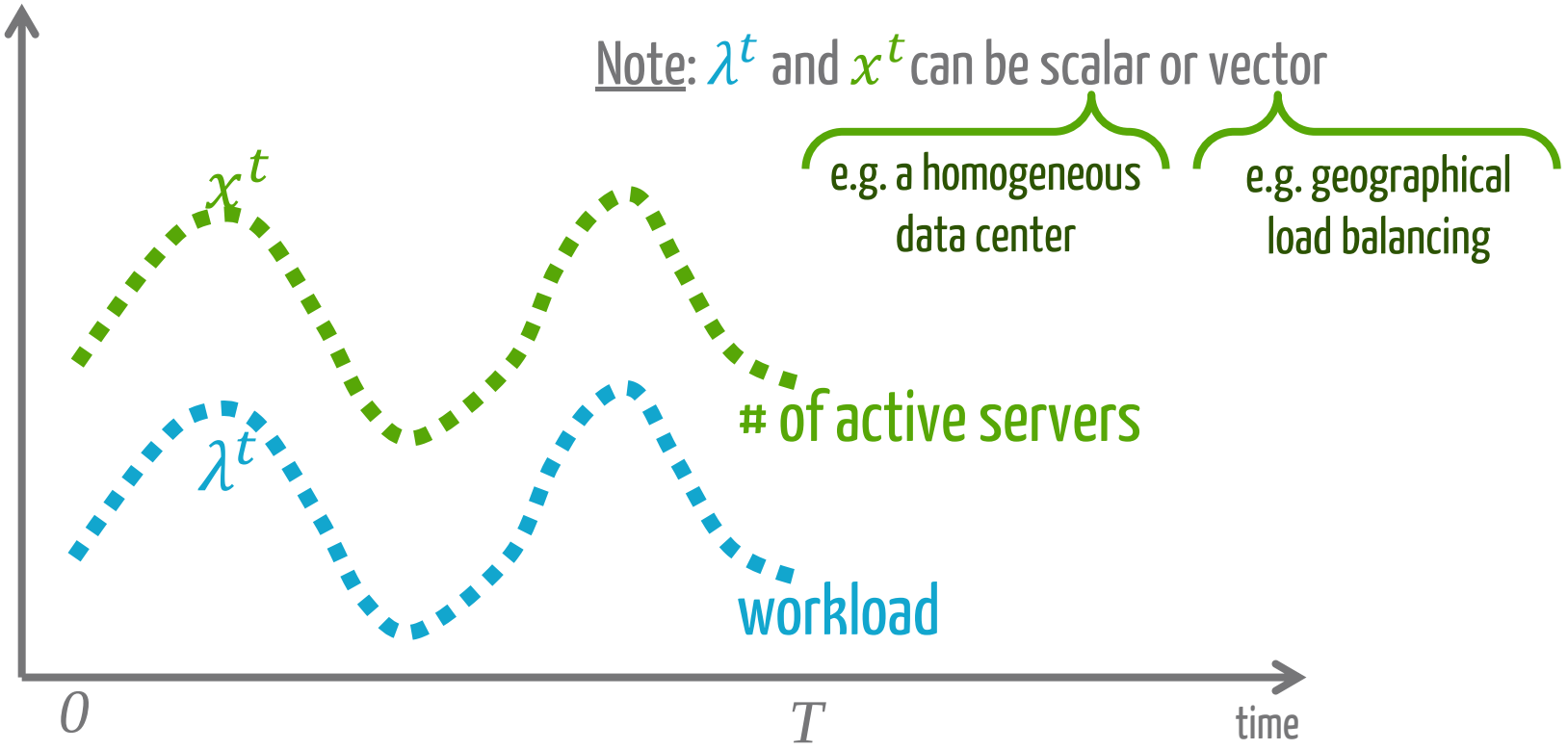
[Pakbaznia et al. 09] [Qureshi et al. 09] [Rao et al. 10] [Stanojevic et al. 10] [Wendell et al 10] [Le et al 2010] [LLWLA 11] [LLWAL11] [LAW12], …

Goal: Adapt routing & provisioning to minimize cost.

Challenges: Switching is costly & prediction is hard.

A (familiar) model


x^t
λ^t
# of active servers
workload
0
~10min
T
time
~1month


Note: λ^t and x^t can be scalar or vector
e.g. a homogeneous data center
e.g. geographical load balancing
x^t
# of active servers
λ^t
workload
0
T
time

Data center goal:

$$\min_{x^t \in F} \sum_t (\text{operating cost}) + (\text{switching cost})$$

(stability constraints)

Data center goal:

$$\min_{x^t \in F} \sum_t \text{(operating cost)} + \text{(switching cost)}$$

stability constraints

~~$x_t \in \mathbb{N}$~~

$N_{max} \geq x_t \geq 0$

$x_t \geq SLA(\lambda_t)$

...

Data center goal:

$$\min_{x^t \in F} \sum_t (\text{operating cost}) + (\text{switching cost})$$

Data center goal:

$$\min_{x^t \in F} \sum_t c^t(x^t) + \|x^t - x^{t-1}\|$$

E.g. A homogeneous data center

$$c^t(x^t; \lambda^t) = \min_{\lambda_{it}} \sum_{i=1}^{x_t} g^t(\lambda_i^t)$$

$g_t(l) :=$ (delay) + (energy cost)

E.g. Cost to switch server on/off

$$\beta \cdot |x^t - x^{t-1}|$$

Data center goal:

$$\min_{x^t \in F} \sum_t c^t(x^t) + \|x^t - x^{t-1}\|$$

convex

Stochastic / Adversarial

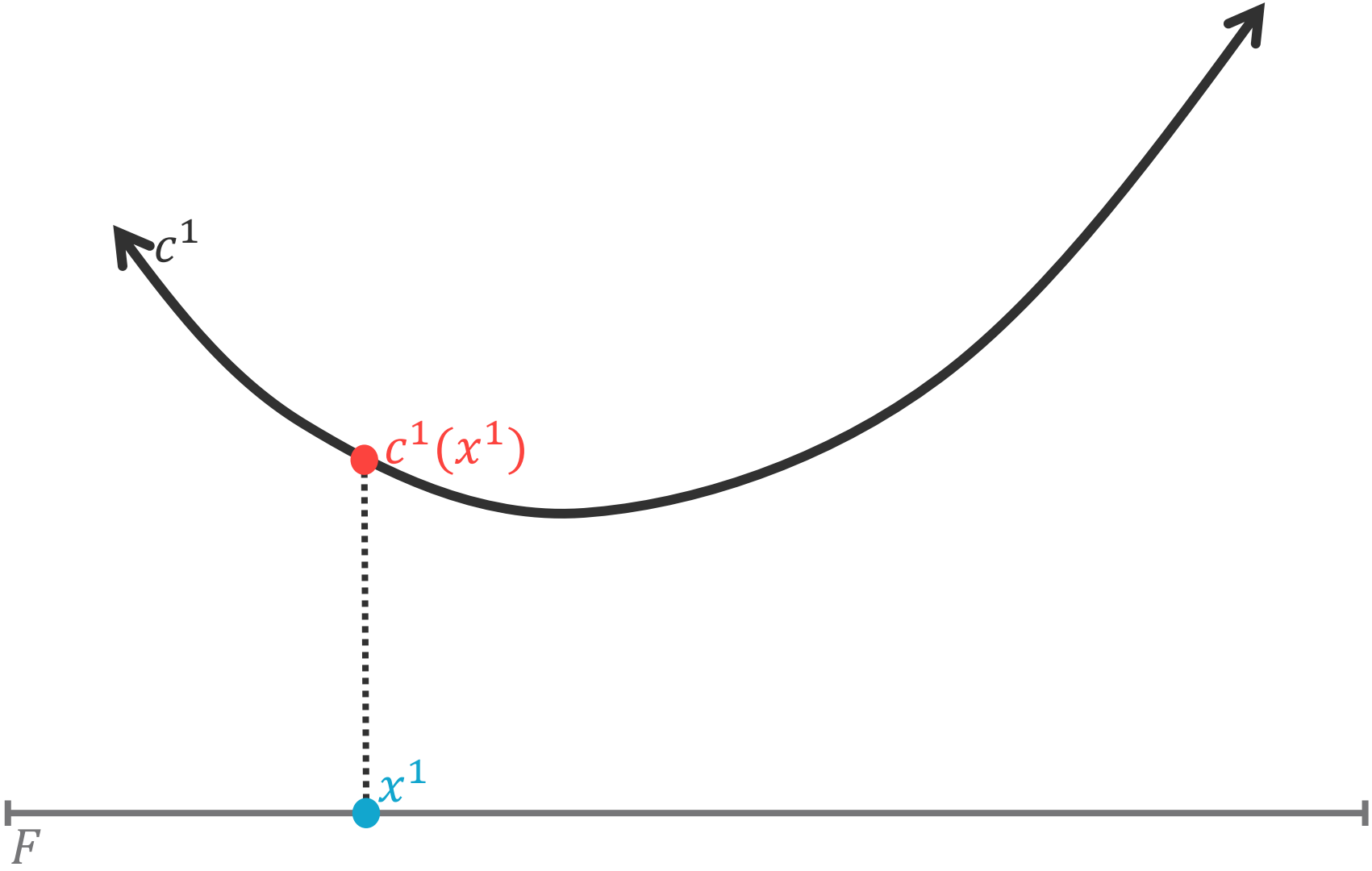
$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

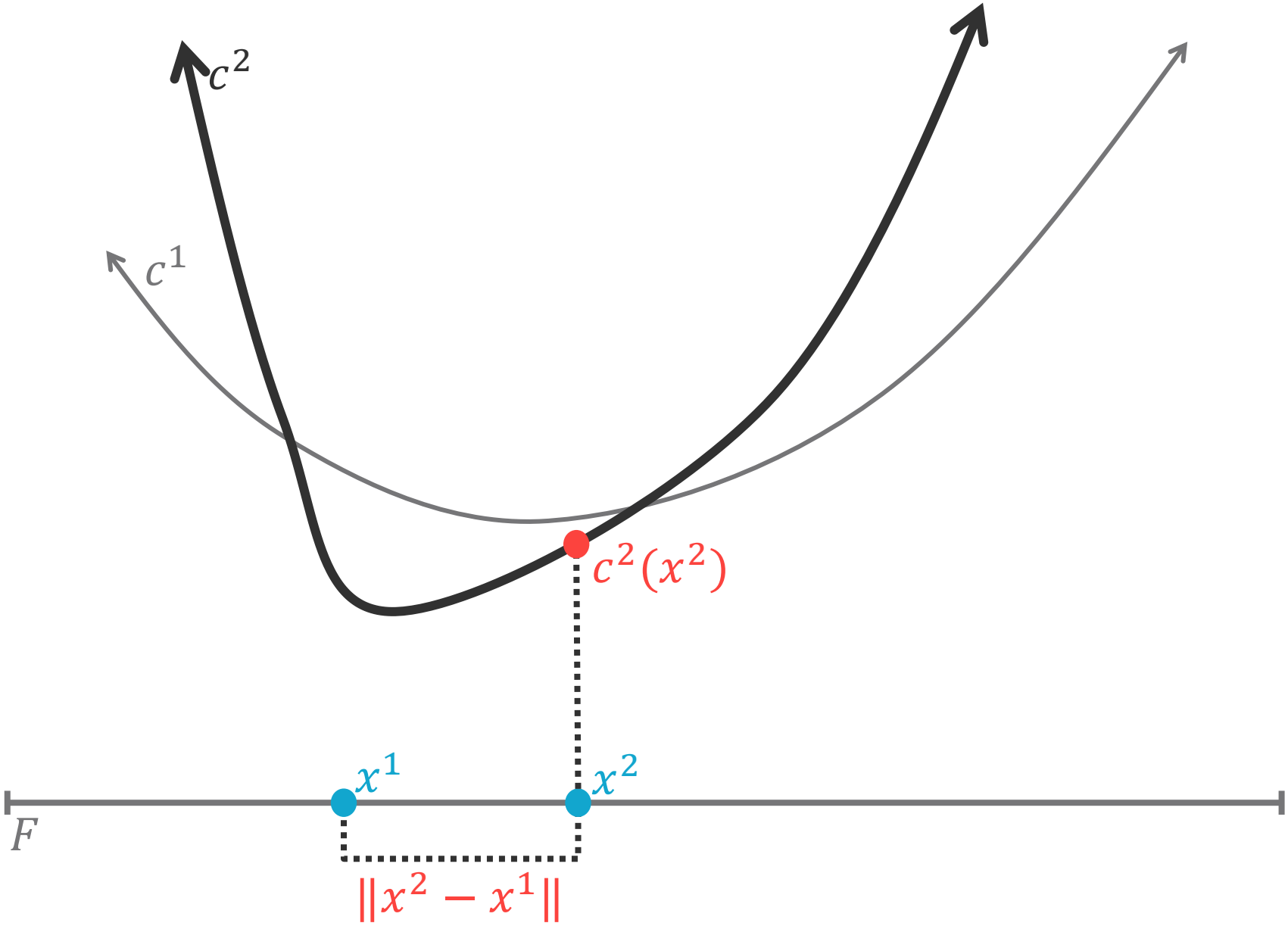
$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Goal: Algorithms to minimize cost

Smoothed online convex optimization (SOCO)


c^1
$c^1(x^1)$
x^1
F


c^2
c^1
$c^2(x^2)$
x^1
x^2
F
$\|x^2 - x^1\|$

SOCO comes up in many applications ...

geographical load balancing
dynamic capacity management
video streaming
electricity generation planning
product/service selection
portfolio management
labor markets
penalized estimation

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Goal: Algorithms to minimize cost

Smoothed online convex optimization (SOCO)

2 Communities

- Online Learning
- Online Algorithms

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Goal: Algorithms to minimize cost

Smoothed online convex optimization (SOCO)

2 Communities

- Online Learning
- Online Algorithms

2 Metrics

- Regret
- Competitive ratio

$$\textbf{Regret(Alg)} = \text{Cost(Alg)} - \text{Cost(Static\_Opt)}$$

$$\textbf{Competitive ratio(Alg)} = \frac{\text{Cost(Alg)}}{\text{Cost(Offline\_Opt)}}$$

Can an algorithm maintain sub-linear regret?
[KV02] [BBCM03] [Z03] … [HAK07]… [LRAMW12]

$$\textbf{Competitive ratio(Alg)} = \frac{\text{Cost(Alg)}}{\text{Cost(Offline\_Opt)}}$$

Can an algorithm maintain sub-linear regret?
[KV02] [BBCM03] [Z03] … [HAK07]… [LRAMW12]

Can an algorithm maintain a constant competitive ratio? [BKRS92] [BLS92]…[BB00]…[LWAT11] [LRAMW12]

Can an algorithm maintain sub-linear regret and a constant competitive ratio?

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Goal: Algorithms with sub-linear regret

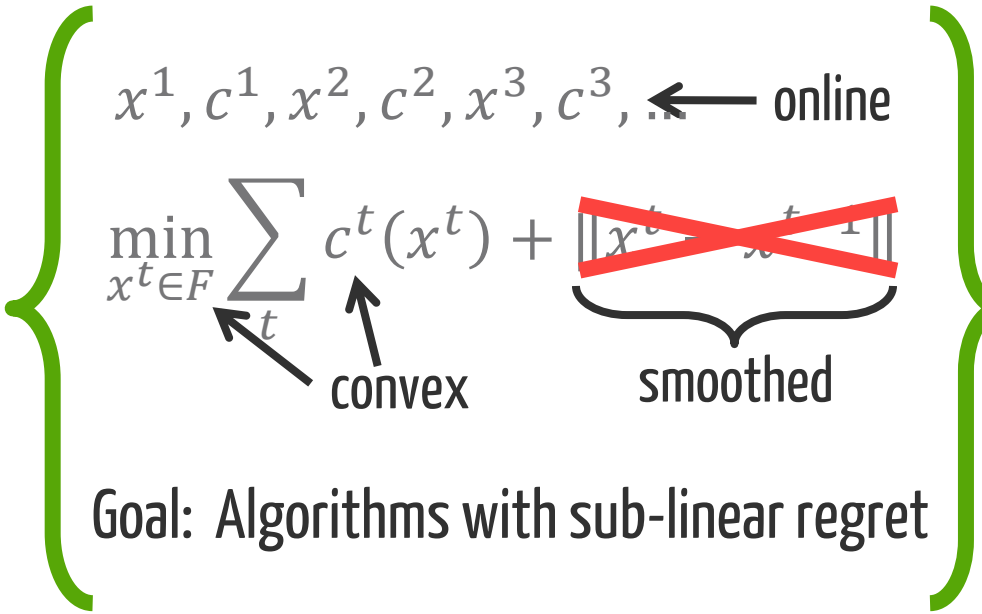
Smoothed online convex optimization (SOCO)

2 Communities

- **Online Learning**
- Online Algorithms

2 Metrics

- **Regret**
- Competitive ratio



Online convex optimization

2 Communities

Online Learning

Online Algorithms

2 Metrics

Regret

Competitive ratio

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$

$\min_{x^t \in F} \sum_t c^t(x^t)$

$\|\nabla c^t(\cdot)\|_2 < C$

Goal: Algorithms with sub-linear regret

Online convex optimization

Many algorithms achieve sub-linear regret

Online gradient descent (OGD) [Z03]

$x^{t+1} = \mathrm{Proj}(x^t - \eta_t \nabla c^t(x^t))$

Theorem [Z03]

When $\eta_t = \Theta(1/\sqrt{t})$, online gradient descent has $O(\sqrt{T})$-regret for OCO.

Theorem [Z03,H06]

Any algorithm for OCO must incur $\Omega(\sqrt{T})$-regret on linear cost functions.

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$

$\min_{x^t \in F} \sum_t c^t(x^t)$

$\|\nabla c^t(\cdot)\|_2 < C$

Goal: Algorithms with sub-linear regret

Online convex optimization

Many algorithms achieve sub-linear regret

- Online gradient descent (OGD) [Z03]
- Multiplicative weights [AHK05] [FS99]
- Newton's method based [HKKA06] [HAK07]

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$

$$\min_{x^t \in F} \sum_t c^t(x^t)$$

$$\|\nabla c^t(\cdot)\|_2 < C$$

Goal: Algorithms with sub-linear regret

Online convex optimization

Do OCO algorithms maintain sub-linear regret for SOCO?

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

$$\|\nabla c^t(\cdot)\|_2 < C$$

Goal: Algorithms with sub-linear regret

Smoothed online convex optimization (SOCO)

Do OCO algorithms maintain sub-linear regret for SOCO?

Corollary:

When $\eta_t = \Theta(1/\sqrt{t})$, online gradient descent still has $O(\sqrt{T})$-regret for SOCO.

Proof:

$$\sum_t \|x^t - x^{t-1}\|$$

$$\leq M \sum \|x^t - x^{t-1}\|_2 \qquad \text{(Equivalence of norms)}$$

$$\leq M \sum \eta_t \|\nabla c^{t-1}(x^{t-1})\|_2 \qquad \text{(Projection is non-expansive)}$$

$$\leq MC \sum \eta_t = O(\sqrt{T}) \qquad \text{(Bounded gradient)}$$

Can an algorithm maintain sub-linear regret?
Yes: Online gradient descent has $O(\sqrt{T})$-regret.

Can an algorithm maintain a constant competitive ratio? [BKRS92] [BLS92]…[BB00]…[LWAT11] [LRAMW12]

Can an algorithm maintain sub-linear regret and a constant competitive ratio?

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Goal: Algorithms with constant comp. ratio

Smoothed online convex optimization (SOCO)

2 Communities

- Online Learning
- Online Algorithms

2 Metrics

- Regret
- Competitive ratio

$c^1, x^1, c^2, x^2, c^3, x^3, \ldots$ online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \|x^t - x^{t-1}\|$$

convex — smoothed

Goal: Algorithms with constant comp. ratio

Metrical task system (MTS)

2 Communities

- Online Learning
- Online Algorithms

2 Metrics

- Regret
- Competitive ratio

$c^1, x^1, c^2, x^2, c^3, x^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

~~convex~~

Goal: Algorithms with constant comp. ratio

Metrical task system (MTS)

Is it possible to achieve a constant competitive ratio?

n is the size of the action set

Theorem [BLS92] [BKRS92]:

Any deterministic algorithm is $\Omega(n)$-competitive.

Any randomized algorithm is $\Omega\left(\sqrt{\log n / \log \log n}\right)$-competitive

$c^1, x^1, c^2, x^2, c^3, x^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

Goal: Algorithms with constant comp. ratio

Metrical task system (MTS)

Is it possible to achieve a constant competitive ratio?

$c^1, x^1, c^2, x^2, c^3, x^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \|x^t - x^{t-1}\|$$

convex scalar smoothed

Goal: Algorithms with constant comp. ratio

Smoothed online convex optimization (SOCO)

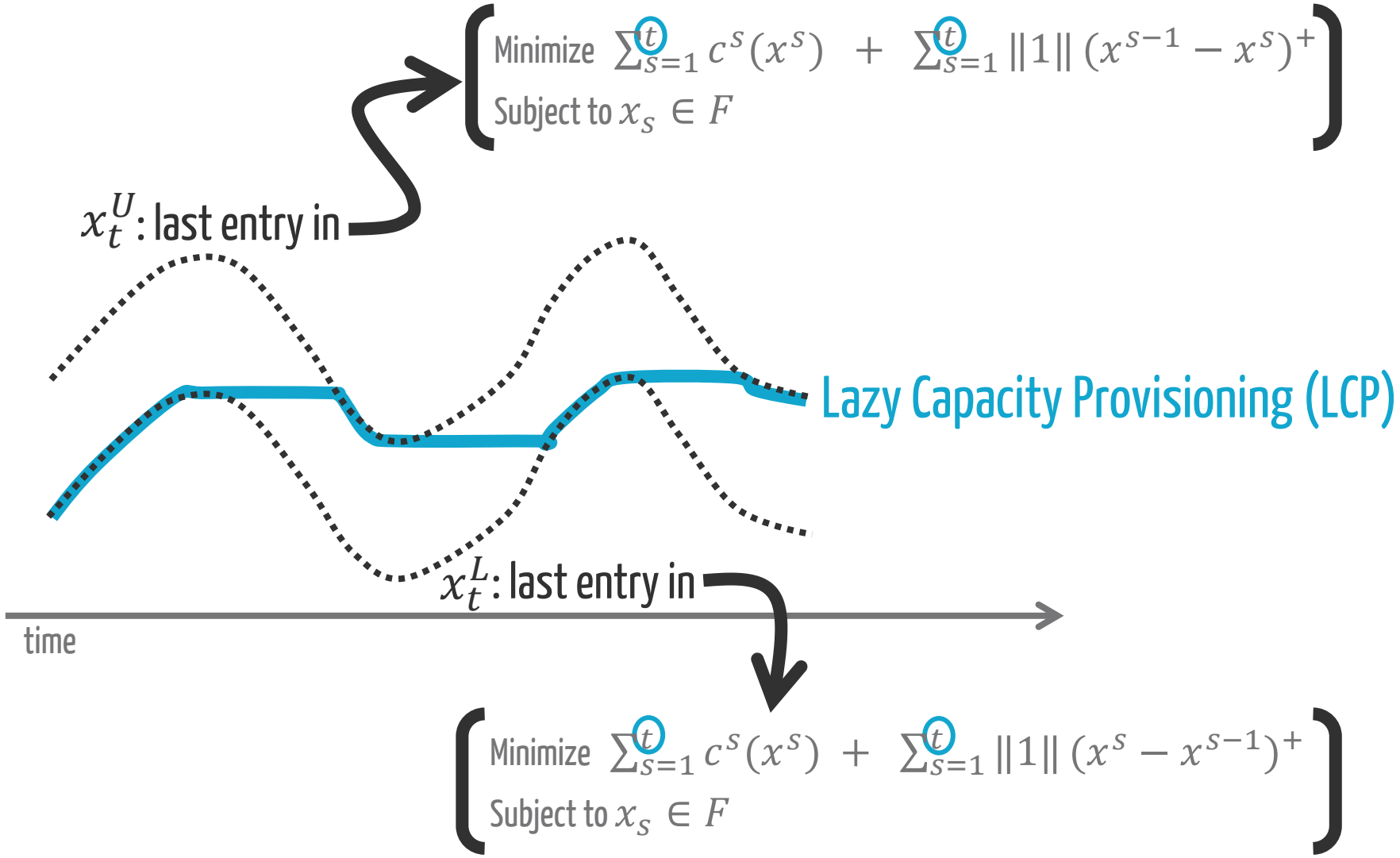
Is it possible to achieve a constant competitive ratio?

Theorem:

"Lazy Capacity Provisioning" (LCP) is 3-competitive.

Further $Cost(LCP) \leq Cost(OPT) + 2\, SwitchingCost(OPT)$

Is it possible to achieve a constant competitive ratio?


Minimize $\sum_{s=1}^{t} c^s(x^s) + \sum_{s=1}^{t} \|1\| (x^{s-1} - x^s)^+$
Subject to $x_s \in F$
x_t^U: last entry in
Lazy Capacity Provisioning (LCP)
x_t^L: last entry in
time
Minimize $\sum_{s=1}^{t} c^s(x^s) + \sum_{s=1}^{t} \|1\| (x^s - x^{s-1})^+$
Subject to $x_s \in F$

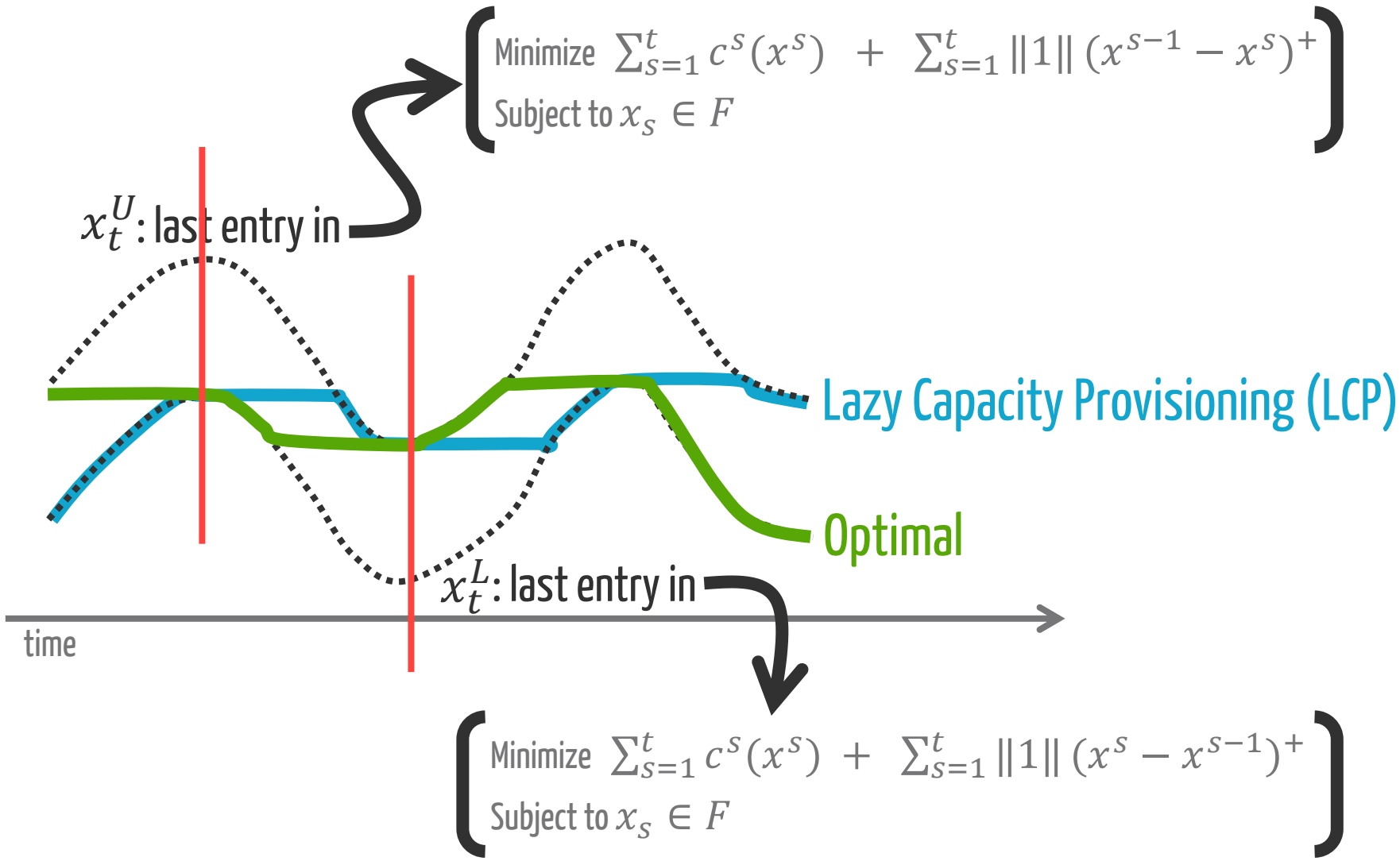
Theorem:

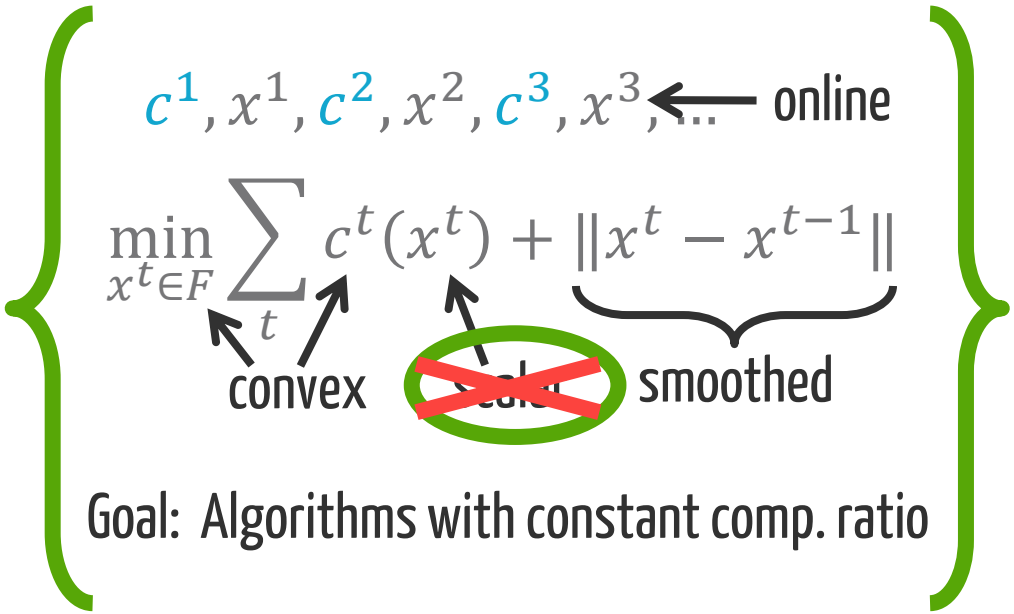
"Lazy Capacity Provisioning" (LCP) is 3-competitive.

Further $Cost(LCP) \leq Cost(OPT) + 2\, SwitchingCost(OPT)$

Lemma:

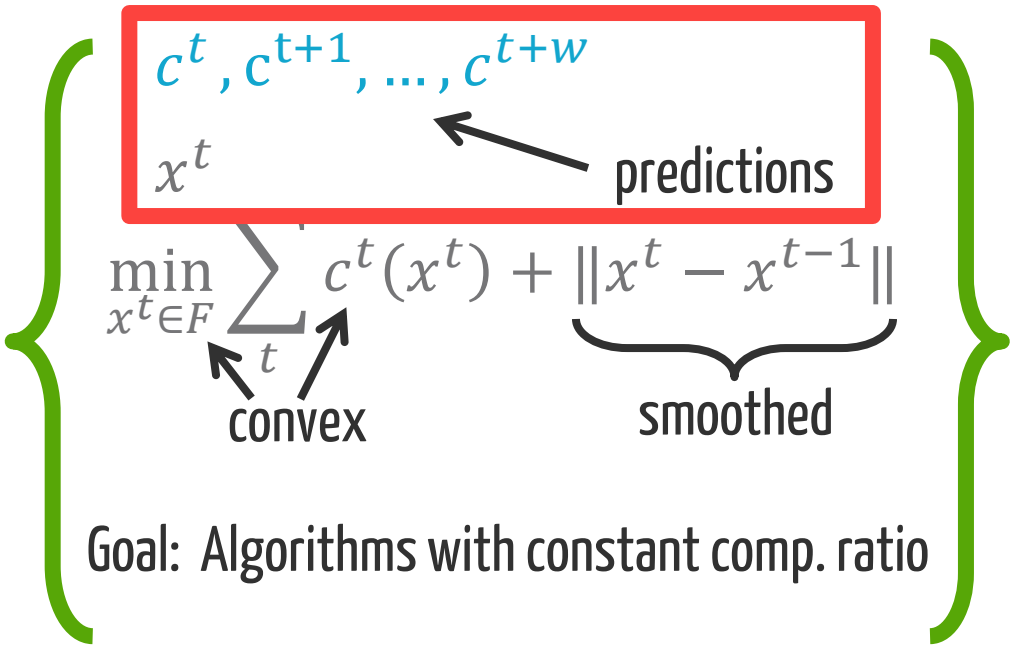
The offline optimal solution is "lazy in reverse time".


Minimize $\sum_{s=1}^{t} c^s(x^s) + \sum_{s=1}^{t} \|1\| (x^{s-1} - x^s)^+$
Subject to $x_s \in F$
x_t^U: last entry in
Lazy Capacity Provisioning (LCP)
Optimal
x_t^L: last entry in
time
Minimize $\sum_{s=1}^{t} c^s(x^s) + \sum_{s=1}^{t} \|1\| (x^s - x^{s-1})^+$
Subject to $x_s \in F$



Smoothed online convex optimization (SOCO)

Is it possible to achieve a constant competitive ratio?



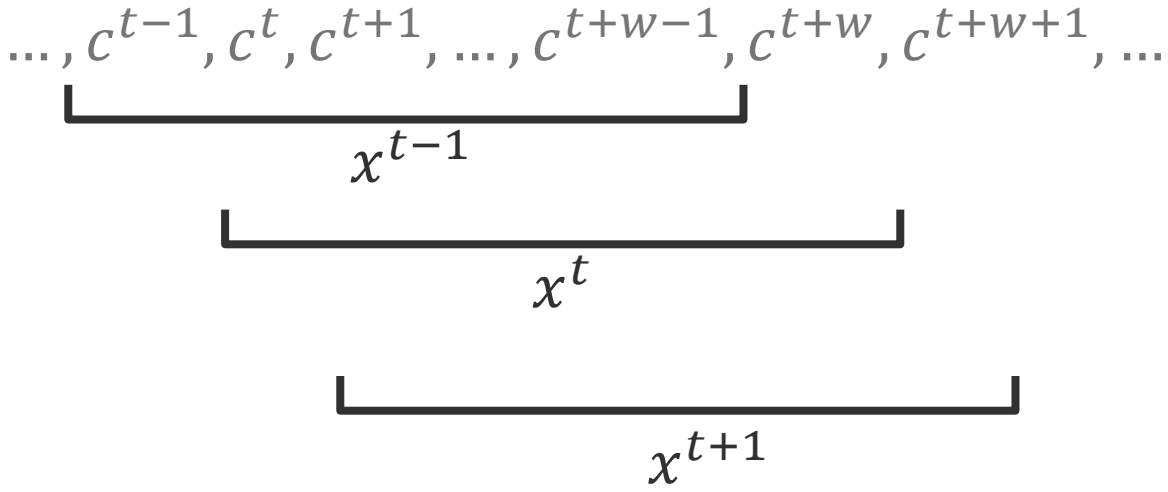
Smoothed online convex optimization (SOCO)

Is it possible to achieve a constant competitive ratio?

Classic Approach: Receding Horizon Control

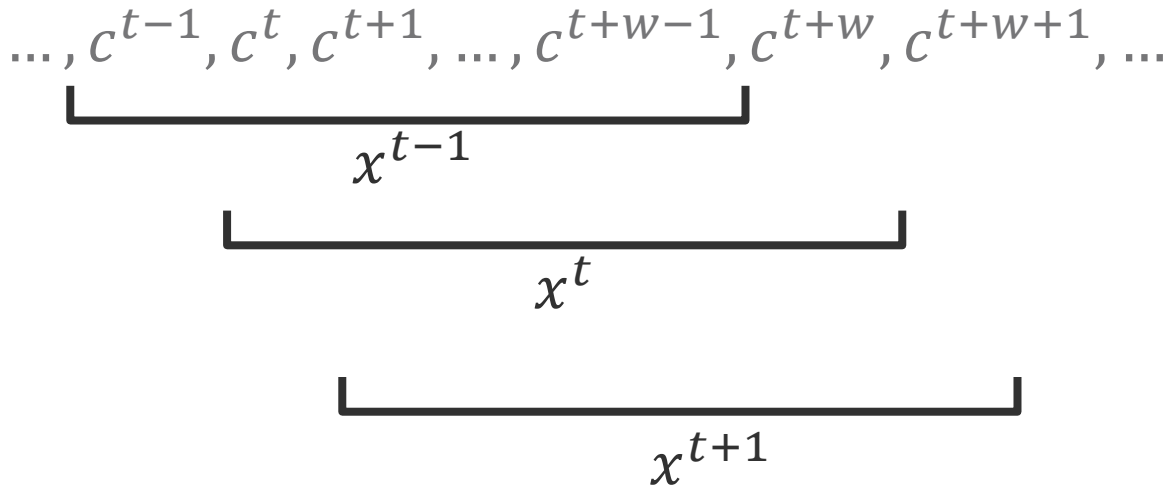
Is it possible to achieve a constant competitive ratio?

Classic Approach: Receding Horizon Control



Receding Horizon Control (RHC):

Choose x_t to minimize cost over $[t, t+w]$, given prediction window and x_{t-1}.



Receding Horizon Control (RHC):

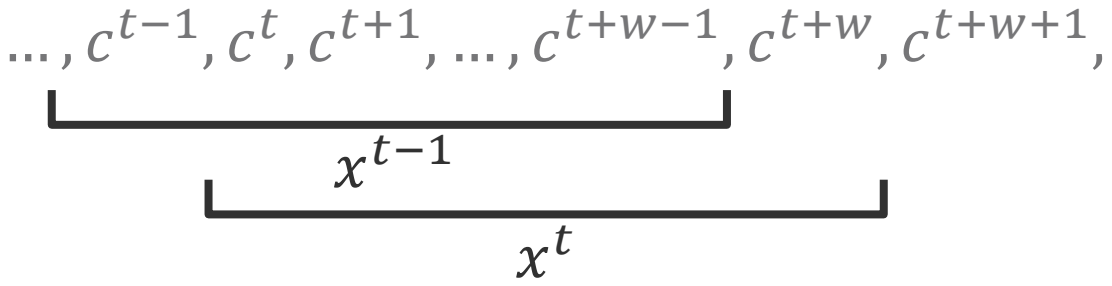
Choose x_t to minimize cost over $[t, t+w]$, given prediction window and x_{t-1}.

Theorem:

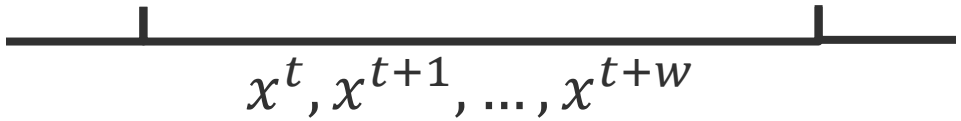
For one-dimensional SOCO, RHC is $O(1 + \|1\|/w)$-competitive . But, in general, RHC is $\Omega(\|1\|)$-competitive.

Does not improve as w grows!

RHC


$\ldots, c^{t-1}, c^t, c^{t+1}, \ldots, c^{t+w-1}, c^{t+w}, c^{t+w+1},$
x^{t-1}
x^t

FHC


$x^t, x^{t+1}, \ldots, x^{t+w}$

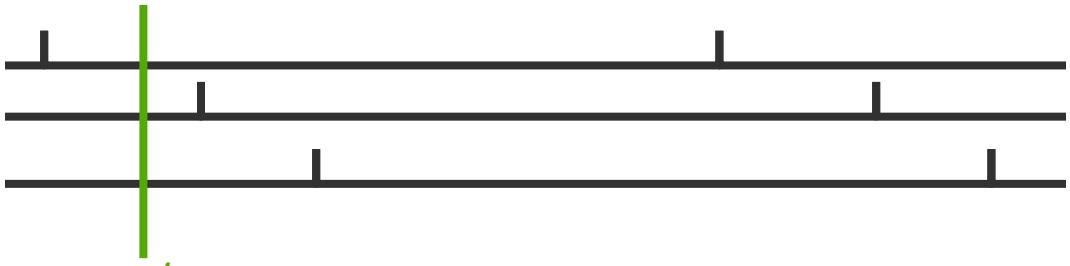
AFHC


x^t is the average

Averaging Fixed Horizon Control (AFHC):

Choose x_t as the average of $w + 1$ fixed horizon control algorithms.

AFHC



x^t is the average

Averaging Fixed Horizon Control (AFHC):

Choose x_t as the average of $w + 1$ fixed horizon control algorithms.

Theorem:

AFHC is $O(1 + ||1||/w)$- competitive

1 Can an algorithm maintain sub-linear regret?
Yes: Online gradient descent has $\Theta(\sqrt{T})$-regret.

2 Can an algorithm maintain a constant competitive ratio? Yes. (in the scalar case): LCP is 3-competitive.
Yes. (in the vector case): AFHC is $O(||1||/w)$-comp

3 Can an algorithm maintain sub-linear regret and a constant competitive ratio? **No!**

Deterministic or randomized

Theorem:

For arbitrary $\gamma > 0$ and any online algorithm A, there exist linear cost functions such that:

$$\mathrm{CR}(A) + \frac{Regret(A)}{T} \geq \gamma$$

What now?

Decide which metric you really care about?

- Is ϵT-regret "good enough"?
- Is $\log \log T$ - competitive "good enough"?
- What about under stochastic assumptions?

Decide which metric you really care about?

- Is ϵT-regret "good enough"?
- Is $\log \log T$ - competitive "good enough"?
- What about under stochastic assumptions?

Theorem:

Given a 1-dimensional SOCO, for arbitrary, increasing $f(T)$, "Randomly Biased Greedy" (RBG) is:

$(1 + f(T)/||1||)$-competitive with

$O(\max\{T/f(T), f(T)\})$-regret

Randomly Biased Greedy ($f(T)$)

$$x^t = \operatorname{argmin} w^t(x^t) + x^t r$$

Where: $\|1\|_{RBG} = f(T)$.

$$w^t(x) = \min_y \{w^{t-1}(y) + c^t(y) + \|x - y\|_{RBG}\}$$

$$r = \text{UnifRand}(-\|1\|_{RBG}, \|1\|_{RBG})$$

Theorem:

Given a 1-dimensional SOCO, for arbitrary, increasing $f(T)$,

"Randomly Biased Greedy" (RBG) is:

$(1 + f(T)/\|1\|)$-competitive with

$O(\max\{T/f(T), f(T)\})$-regret

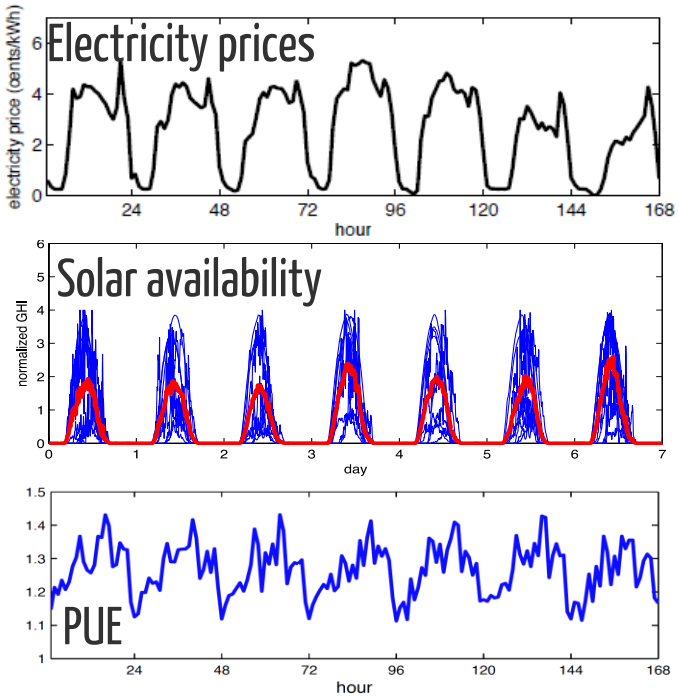
1 Can an algorithm maintain sub-linear regret?
Yes: Online gradient descent has $\Theta\left(\sqrt{T}\right)$-regret.

2 Can an algorithm maintain a constant competitive ratio? Yes. (in the scalar case): LCP is 3-competitive.
Yes. (in the vector case): AFHC is $O(||1||/w)$-comp

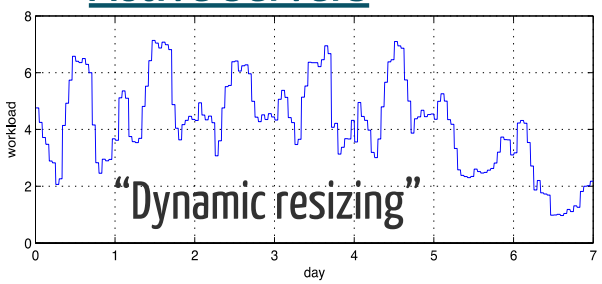
3 Can an algorithm maintain sub-linear regret and a constant competitive ratio? No! But (in the scalar case): RBG is $O(f(T))$-competitive with $O(\max\{T/f(T), f(T)\})$-regret.

Back to the applications...

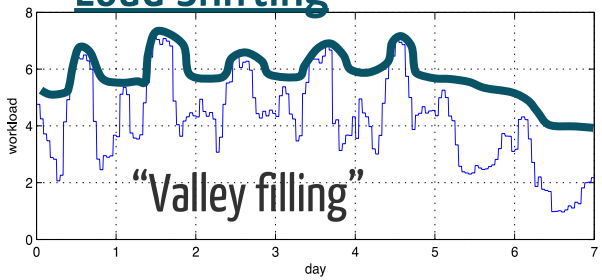
An implementation: Dynamic resizing



Active Servers



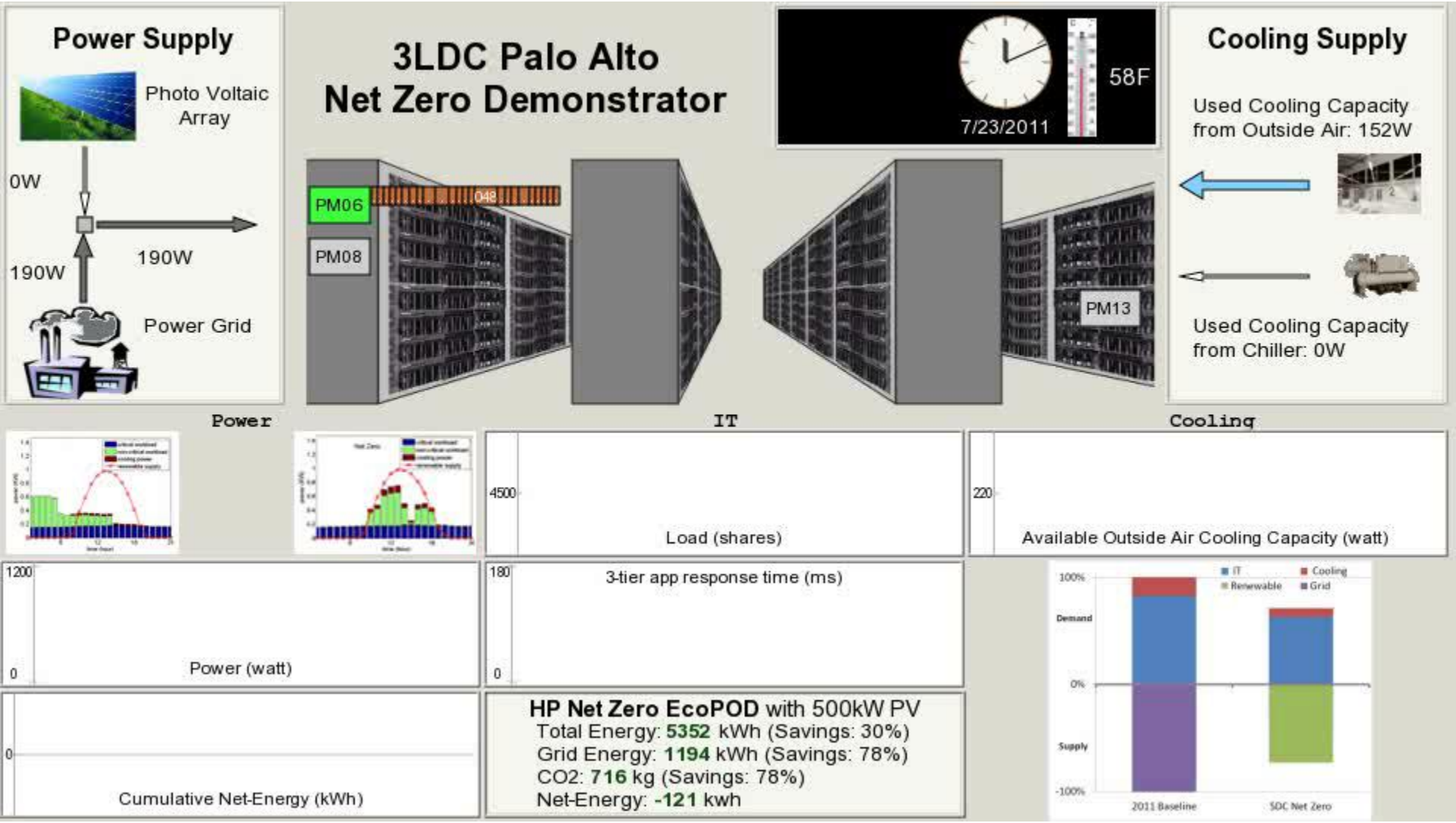
Load shifting



An implementation: Dynamic resizing

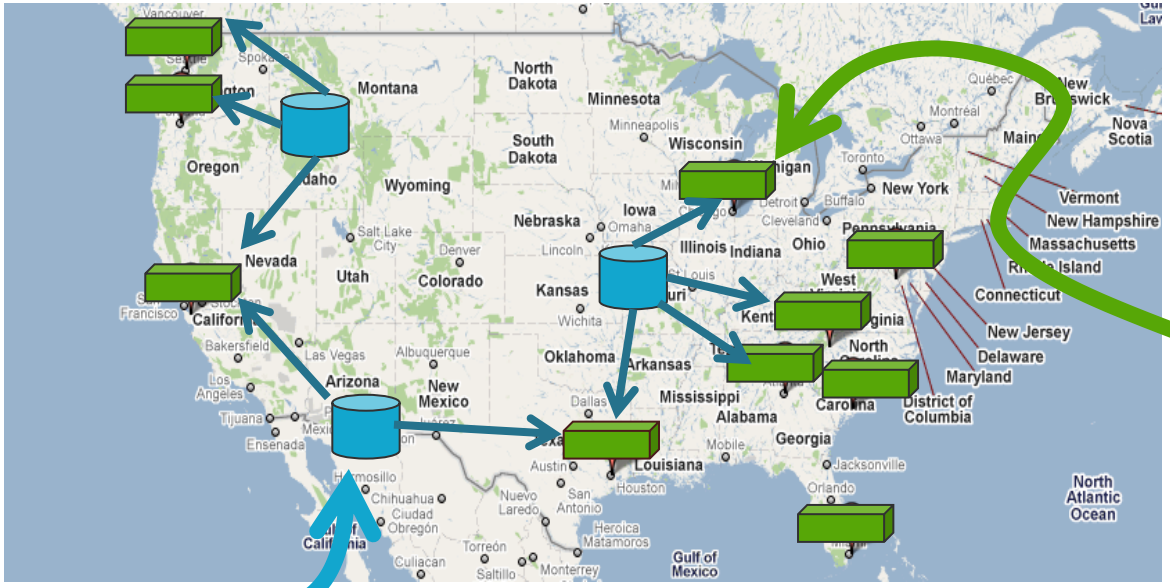
HP collaborators: Yuan Chen, Cullen Bash, Martin Arlitt, Daniel Gmach, Zhikui Wang, Manish Marwah and Chris Hyser

An implementation: Dynamic resizing



HP collaborators: Yuan Chen, Cullen Bash, Martin Arlitt, Daniel Gmach, Zhikui Wang, Manish Marwah and Chris Hyser

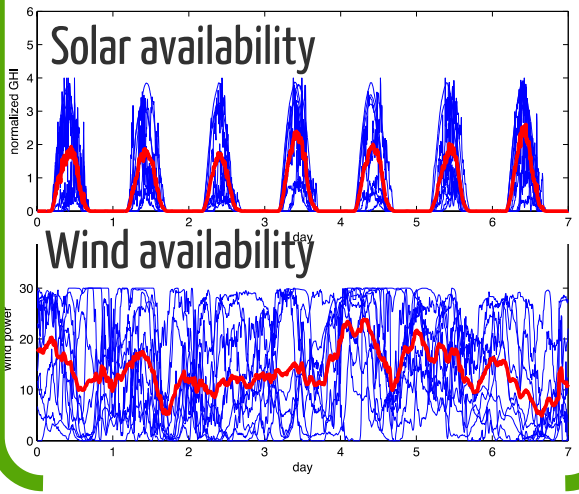
A case study: Geographical load balancing



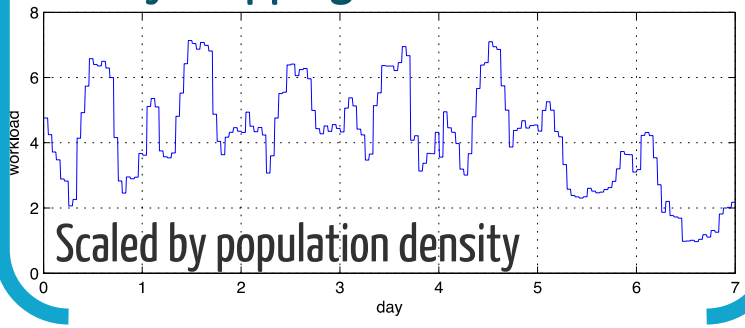
Data centers:

Google data center locations

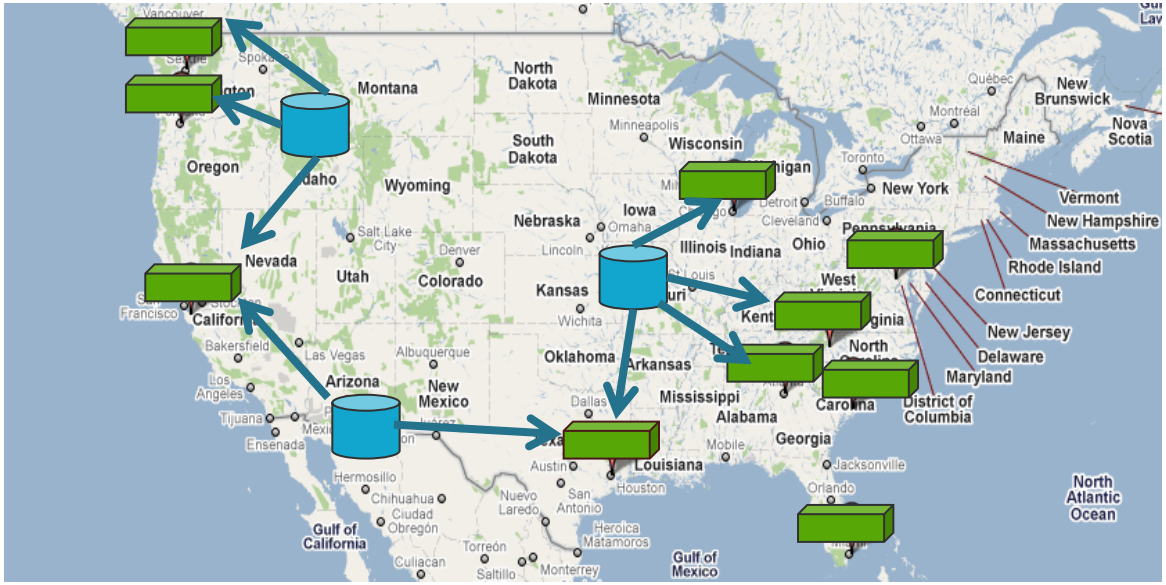
Real traces for wind, solar, electricity price, etc.



Proxy/Mapping Nodes:



A case study: Geographical load balancing



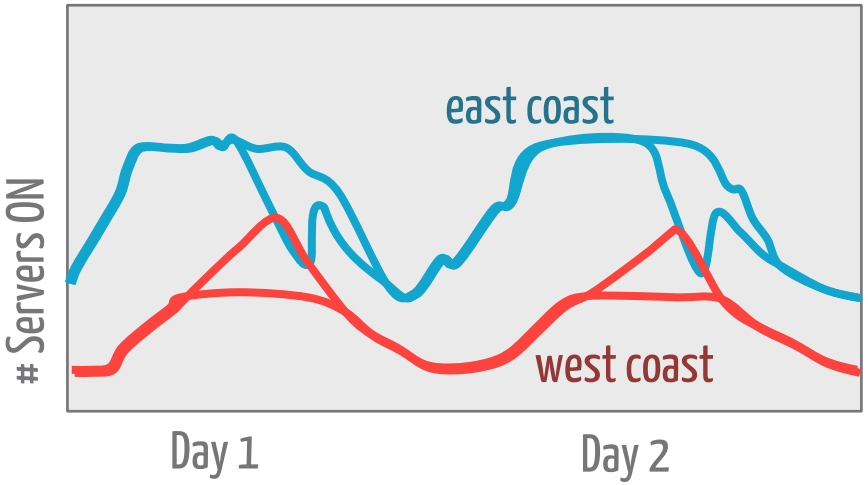
GLB: geographical load balancing & dynamic resizing

vs.

LOCAL: Route to the closest data center & dynamic resizing

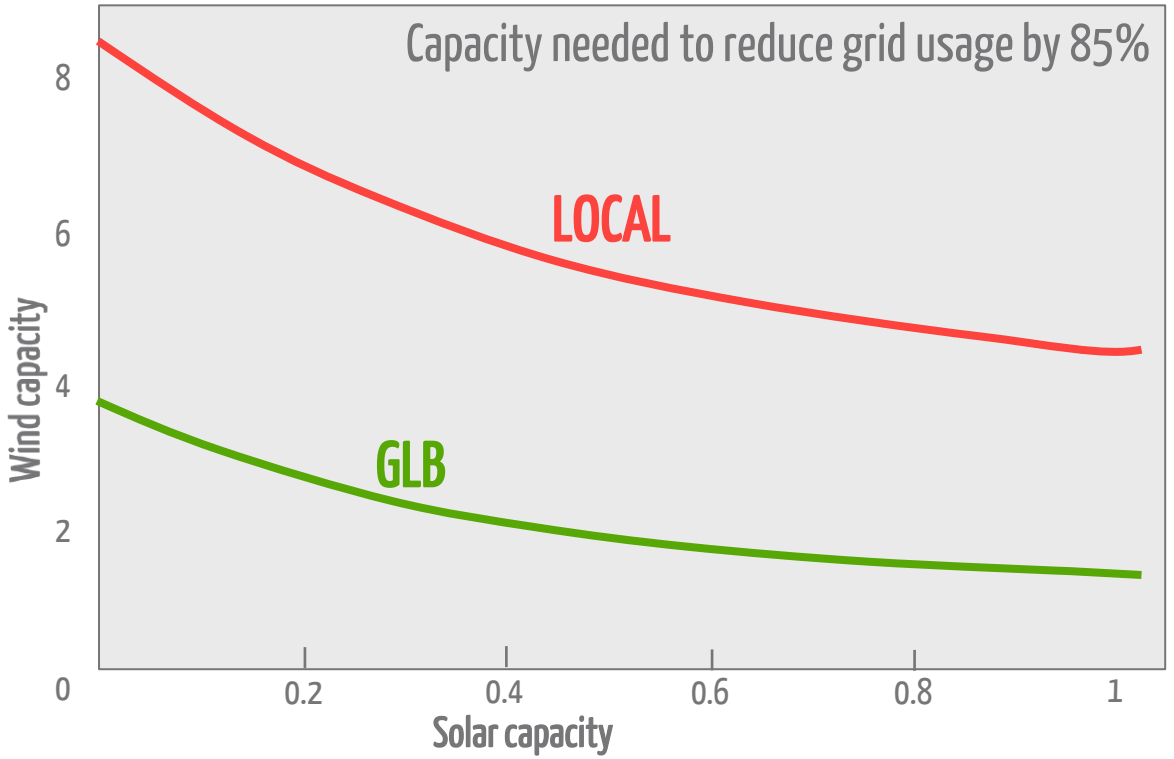
The good

Follow the renewables routing emerges.



The good

Follow the renewables routing emerges.
Huge reductions in grid usage become possible.

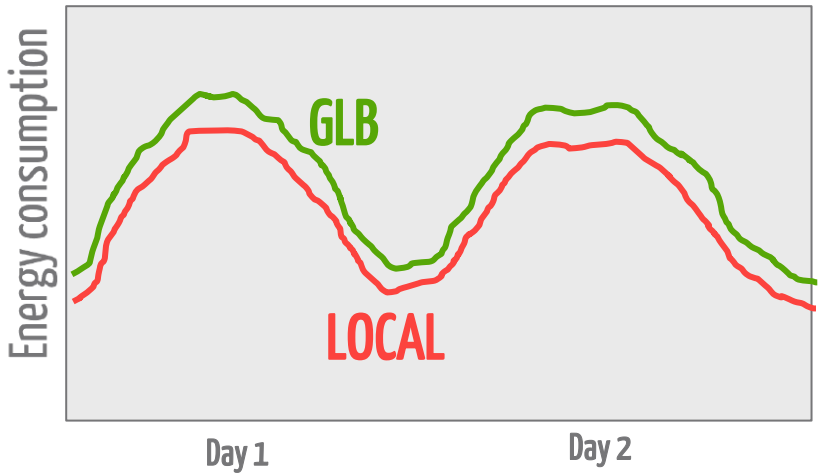


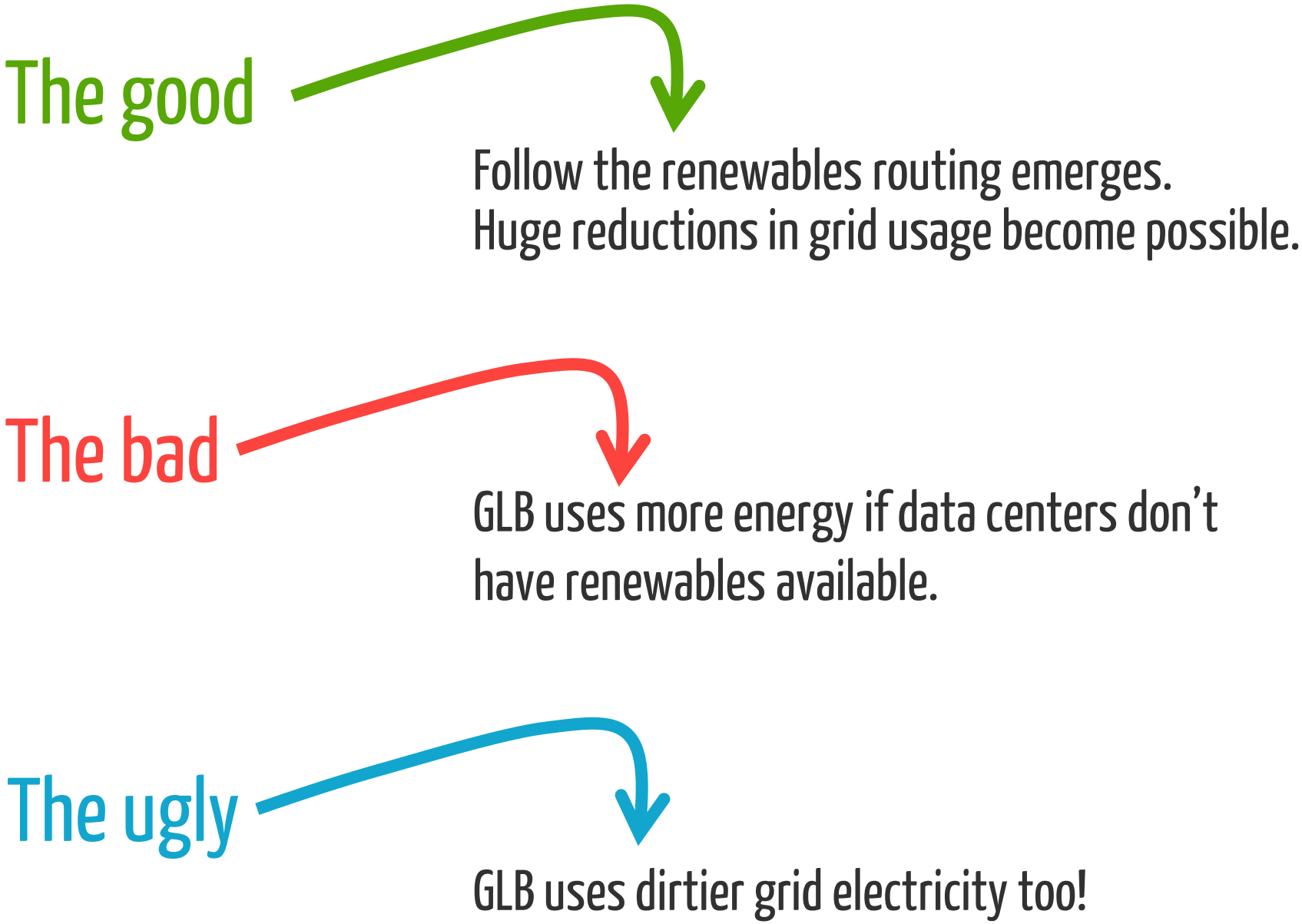
The good

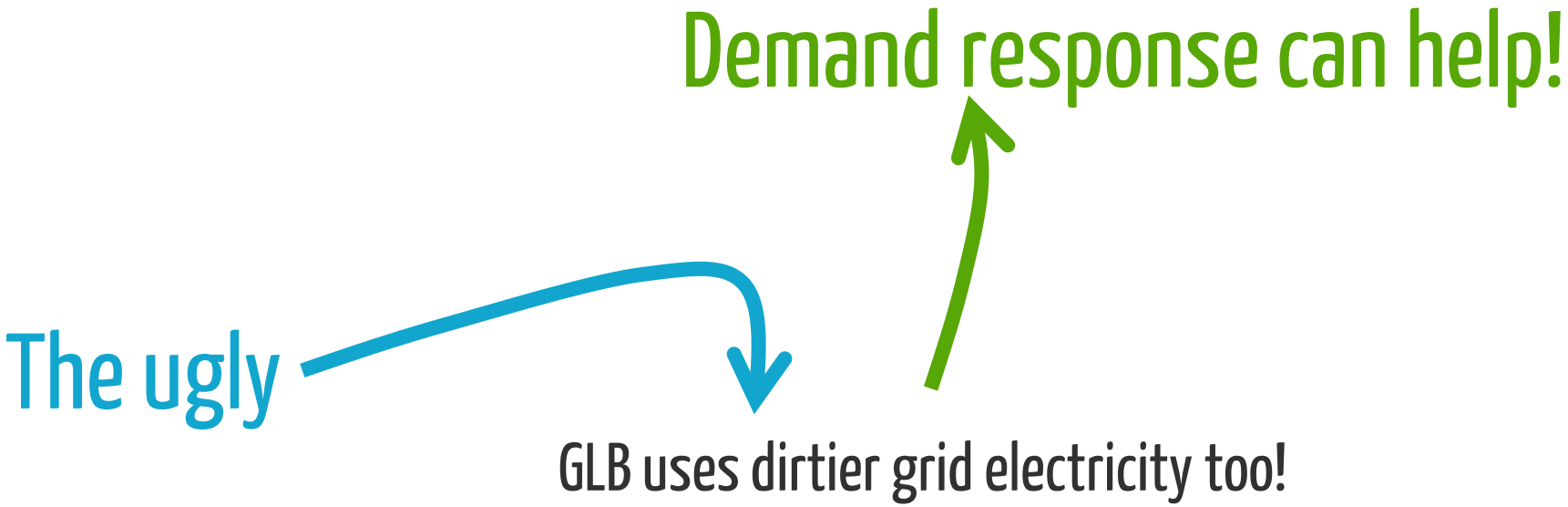
Follow the renewables routing emerges.
Huge reductions in grid usage become possible.

The bad

GLB uses more energy if data centers don't have renewables available.



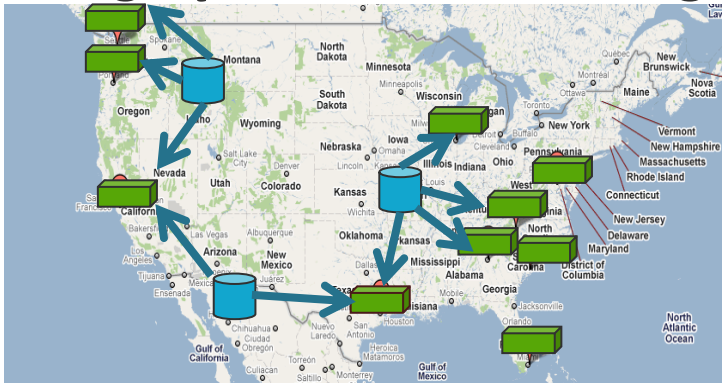

The good
Follow the renewables routing emerges.
Huge reductions in grid usage become possible.
The bad
GLB uses more energy if data centers don't
have renewables available.
The ugly
GLB uses dirtier grid electricity too!


Demand response can help!
The ugly
GLB uses dirtier grid electricity too!

Final thoughts

Dynamic resizing

Geographical load balancing



$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Smoothed online convex optimization

Implementation

Lots of interesting theoretical questions remain:

Stochastic models for cost functions → better algorithms?

Predictions are extremely important in practice → Stochastic variations?

The full cost function is not observed in practice → Bandit versions?

Can an algorithm balance regret and the competitive ratio?

$x^1, c^1, x^2, c^2, x^3, c^3, \ldots$ ← online

$$\min_{x^t \in F} \sum_t c^t(x^t) + \underbrace{\|x^t - x^{t-1}\|}_{\text{smoothed}}$$

convex

Smoothed online convex optimization

Algorithms for sustainable data centers

Adam Wierman (Caltech)
Minghong Lin (Caltech)
Zhenhua Liu (Caltech)
Lachlan Andrew (Swinburne)
and many others…

- Minghong Lin, Adam Wierman, Lachlan Andrew, and Eno Thereska. Dynamic right-sizing for power proportional data centers. Infocom, 2011. Best Paper award winner.
- Zhenhua Liu, Minghong Lin, Adam Wierman, Steven Low, and Lachlan Andrew. Greening geographical load balancing. Sigmetrics 2011.
- Zhenhua Liu, Minghong Lin, Adam Wierman, Steven Low, and Lachlan Andrew. Geographical load balancing with renewables. Greenmetrics, 2011. Best Student Paper award winner.
- Zhenhua Liu, Yuan Chen, Cullen Bash, Adam Wierman, et al. Renewable and cooling aware workload management for sustainable data centers. Sigmetrics 2012.
- Minghong Lin, Lachlan Andrew, and Adam Wierman. Online Algorithms for Geographical Load Balancing. Green Computing Conference, 2012. Best Paper award winner.