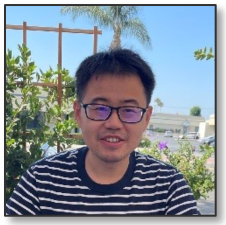
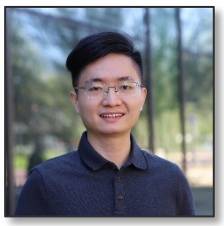
# Learning-Augmented Algorithms for Sustainable Systems

Adam Wierman, Caltech

Guannan Qu



Yuanyuan Shi



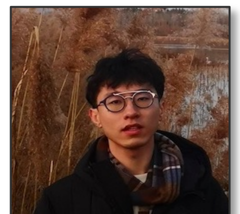
Guanya Shi

Yingying Li

Nico Christianson

Jing Yu

Tinashe Handina



Tongxin Li

Chris Yeh

Yiheng Lin

amazon

vmware

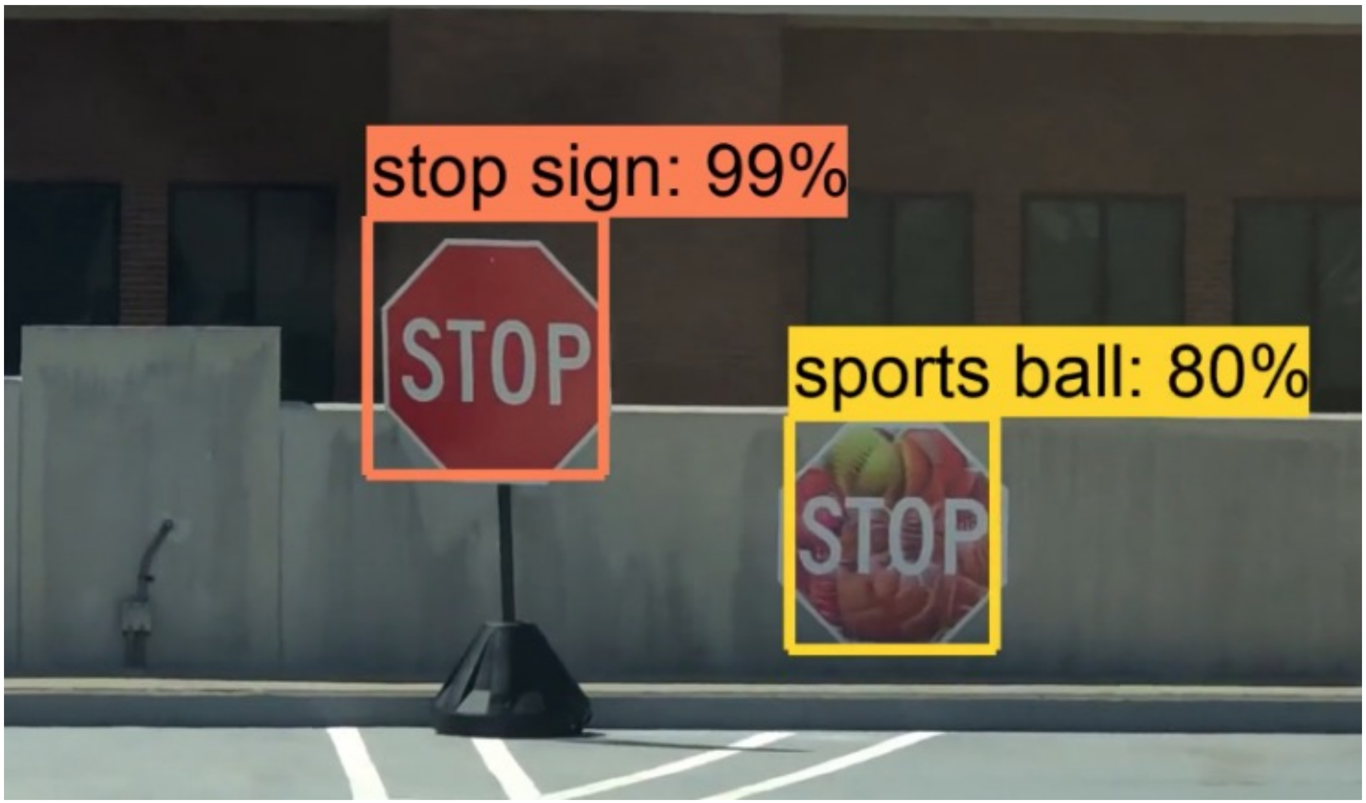
NSF

RESNICKINSTITUTE
science + energy + sustainability
CALIFORNIA INSTITUTE OF TECHNOLOGY

BEYOND LIMITS

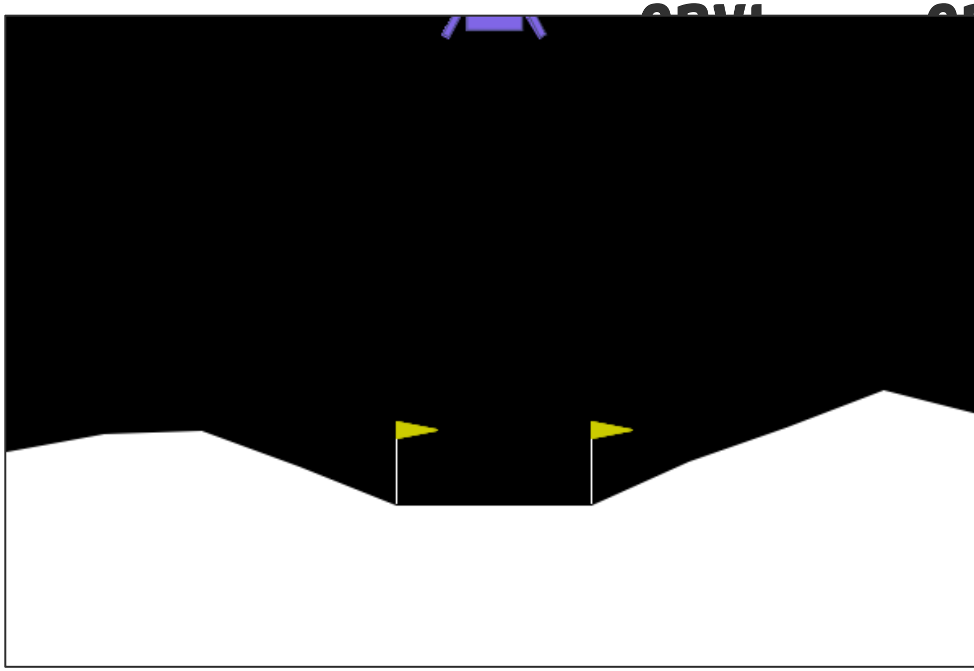

AI can potentially give us more resilient, sustainable,
and autonomous energy systems...
Are AI tools ready?

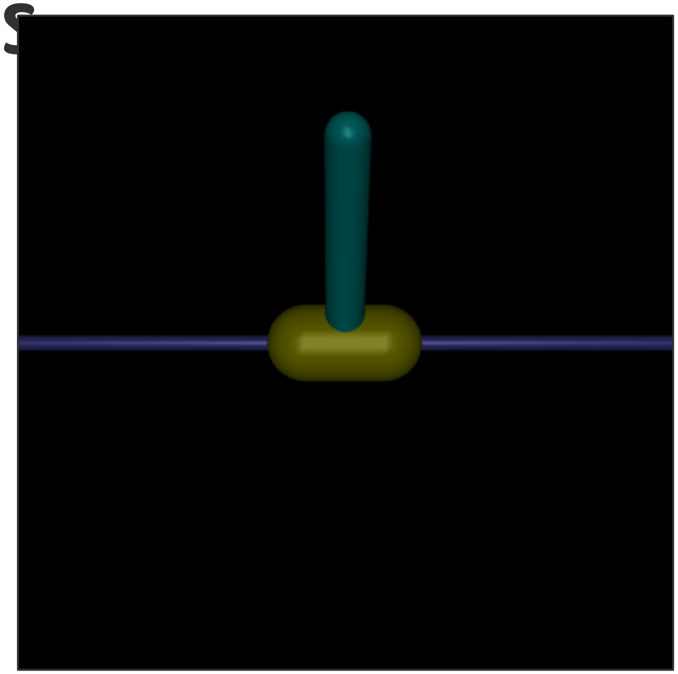
[Pei et al 2017]



[Pei et al 2017]

# Most algorithms are benchmarked on toy environments





# Most algorithms are benchmarked on toy environments

Energy systems must deal with

- physical constraints
- distribution shifts
- distributed, multi-agent control

# Introducing Caltech/UCSD SustainGym

Five environments (so far):

1. Adaptive EV charging (local and multi-location)
2. Grid-scale battery storage management for price arbitrage
3. Data center dynamic capacity management (VCCs, local and global)
4. Cogeneration management of a plant producing steam and electricity
5. Smart building management to meet temperature requirements

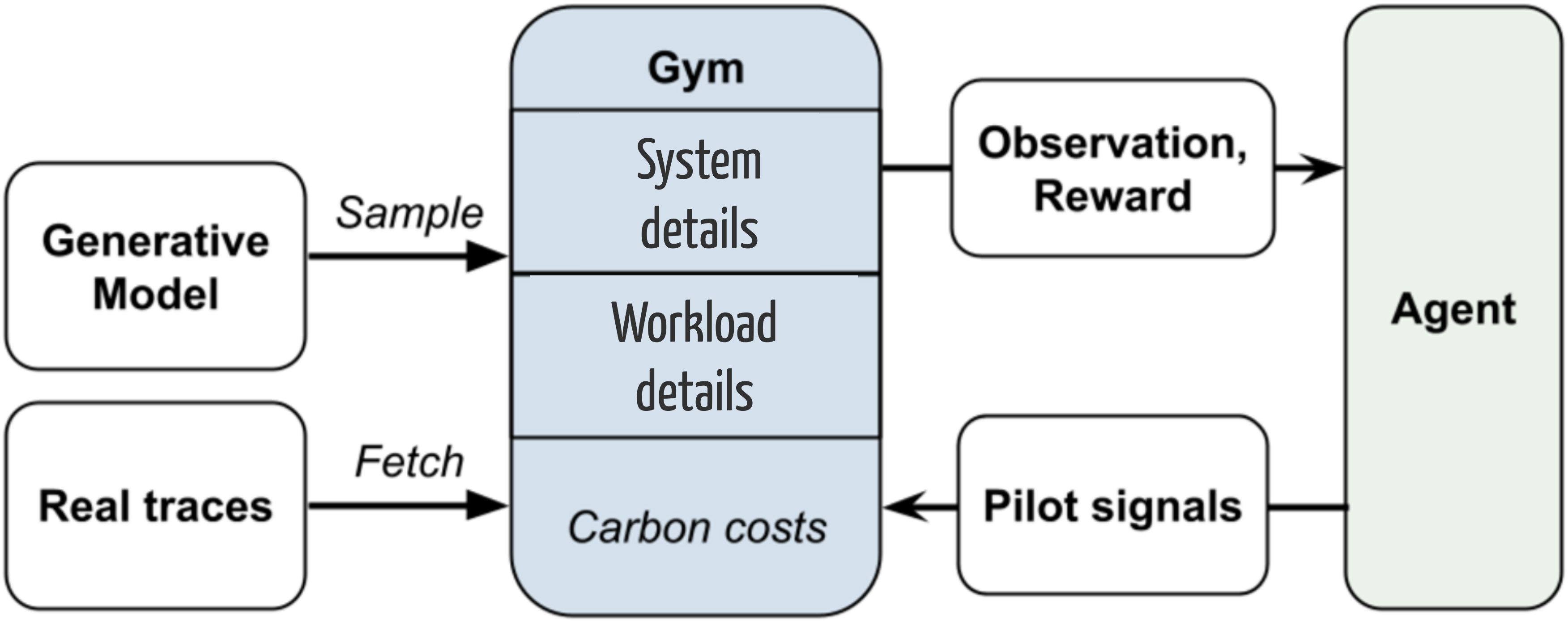
Caltech/UCSD Collaboration led by Christopher Yeh with co-authors:
Victor Li, Rajeev Datta, Julio Arroyo, Nicolas Christianson, Chi Zhang, Yize Chen, Mohammad Hosseini, Azarang Golmohammadi, Yuanyuan Shi, Yisong Yue

# Introducing Caltech/UCSD SustainGym

Environments feature

- Focus on marginal carbon emissions (uses other Umass work)
- Real-world data and distribution shifts (from Google/etc.)
- Distribution shifts in demand & environmental parameters
- Physical constraints
- Mix of discrete and continuous actions
- Multi-agent settings

# Introducing Caltech/UCSD SustainGym



# An example: Carbon-first Cloud Computing

HOME › NEWS › NORTH AMERICA

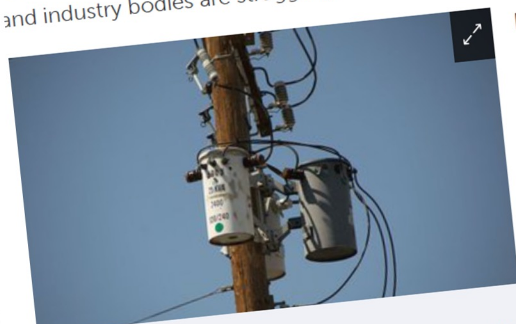
# Dominion Energy admits it can't meet data center power demands in Virginia

## The high-voltage lines simply can't handle more power, says the utility

July 29, 2022 By: Peter Judge Have your say

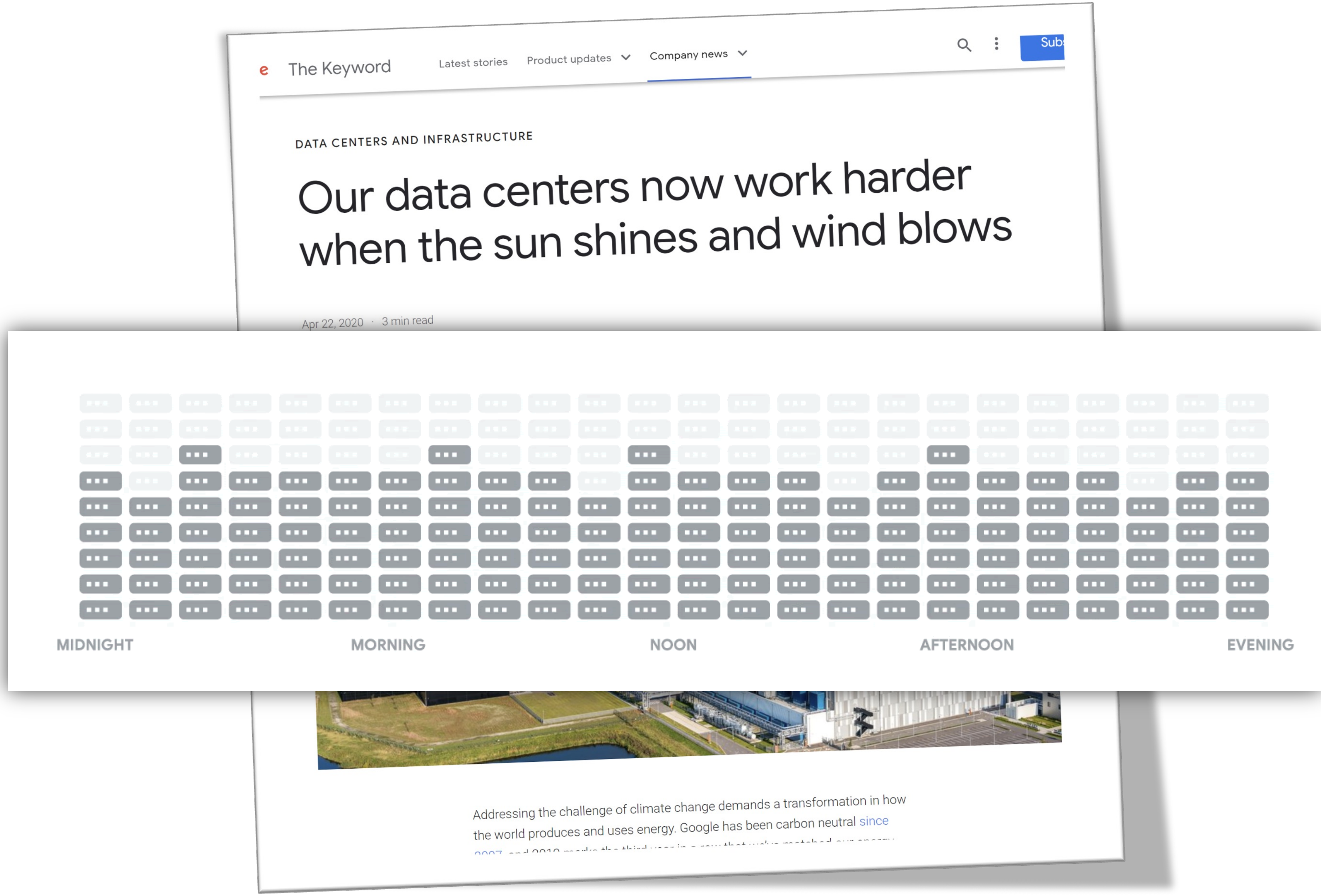
North American utility Dominion Energy says it may not be able to meet demands for power in Ashburn, Northern Virginia, delaying building projects in the world's fastest-growing data center hub by many years.

Dominion has told customers that it has power supplies, but can no longer guarantee to deliver the quantity of electricity customers want via overhead powerlines. If these warnings prove true, this could stall projects with billions invested, and Loudoun County's tax revenue would take a severe hit if the hub of data centers in Ashburn stalls. For now, local authorities and industry bodies are struggling to understand the sudden warning from Dominion.



Dominion supplies electricity in Virginia, North Carolina, and South Carolina, as well as natural gas to parts of the US. In the data center-rich counties of Loudoun, Prince William, and Fauquier, most of the electricity is carried by overhead powerlines marching along roads - a delivery method that has led to protests.

Loudoun County has 26 million square feet of data center space, with 5 million more in development and many more projects planned. Data center equipment taxes provide one-third of the County's tax income, but has



The Keyword

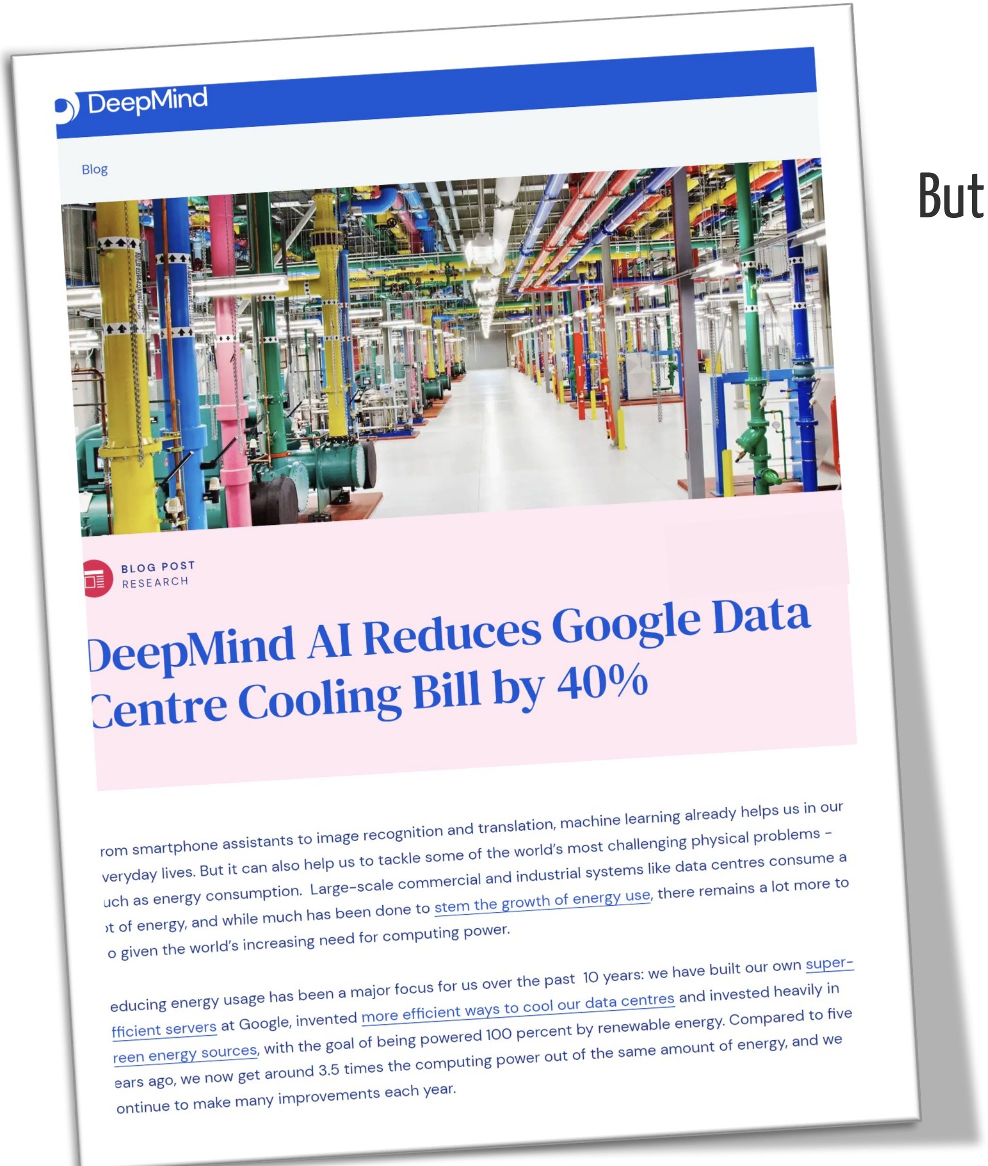
Latest stories | Product updates | Company news

DATA CENTERS AND INFRASTRUCTURE

# Our data centers now work harder when the sun shines and wind blows

Apr 22, 2020 · 3 min read

Addressing the challenge of climate change demands a transformation in how the world produces and uses energy. Google has been carbon neutral since


DeepMind

Blog

BLOG POST
RESEARCH

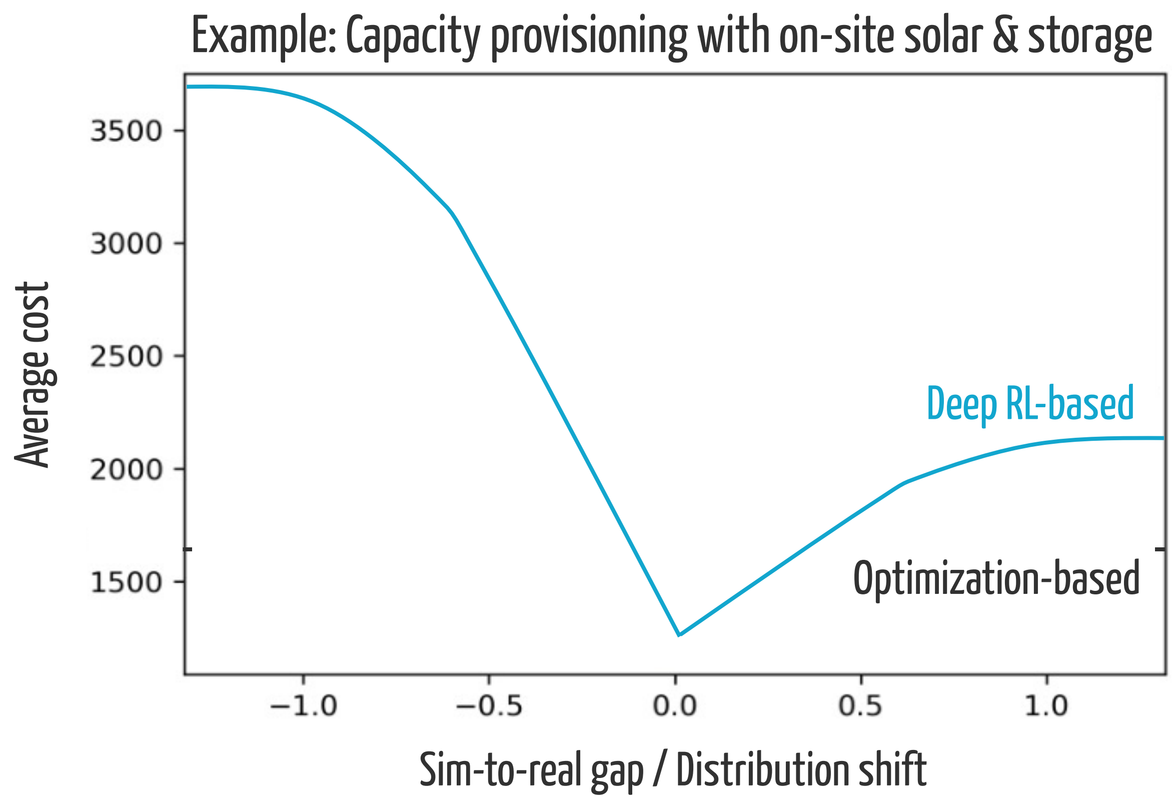
DeepMind AI Reduces Google Data Centre Cooling Bill by 40%

rom smartphone assistants to image recognition and translation, machine learning already helps us in our veryday lives. But it can also help us to tackle some of the world's most challenging physical problems – uch as energy consumption. Large-scale commercial and industrial systems like data centres consume a ot of energy, and while much has been done to stem the growth of energy use, there remains a lot more to o given the world's increasing need for computing power.

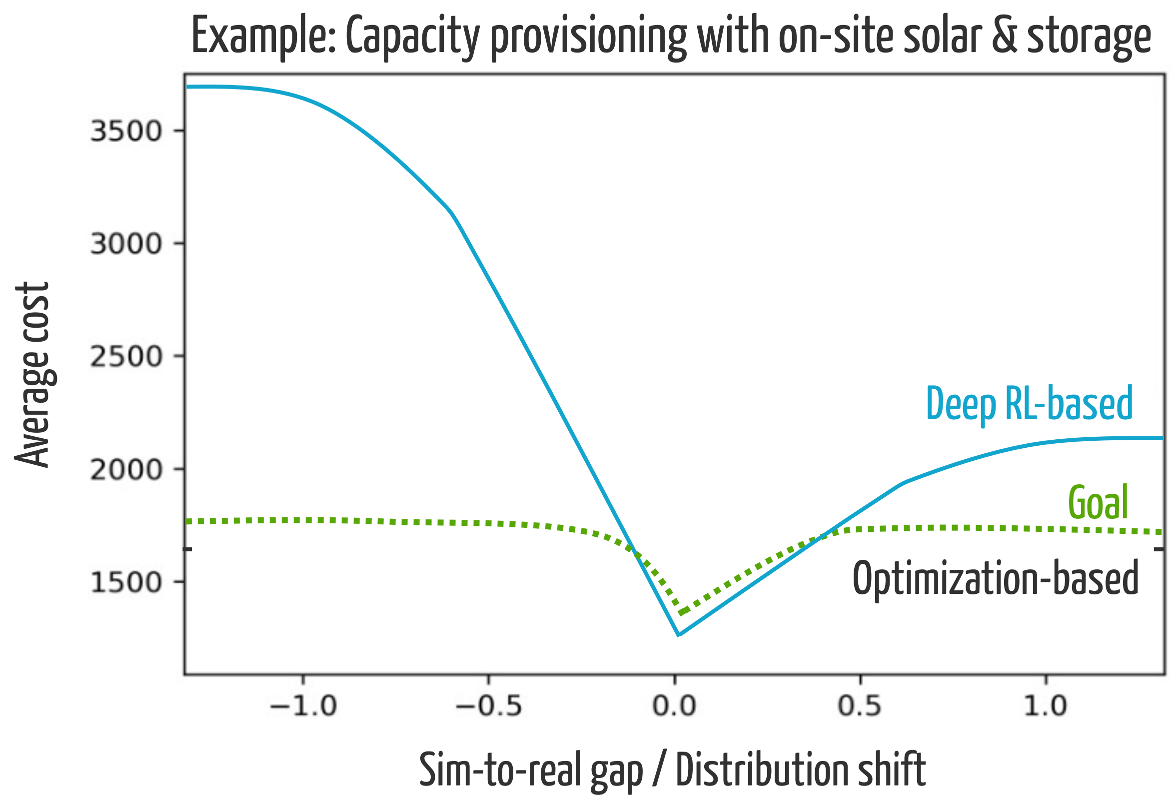
educing energy usage has been a major focus for us over the past 10 years: we have built our own super-fficient servers at Google, invented more efficient ways to cool our data centres and invested heavily in reen energy sources, with the goal of being powered 100 percent by renewable energy. Compared to five ears ago, we now get around 3.5 times the computing power out of the same amount of energy, and we ontinue to make many improvements each year.

But ML/AI tools are not in use in practice…

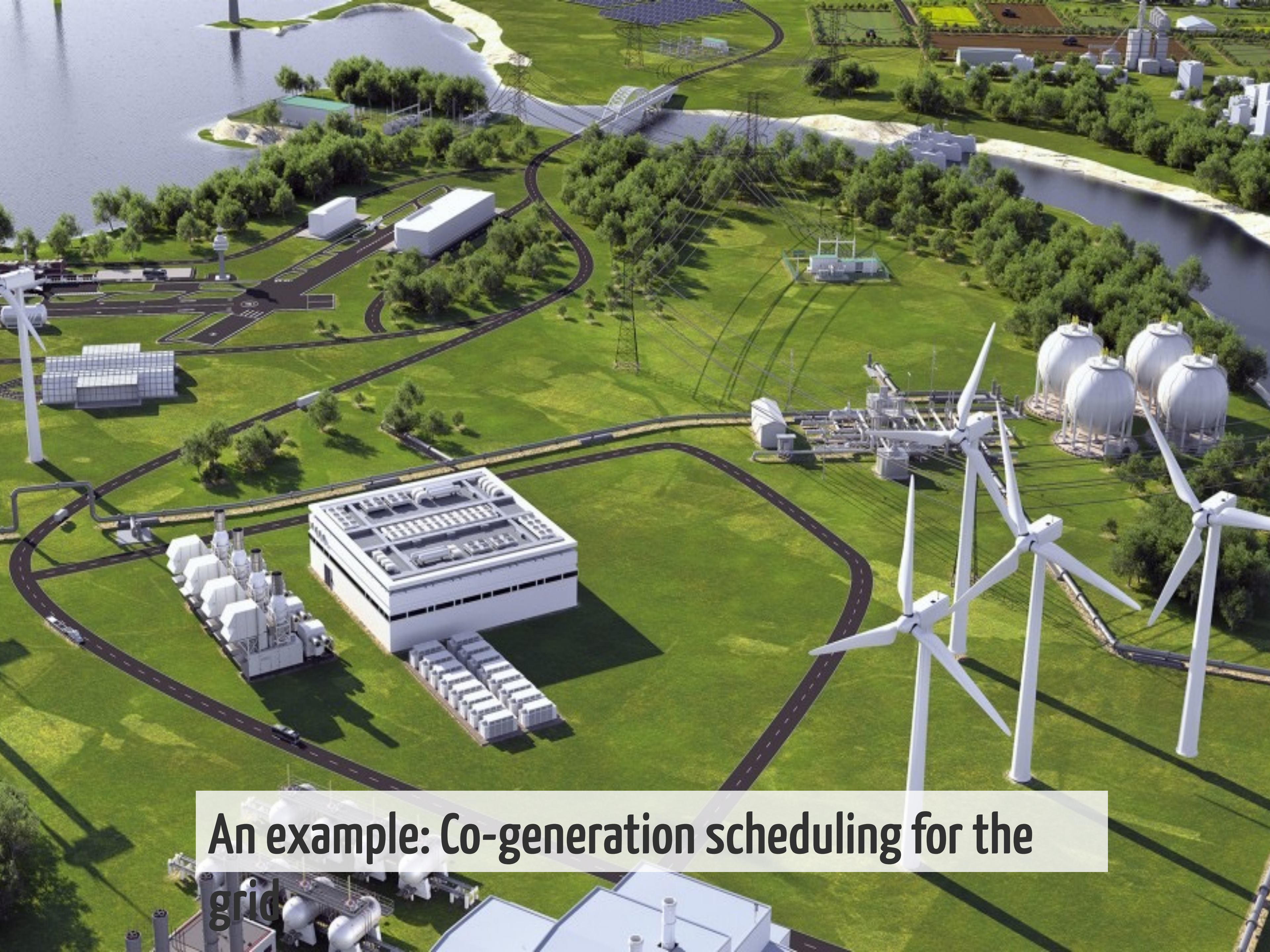
**<u>Can't afford to "fail at scale"</u>**



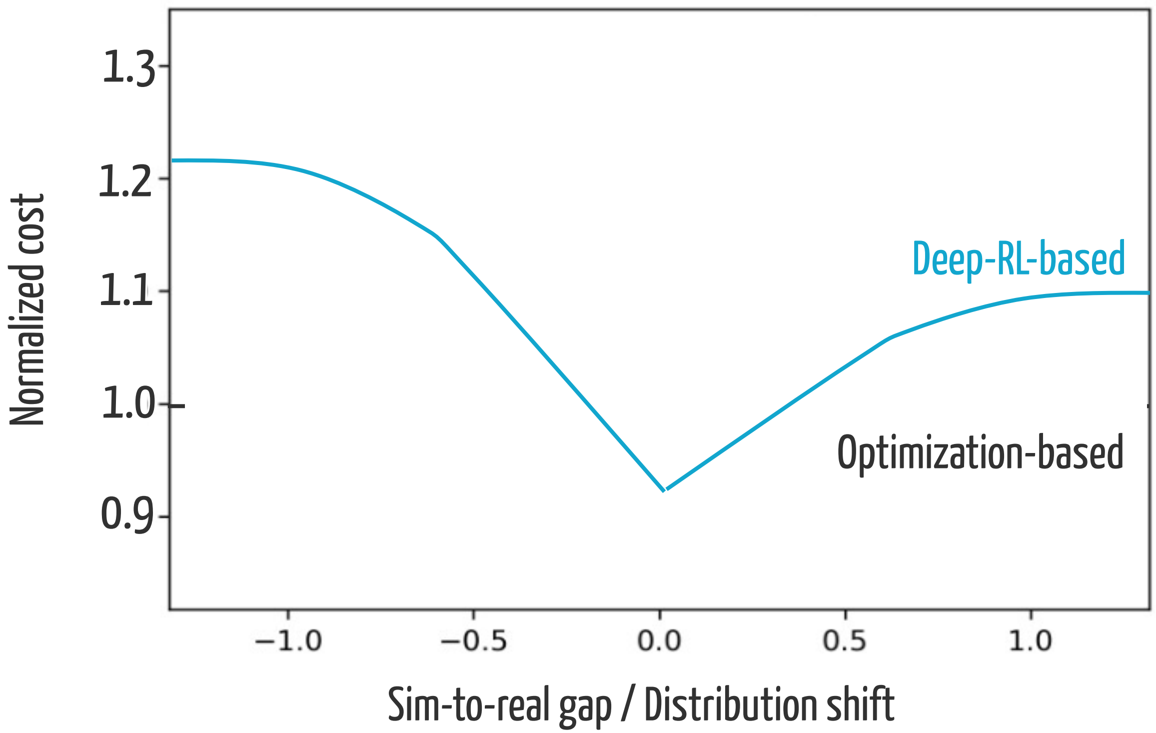
Example: Capacity provisioning with on-site solar & storage
Average cost
3500
3000
2500
2000
1500
−1.0
−0.5
0.0
0.5
1.0
Sim-to-real gap / Distribution shift
Deep RL-based
Optimization-based


Example: Capacity provisioning with on-site solar & storage
Average cost
3500
3000
2500
2000
1500
−1.0
−0.5
0.0
0.5
1.0
Sim-to-real gap / Distribution shift
Deep RL-based
Goal
Optimization-based

# An example: Co-generation scheduling for the grid

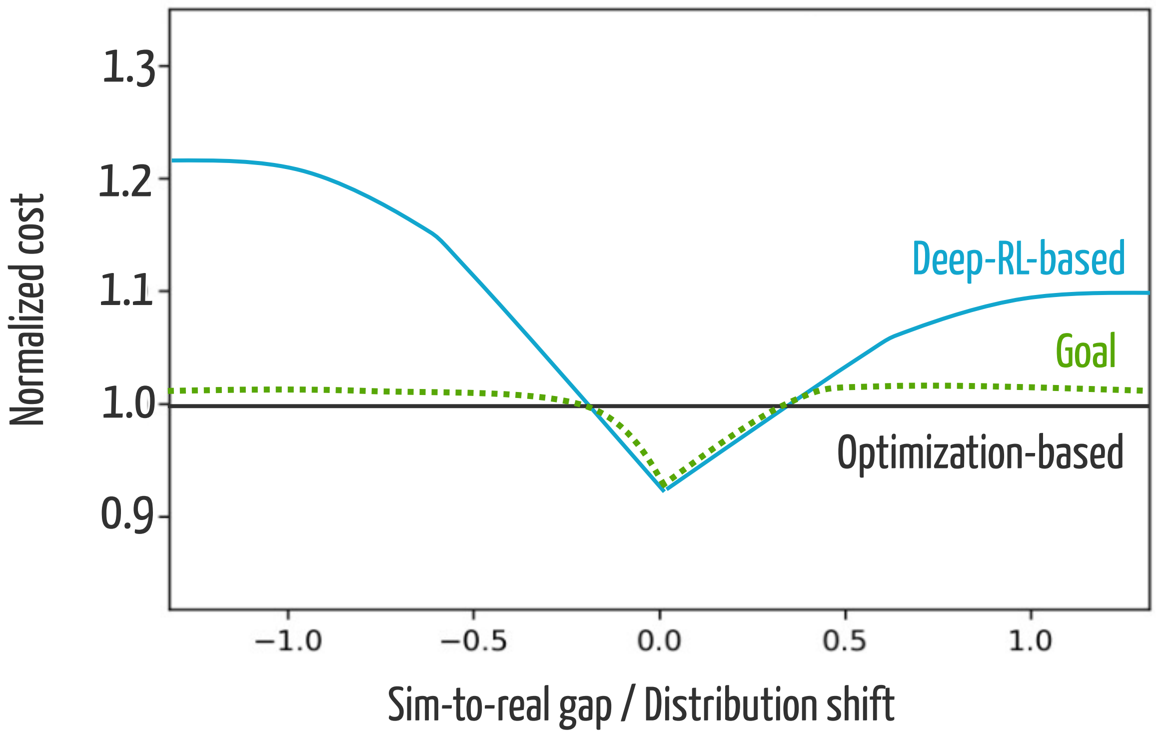


Example: Co-generation plant (steam+electricity) with co-located wind generation

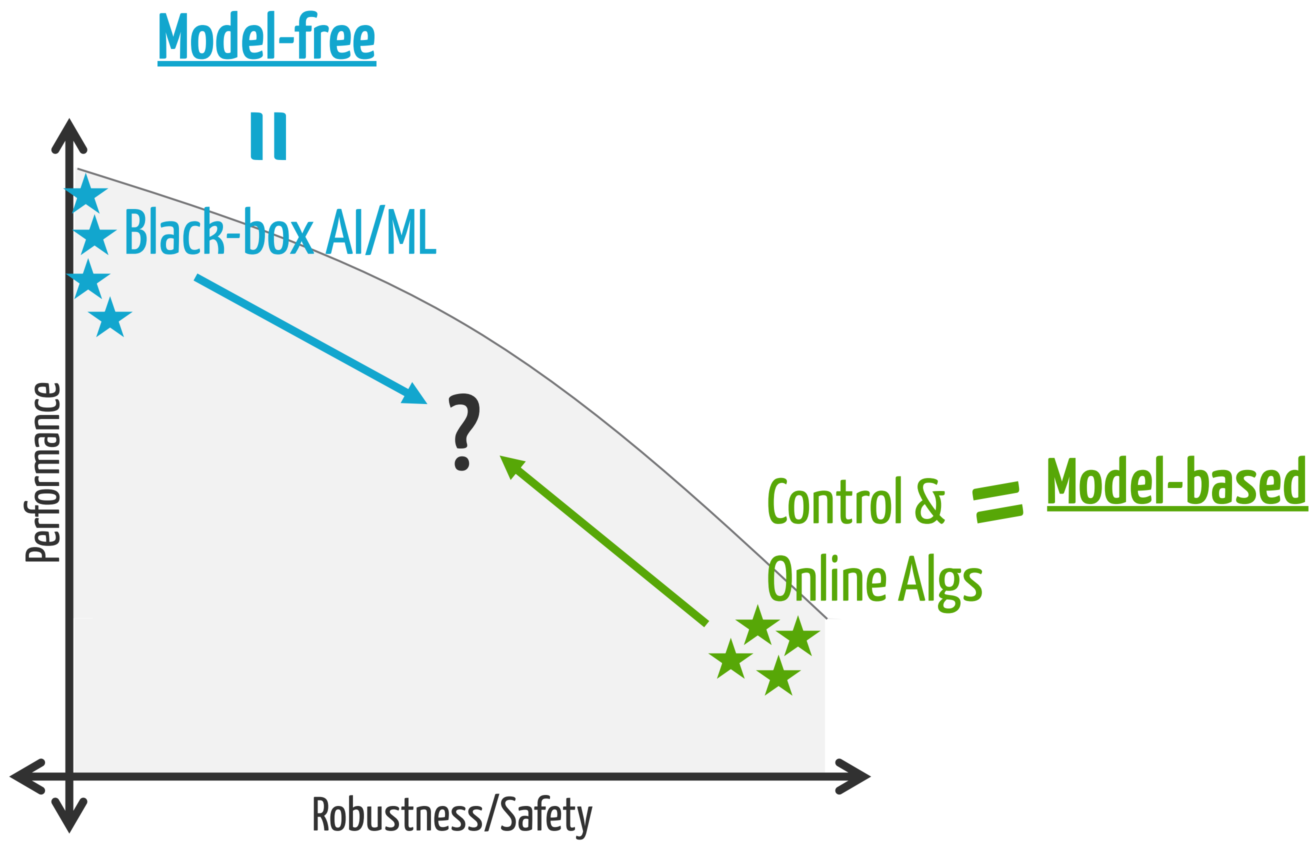


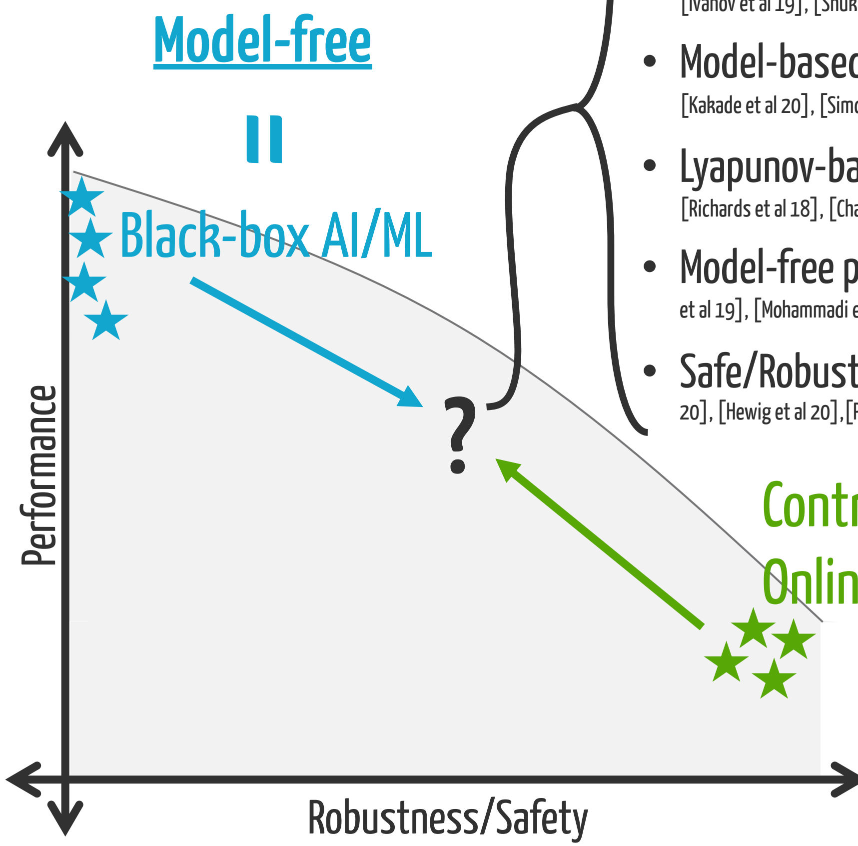

BEYOND LIMITS

Example: Co-generation plant (steam+electricity) with co-located wind generation



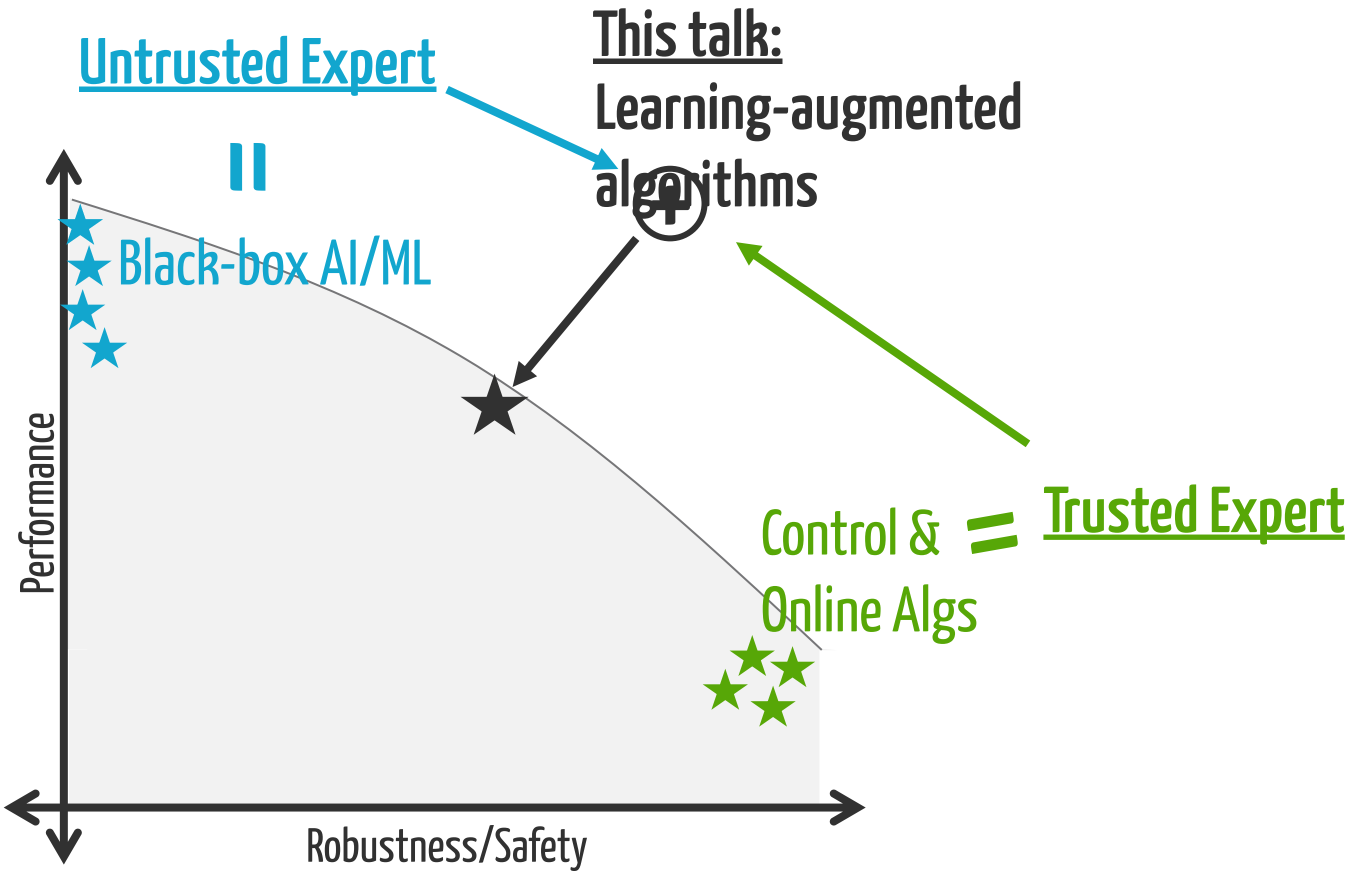

BEYOND LIMITS

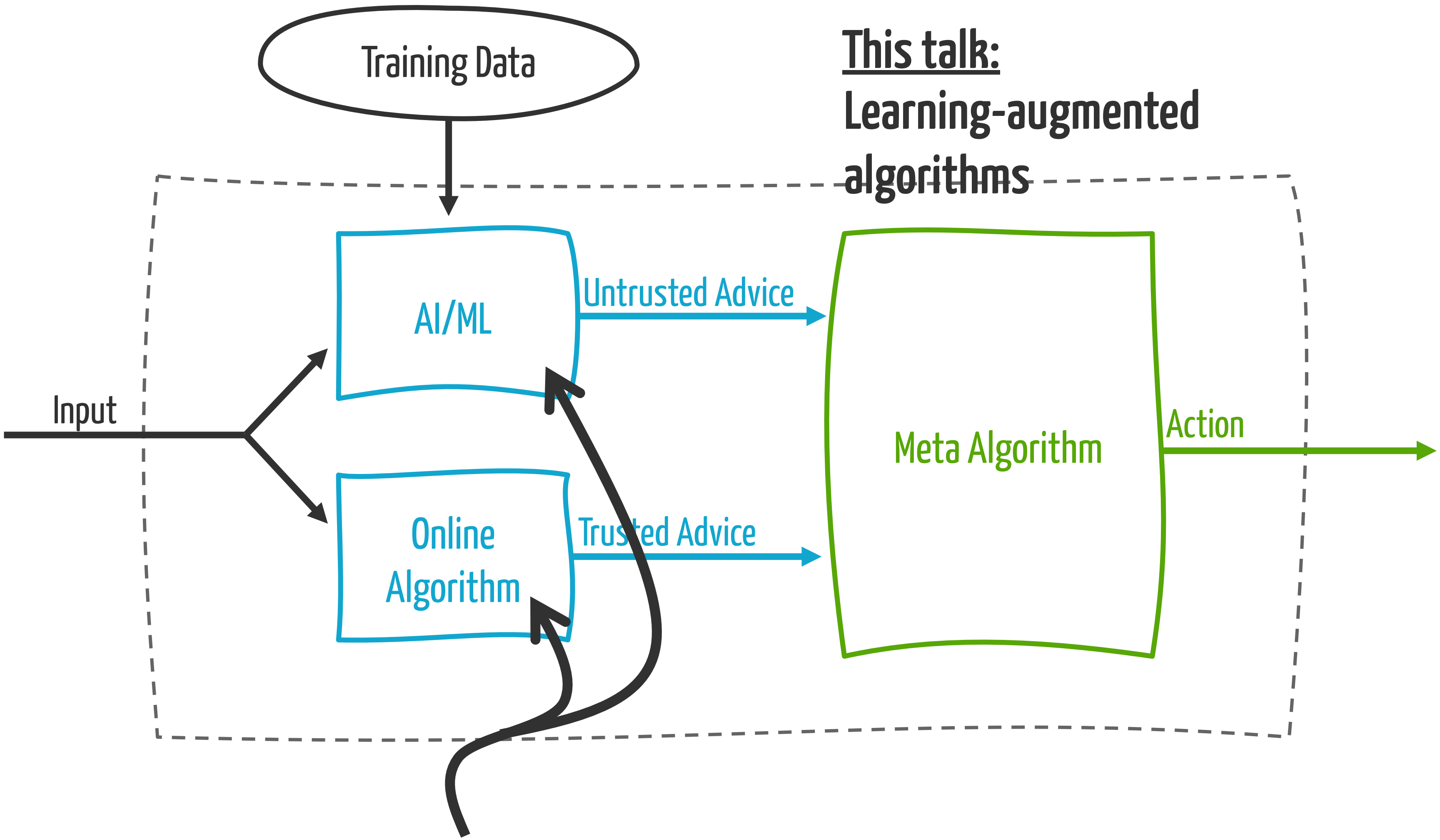

Model-free
=
Black-box AI/ML
?
Control &
Online Algs
=
Model-based
Performance
Robustness/Safety




- **Robust/adversarial training** [Ehlers 17], [Katz et al 17], [Maganti 17], [Dutta et al 18], [Tjeng et al 18], [Gehr 18,], [Salman 19], [Bak 20], [Fazlyab 19, 22], [Robey et al 21,22], [Eastwood et al 23]. ...
- **Post-training verification** [Huang et al 17], [Kuper et al 18], [Ivanov et al 19], [Shukla et al 19], [Matni et al 20], [Fazlyab et al 22], ...
- **Model-based RL in dynamical systems** [Recht 19], [Kakade et al 20], [Simchowitz & Foster 20], [Lale et al 21], ...
- **Lyapunov-based policy learning** [Chow et al 18], [Richards et al 18], [Chang et al 19], [Jin et al 20], [Shi et al 21], ...
- **Model-free policy search** [Fazel et al 18], [Malik et al 18], [Bu et al 19], [Mohammadi et al 19], [Li et al 19], [Qu et al 20], ...
- **Safe/Robust RL** [Garcia & Fernandez 15], [Fisac et al 19], [Taylor et al 20], [Hewig et al 20],[Panaganti at al 21, 22], [Shi et al 21, 22], ...

Control & Online Algs = Model-based

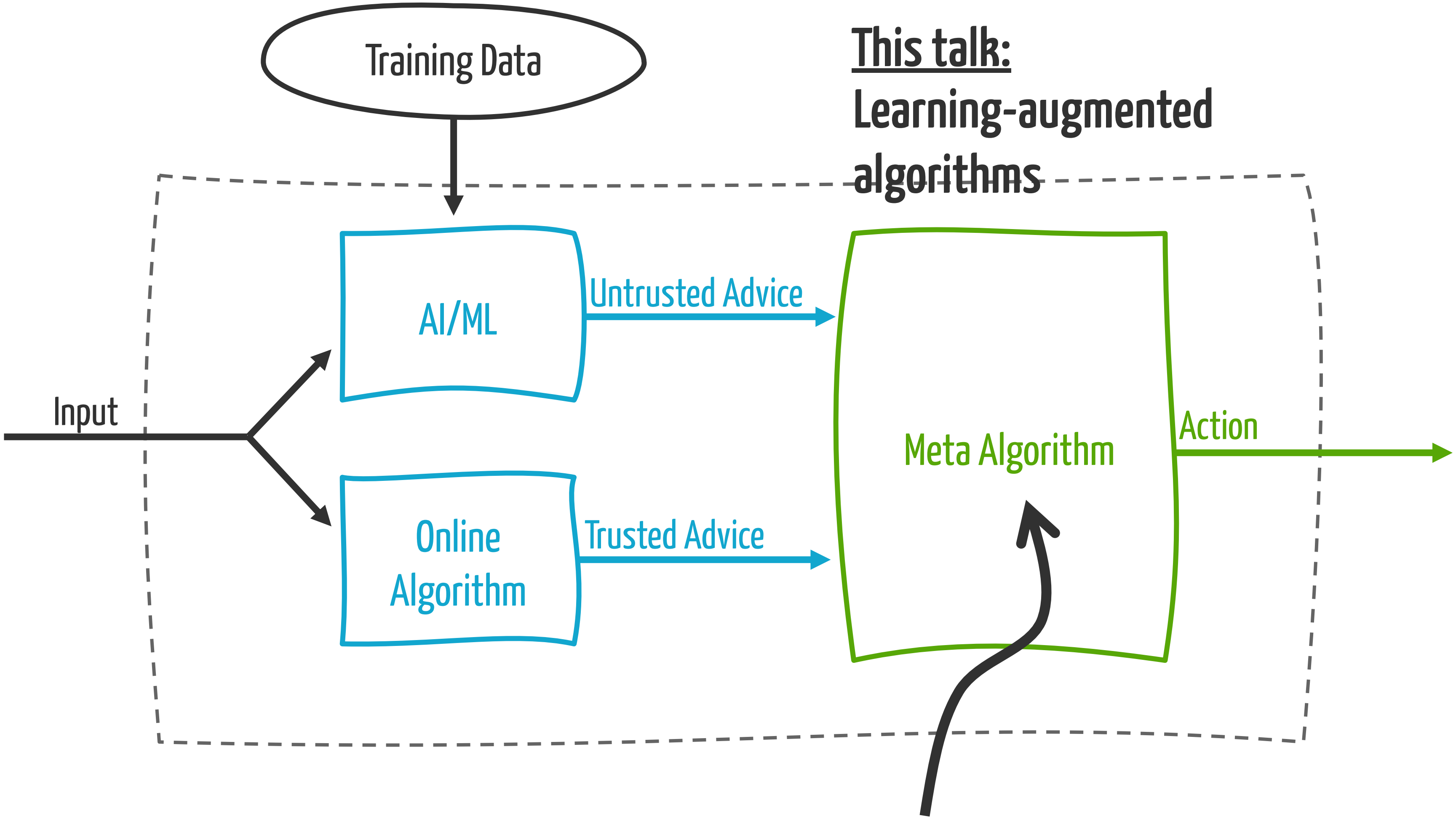

Untrusted Expert
=
Black-box AI/ML
This talk:
Learning-augmented algorithms
⊕
Control &
Online Algs
=
Trusted Expert
Performance
Robustness/Safety




**Treated as black boxes**

Allows adoption of new AI tools by combining with current trusted approach

**This talk:**
**Learning-augmented algorithms**




**How should advice be used?**
**Switch between them? Combine them? Hedge?**

bicompetitive guarantee

## Goal 1: Consistency

(Nearly) Match the performance of the untrusted expert (AI tool), when it does well.

$$Cost(Alg) \leq (1+\delta)Cost(Untrusted)$$

## Goal 2: Robustness

Always ensure a worst-case performance guarantee.

$Cost(Alg) \leq \gamma_{Alg}\ Cost(Opt)$, where $\gamma_{Alg}$ is "close to" $\gamma_{trusted}$

## Goal 3: Smoothness

Trade off between robustness and consistency smoothly in prediction error.

## Goal 4: Frugality / Succinctness

Use only as much advice as necessary to be robust and consistent.

Skip for today

# The study of learning augmented algorithms with <u>untrusted advice</u> is exploding

Introduced by [Lykouris & Vassilvitskii, 2018] in the context of online caching

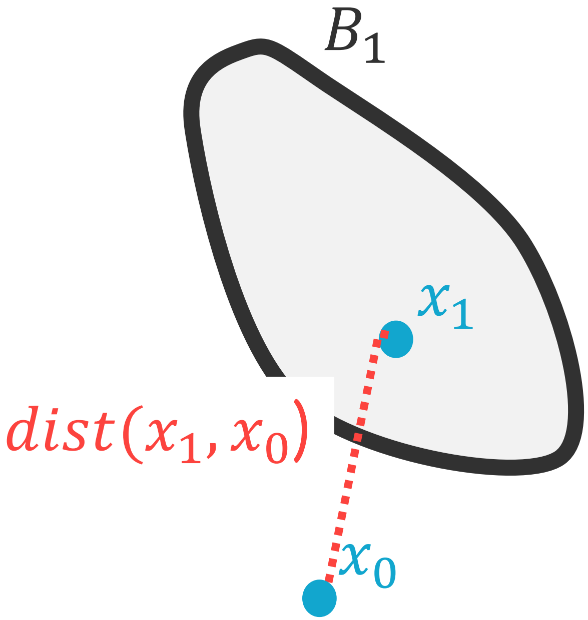
Since then, applied in a wide variety of settings:

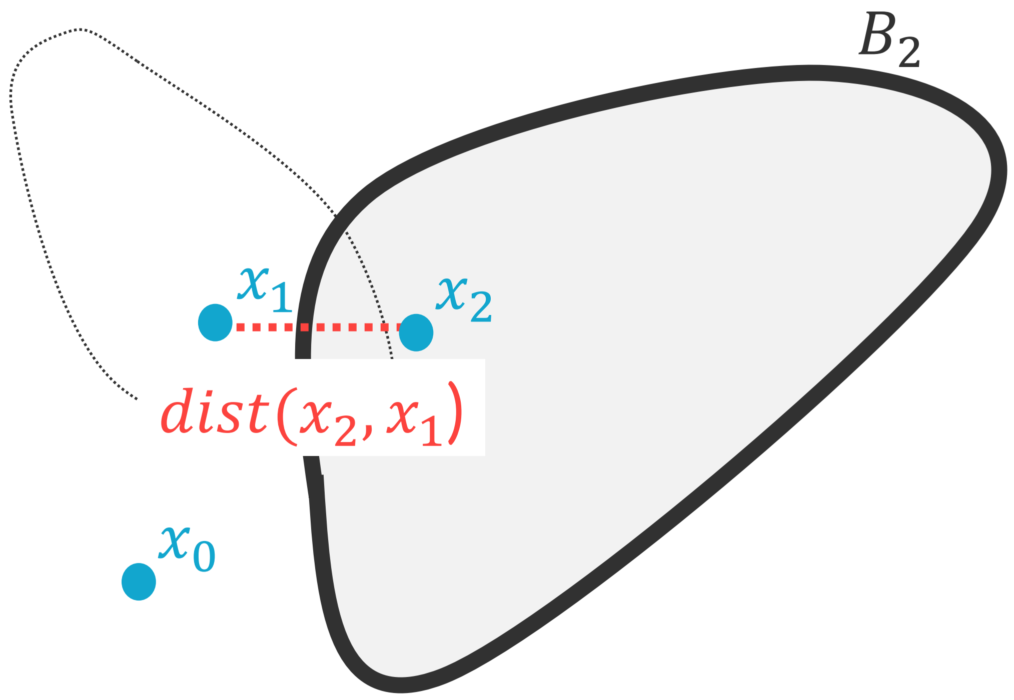
- ski rental [Purohit et al 18] [Angelopoulos et al 19] [Bamas et al 20] [Wei & Zhang 20], ...
- bloom filters [Mitzenmacher 18]
- online set cover [Bamas et al 20]
- online matching [Antoniadis et al 20]
- metrical task systems [Antoniadis et al 20]
- Scheduling [Scully et al 22]
- data center capacity [Rutten & Mukherjee 21]
- demand response [Lee et al 21]
- online optimization [Christianson et al 21]
- online conversion problems [Sun et al 21]
- convex body chasing [Christianson et al 21]
- linear quadratic control [Li et al 21]
- Online knapsack [Sun et al 22]

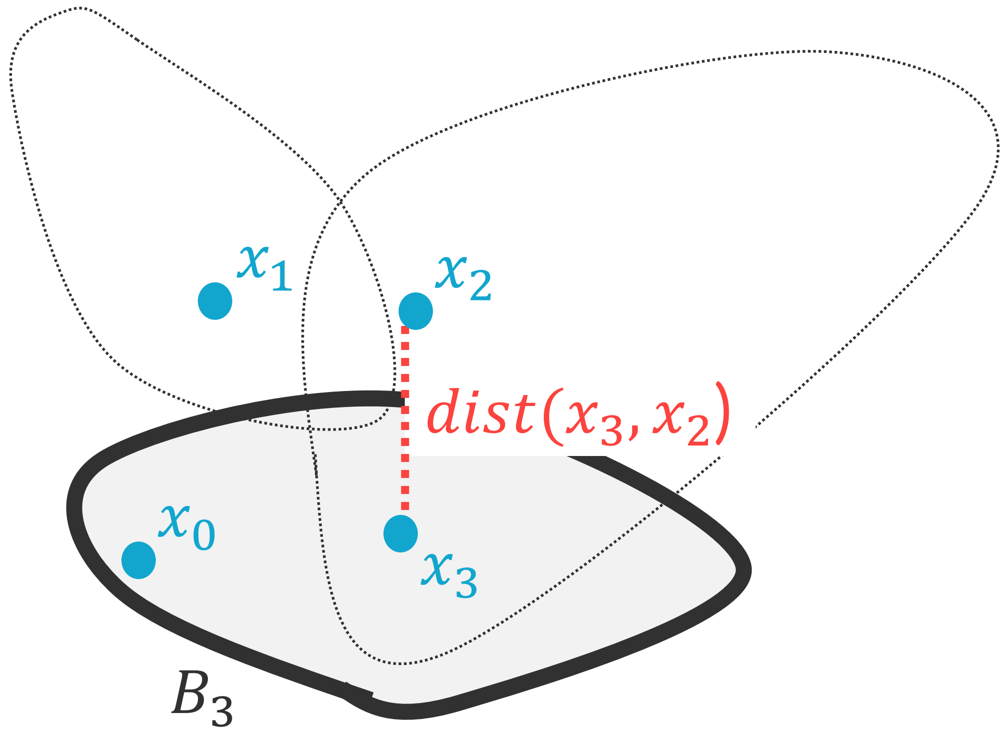
**Bibliography of 130+ papers at https://algorithms-with-predictions.github.io/**

This talk: Algorithm design & fundamental limits on the use of learning-augmented algorithms.

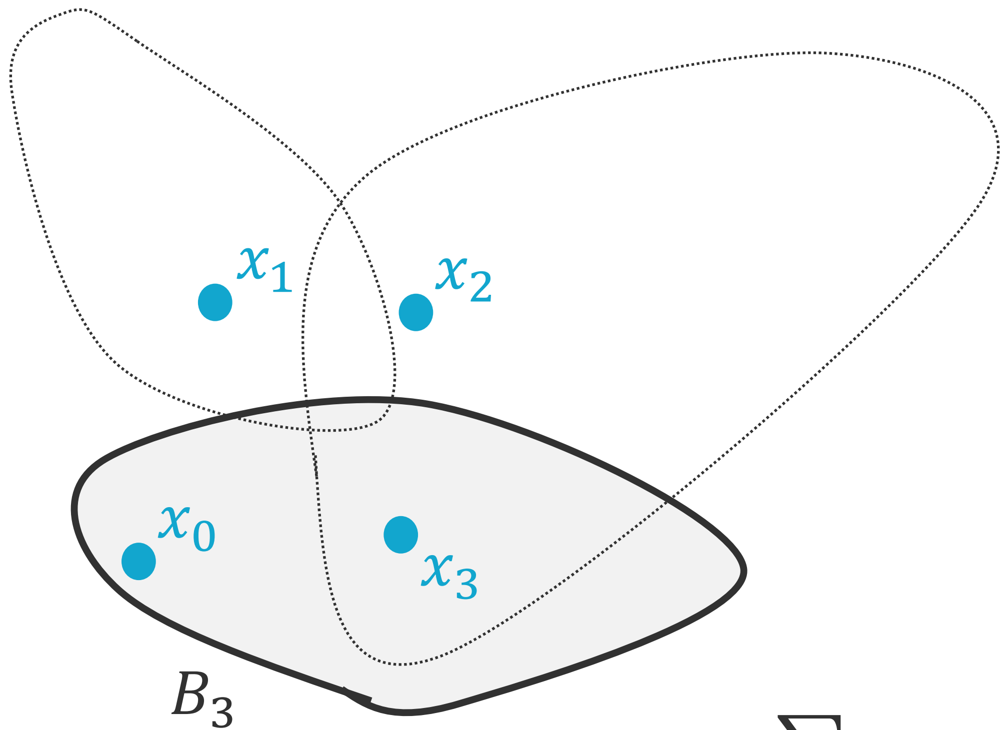
Running Example: Convex Body Chasing


$B_1$
$x_1$
$dist(x_1, x_0)$
$x_0$

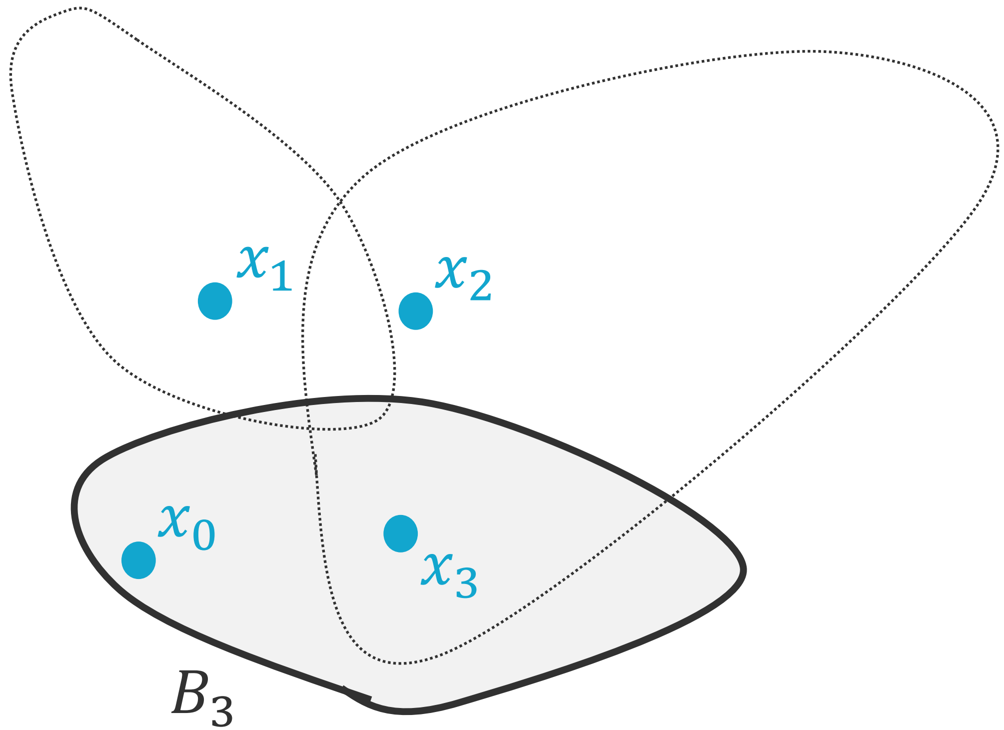

$B_2$
$x_1$
$x_2$
$dist(x_2, x_1)$
$x_0$



$x_1$
$x_2$
$dist(x_3, x_2)$
$x_0$
$x_3$
$B_3$




$$\min_{x_t \in B_t} \sum_t dist(x_t, x_{t-1})$$




How do you decide where to move without knowing the future?

# Convex body chasing has a long history & many applications

Reductions to online convex optimization and online control. Applications to data centers, video streaming, drone trajectory tracking, "learning to control" and "safe control", among others.

Exciting algorithmic progress in recent years [Antoniadis et al 16], [Bansal et al 20], [Bubeck et al 19], [Sellke 20], [Argue 20], [Bubeck et al 20], [Argue 21], ...

Theorem [Bubeck et al 20]. Moving to the Steiner point of the body each round obtains an $O\left(\min\left(d, \sqrt{d \log(T)}\right)\right)$-competitive ratio, and any online algorithms is $\Omega\left(\sqrt{d}\right)$.

dimension of action space

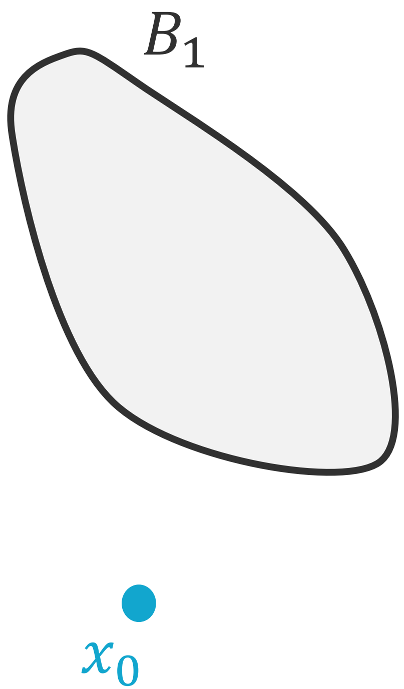
## Convex body chasing has a long history & many applications

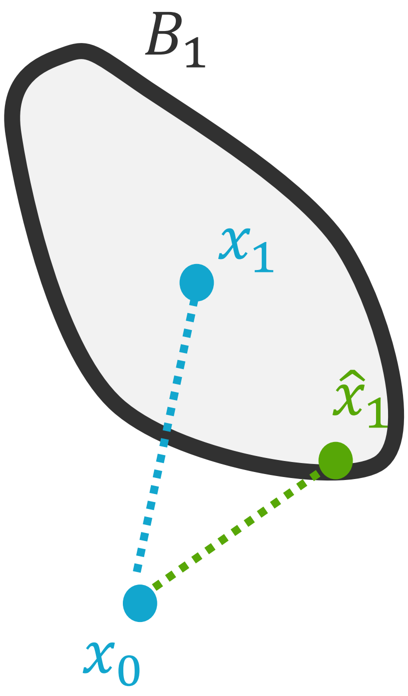
Reductions to online convex optimization and online control. Applications to data centers, video streaming, drone trajectory tracking, “learning to control” and “safe control”, among others.

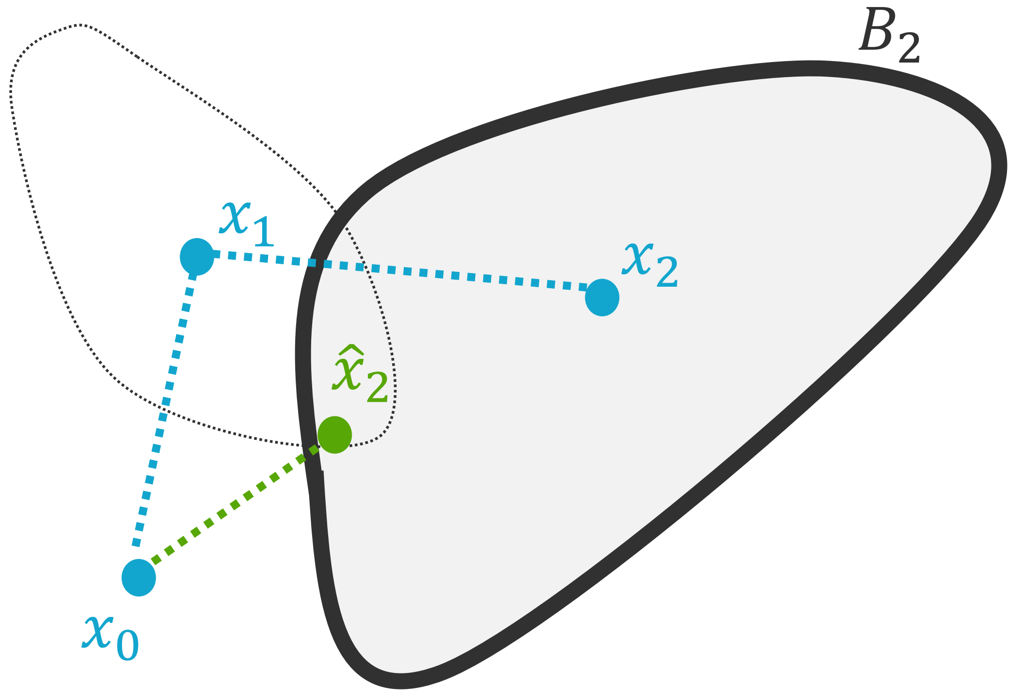
Exciting algorithmic progress in recent years [Antoniadis et al 16], [Bansal et al 20], [Bubeck et al 19], [Sellke 20], [Argue 20], [Bubeck et al 20], [Argue 21], …

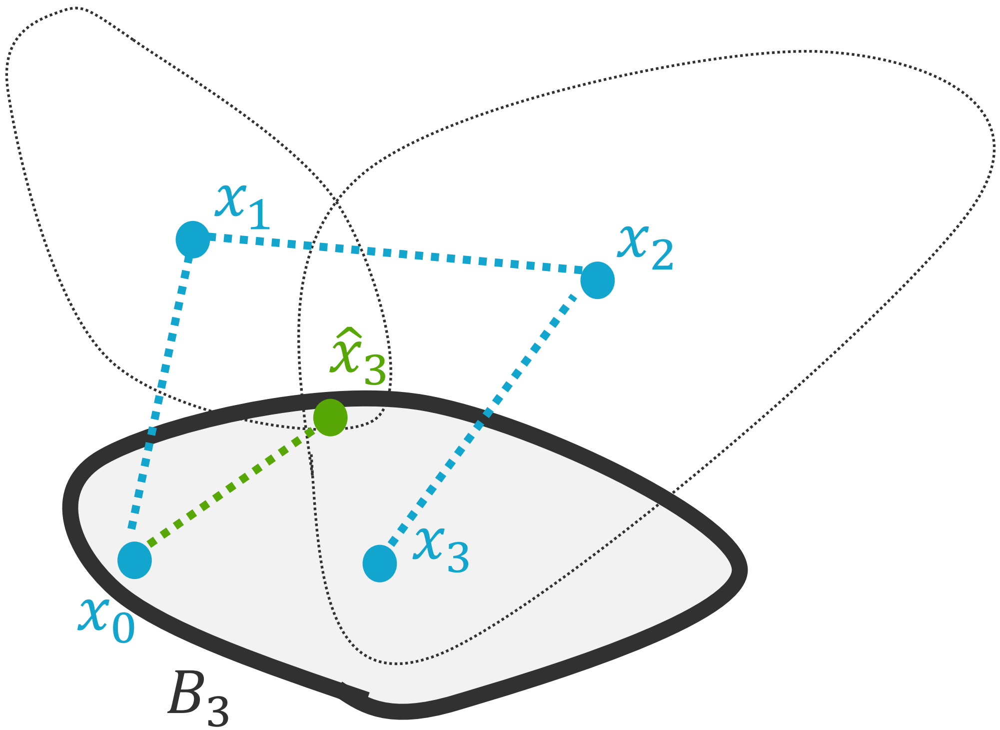
Theorem [Bubeck et al 20]. Moving to the Steiner point of the body each round obtains an $O\left(\min\left(d, \sqrt{d \log(T)}\right)\right)$-competitive ratio, and any online algorithms is $\Omega\left(\sqrt{d}\right)$.

**Choices of algorithm are quite conservative. Advice can help.**

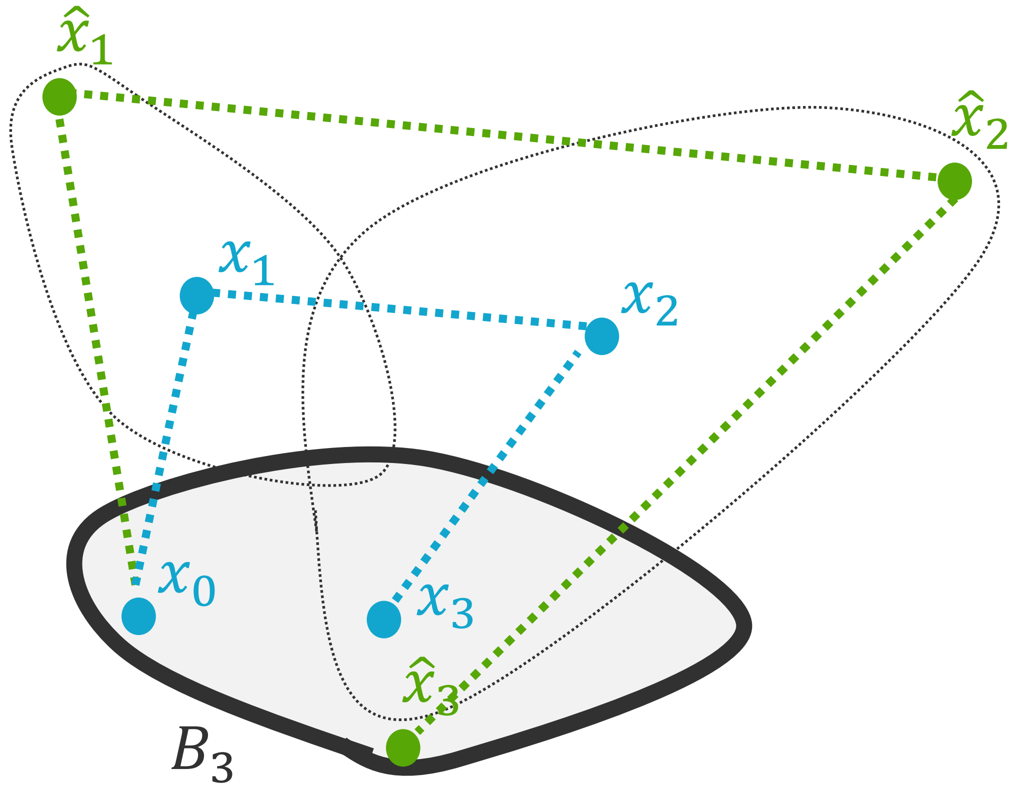

$B_1$
$x_0$


$B_1$
$x_1$
$\hat{x}_1$
$x_0$


$B_2$
$x_1$
$x_2$
$\hat{x}_2$
$x_0$




But the advice could have been bad...




**But the advice could have been bad...**

When should an algorithm “switch” between the trusted/untrusted advice?

How much “memory” is needed to decide between trusted/untrusted advice?

## Attempt 1: A switching algorithm

1. Follow the untrusted advice until total distance traveled is $r$.
2. Follow the trusted advice until total distance traveled is $r$.
3. Set $r \leftarrow 2r$

Treats advice as black boxes.

## Attempt 1: A switching algorithm

1. Follow the untrusted advice until total distance traveled is $r$.
2. Follow the trusted advice until total distance traveled is $r$.
3. Set $r \leftarrow 2r$ and repeat.

Optimize to bias toward consistency

Theorem. For nested convex body chasing, the switching algorithm is $(1+\delta)$-consistent & $O(dD/\delta)$-robust.

diameter of action space

## Attempt 1: A switching algorithm

1. Follow the untrusted advice until total distance traveled is $r$.
2. Follow the trusted advice until total distance traveled is $r$.
3. Set $r \leftarrow 2r$ and repeat.

Optimize to bias toward consistency

Theorem. For nested convex body chasing, the switching algorithm is $(1+\delta)$-consistent & $O(dD/\delta)$-robust.

"Best of both worlds": Black-box AI/ML imbued with robustness guarantee.
Constant factor loss in robustness yields near-optimal consistency.

## A Fundamental Limit

<u>Theorem.</u> For general convex body chasing, any switching algorithm that is robust must be at least 3-consistent.

<u>Theorem.</u> For nested convex body chasing, the switching algorithm is $(1+\delta)$-consistent & $O(dD/\delta)$-robust.

A Fundamental Limit

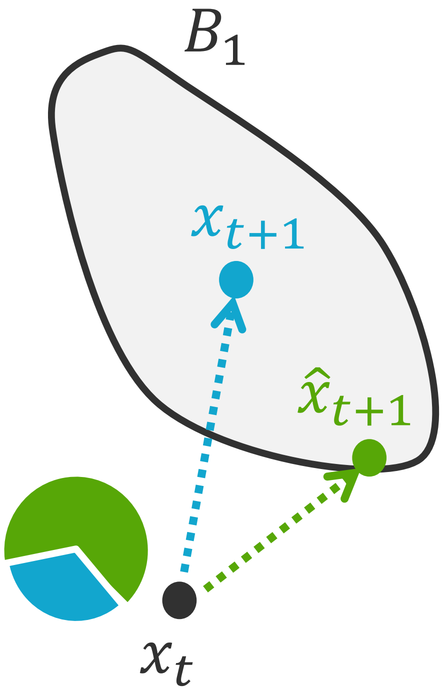
<u>Theorem.</u> For general convex body chasing, any switching algorithm that is robust must be at least 3-consistent.

<u>Theorem.</u> For general convex body chasing, any memoryless algorithm that is robust cannot have non-trivial consistency.

Consistency better than if advice had been ignored

# Attempt 2: A Randomized Algorithm

Apply multiplicative weights a la [Blum & Burch 2000]




<u>Multiplicative Weights [Blum & Burch 2000]</u>

Update weights for each expert

$$w_{ALG_i}^{t+1} = w_{ALG_i}^{t} \cdot (1-\beta)^{Cost_{t,t}(ALG_i)/D}$$

Update probability of following each expert

$$p_i^{t+1} = {}^{w_{ALG_i}}/_{\sum w_{ALG_i}}$$

Switch to other expert with probability proportional to mass transferred from $p_{ALG_i}^{t}$ to $p_{ALG_j}^{t+1}$

## Attempt 2: A Randomized Algorithm

Apply multiplicative weights a la [Blum & Burch 2000]

Theorem [Antoniadis et al 2020]. For general convex body chasing, multiplicative weights has cost

$$(1+\delta)\cdot 4\eta Cost(Untrusted) + O(D/\delta) \text{ [Consistency]}$$

and

$$(1+\delta)\cdot O(d) Cost(Opt) + O(D/\delta) \text{ [Robustness]}$$

Aggregate prediction quality of untrusted advice

## Attempt 2: A Randomized Algorithm

Apply multiplicative weights a la [Blum & Burch 2000]

Theorem [Antoniadis et al 2020]. For general convex body chasing, multiplicative weights has cost

$$(1+\delta)\cdot 4\eta Cost(Untrusted) + O(D/\delta)$$ [Consistency]

and

$$(1+\delta)\cdot O(d) Cost(Opt) + O(D/\delta)$$ [Robustness]

Multiplicative Weights has been used to incorporate untrusted advice broadly.
(This result extends to metrical task systems, MTS. )

## Attempt 2: A Randomized Algorithm

Apply multiplicative weights a la [Blum & Burch 2000]

Theorem [Antoniadis et al 2020]. For general convex body chasing, multiplicative weights has cost

$$(1+\delta) \cdot 4\eta Cost(Untrusted) + O(D/\delta) \text{ [Consistency]}$$

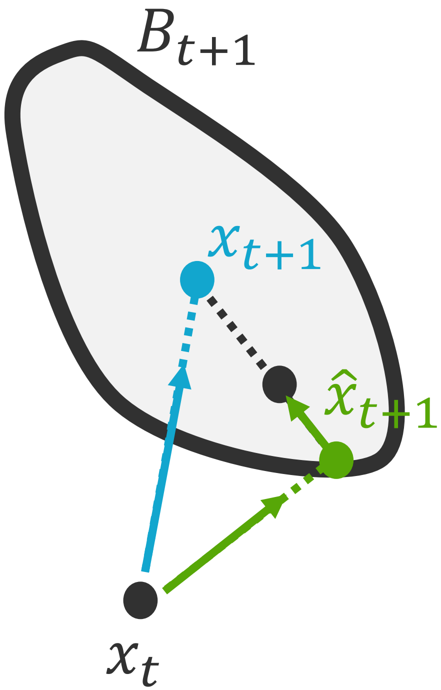
and

$$(1+\delta) \cdot O(d) Cost(Opt) + O(D/\delta) \text{ [Robustness]}$$

Diameter dependence

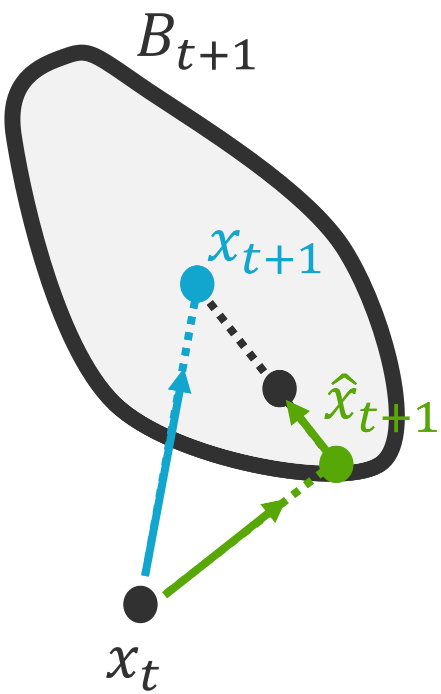
## Attempt 3: Exploiting Convexity

Adaptively choose a convex combination of the two advice points.



# Attempt 3: Exploiting Convexity

Adaptively choose a convex combination of the two advice points.




<u>Bicompetitive Line Chasing</u>

If $Cost_{0,t}(x) > \delta \cdot Cost_{0,t}(\hat{x})$

then follow $\hat{x}_{t+1}$

Else, take a greedy step from $\hat{x}_{t+1}$ toward $x_{t+1}$

with a series of radial projections depending on

$Cost_{t,t}(\hat{x})$ and $dist(\hat{x}_t, x_t)$.

# Attempt 3: Exploiting Convexity

Adaptively choose a convex combination of the two advice points.

<u>Theorem.</u> For general convex body chasing, the interpolation algorithm is $(\sqrt{2}+\delta)$-consistent & $O(d/\delta^2)$-robust.

**Dependence on the diameter $D$ is gone!**

## Attempt 3: Exploiting Convexity

Adaptively choose a convex combination of the two advice points.

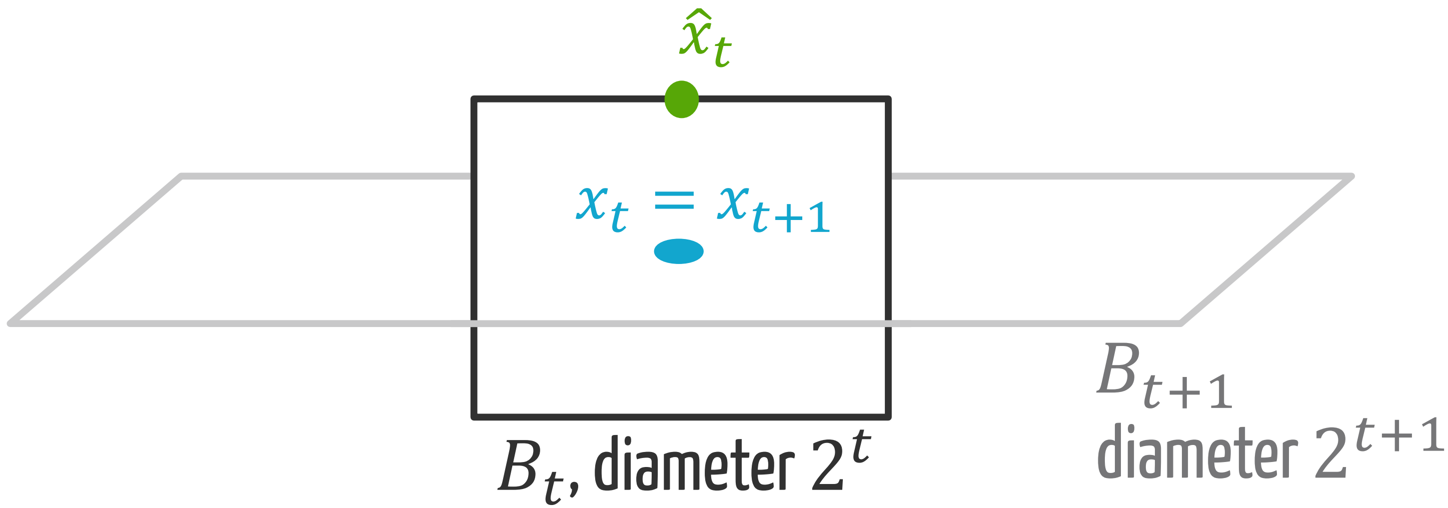
<u>Theorem.</u> For general convex body chasing, the interpolation algorithm is $(\sqrt{2} + \delta)$-consistent & $O(d/\delta^2)$-robust.

Adding robustness means sacrificing performance of black-box AI.

<u>Is this a fundamental limit</u>?

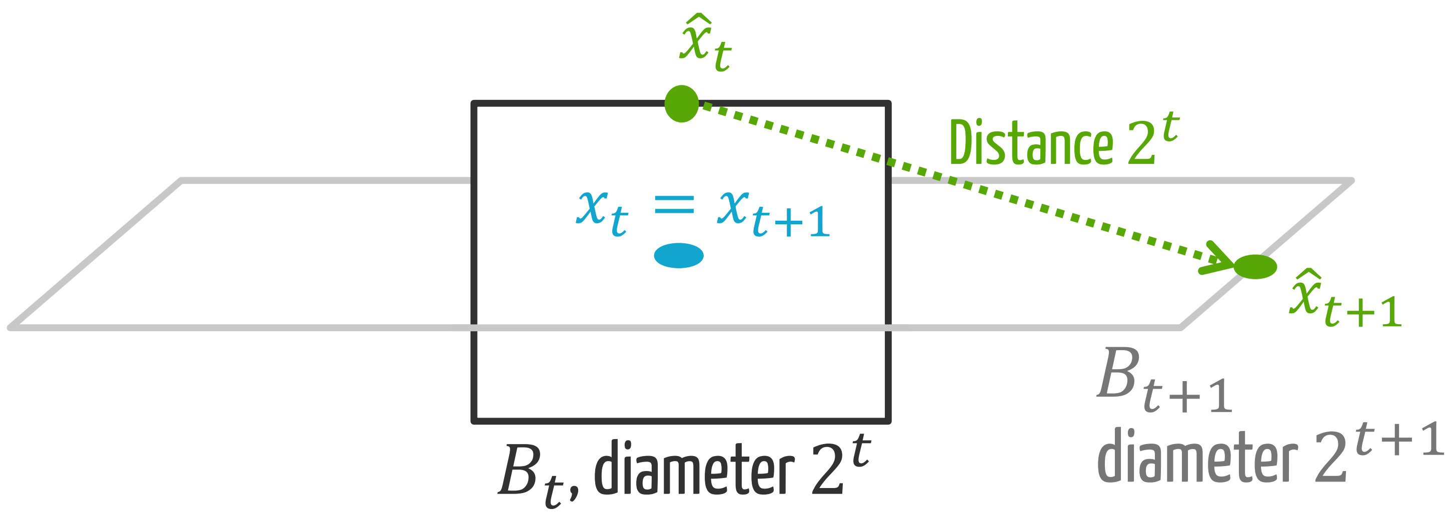
## A Fundamental Limit

<u>Theorem.</u> For general convex body chasing, given a $C$-competitive algorithm, any $(1+\delta)$-consistent algorithm is $2^{\Omega(1/\delta)}C$-robust.



## A Fundamental Limit

**Theorem.** For general convex body chasing, given a $C$-competitive algorithm, any $(1+\delta)$-consistent algorithm is $2^{\Omega(1/\delta)}C$-robust.

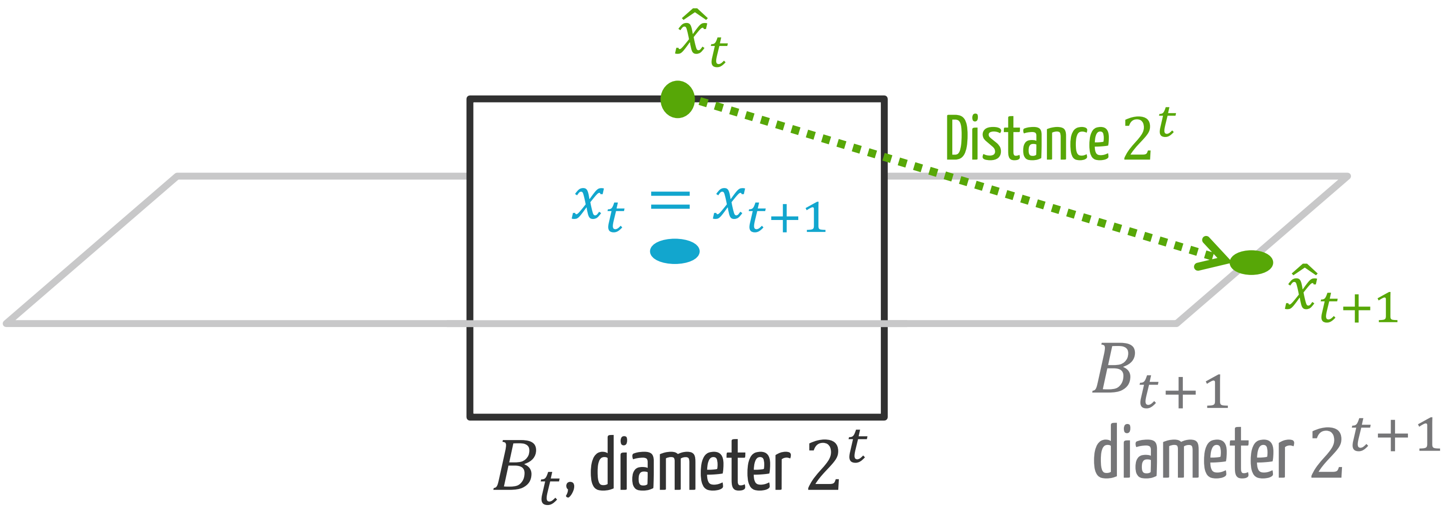



Key Property: $Cost_{0,t+1}(\hat{x}) = dist(\hat{x}_{t+1}, x_{t+1})$

(Note: $L1$ distance, not Euclidean distance.)

# A Fundamental Limit

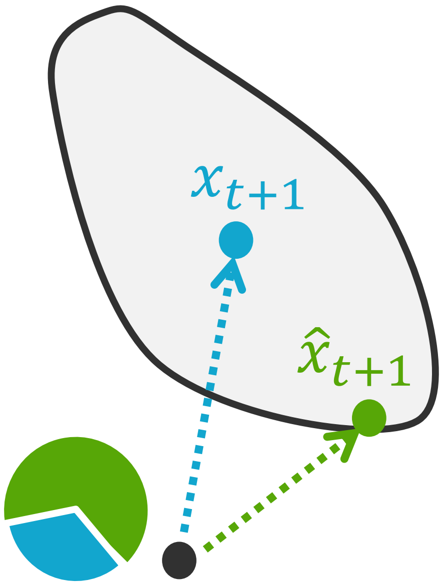
**Theorem.** For general convex body chasing, given a $C$-competitive algorithm, any $(1+\delta)$-consistent algorithm is $2^{\Omega(1/\delta)}C$-robust.




1. Any consistent algorithm must start following $\hat{x}_t$.
2. No algorithm can move more than $\delta/2$ probability to $x_t$ in any round.

So, at $T = 1/\delta$, only ½ probability can be on $x_T$, which means the total cost is at least $2^T = 2^{1/\delta}$.

# An Optimal Algorithm: Distance-Adaptive Robust weight Transport (DART)




<u>DART</u>

If $Cost_{0,t}(x) > \delta/4 \cdot Cost_{0,t}(\hat{x})$

then follow $\hat{x}_{t+1}$

Else, update probability of following the advice

$$p_{ADV}^{t+1} = \max\left(p_{ADV}^{t} - \frac{\delta Cost_{t,t}(\hat{x}_t)}{4dist(\hat{x}_t, x_t)}, 0\right)$$

Sample action through optimal transport plan (Wasserstein-1) for $p_{ALG_i}^{t} \to p_{ALG_j}^{t+1}$

## An Optimal Algorithm: Distance-Adaptive Robust weight Transport (DART)

Theorem. For general convex body chasing, DART is $(1+\delta)$-consistent and $2^{O(1/\delta)}O(n)$-robust.

## An Optimal Algorithm: Distance-Adaptive Robust weight Transport (DART)

<u>Theorem.</u> For general convex body chasing, DART is $(1+\delta)$-consistent and $2^{O(1/\delta)}O(n)$-robust.

<u>Theorem.</u> For convex body chasing with bounded diameter DART is $(1+\delta)$-consistent and $O(1/\delta)$-robust with an additive $O(D/\delta)$.

<u>Theorem.</u> For metrical task systems DART is $(1+\delta)$-consistent and$2^{O(1/\delta)}O(\log^2 n)$-robust.

<u>Theorem.</u> For $k$-server, DART is $(1+\delta)$-consistent and $O(k/\delta)$-robust.

<u>Theorem.</u> For $k$-function chasing in $\mathbb{R}$, DART is $(1+\delta)$-consistent and $O(k/\delta)$-robust.

## An Optimal Algorithm: Distance-Adaptive Robust weight Transport (DART)

<u>Theorem.</u> For general convex body chasing, DART is $(1+\delta)$-consistent and $2^{O(1/\delta)}O(n)$-robust.

Matches state of the art

<u>Theorem.</u> For convex body chasing with bounded diameter DART is $(1+\delta)$-consistent and $O(1/\delta)$-robust with an additive $O(D/\delta)$.

1st w/o $D$ dependence

<u>Theorem.</u> For metrical task systems DART is $(1+\delta)$-consistent and $2^{O(1/\delta)}O(\log^2 n)$-robust.

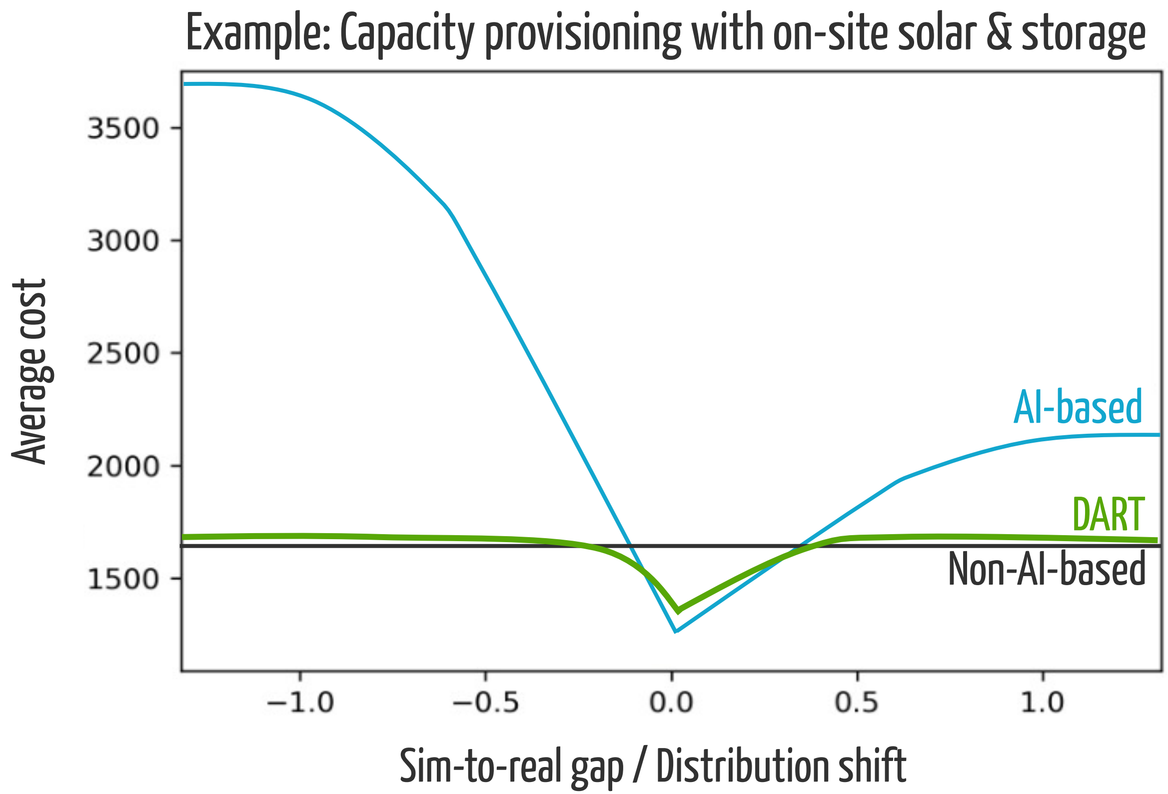
Prior: $O(1/\delta^{k-1})$

<u>Theorem.</u> For $k$-server, DART is $(1+\delta)$-consistent and $O(k/\delta)$-robust.

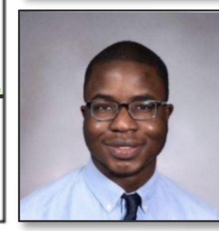
1st w/o $D$ dependence

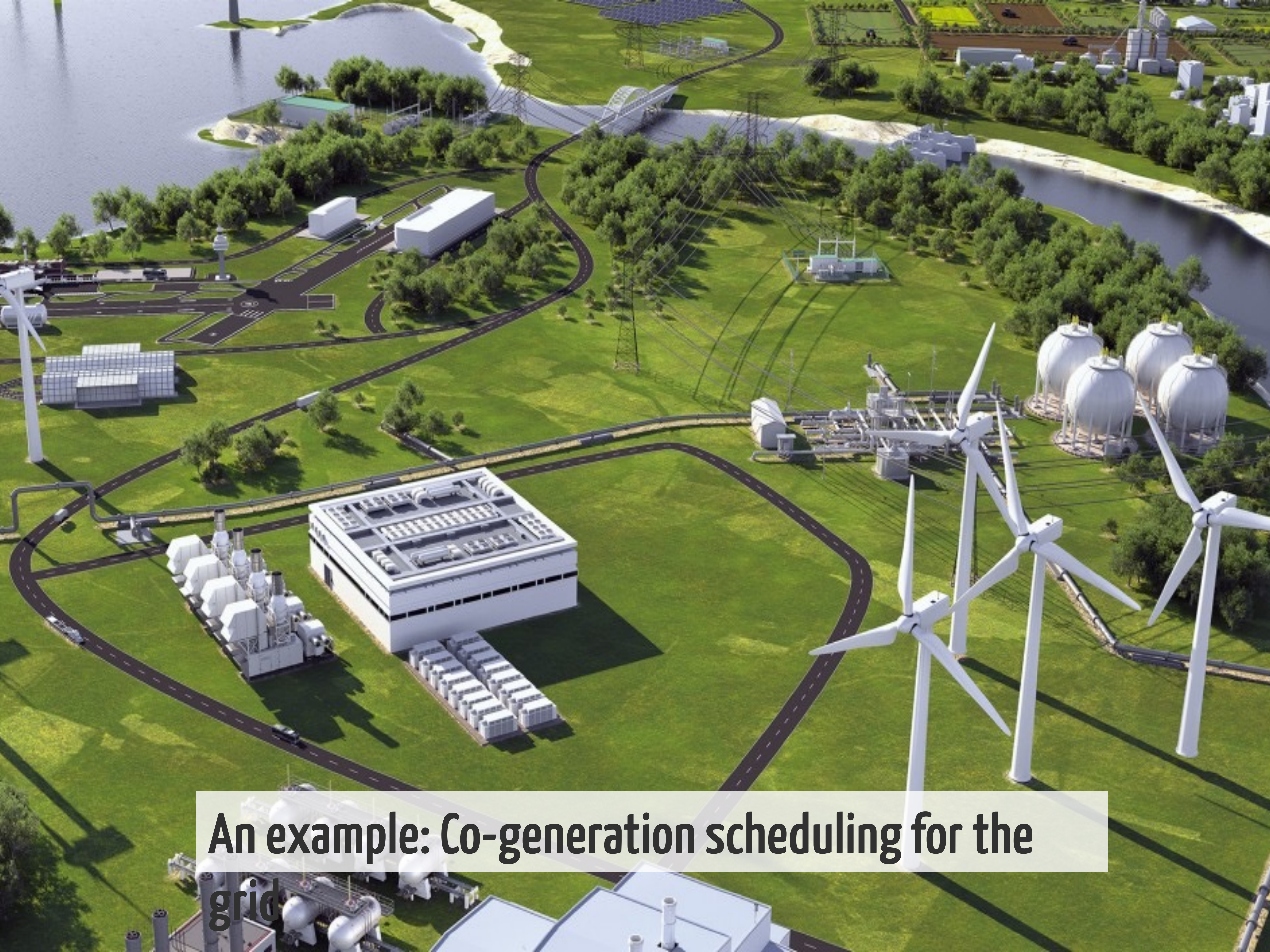
<u>Theorem.</u> For $k$-function chasing in $\mathbb{R}$, DART is $(1+\delta)$-consistent and $O(k/\delta)$-robust.

# An example: Carbon-first Cloud Computing

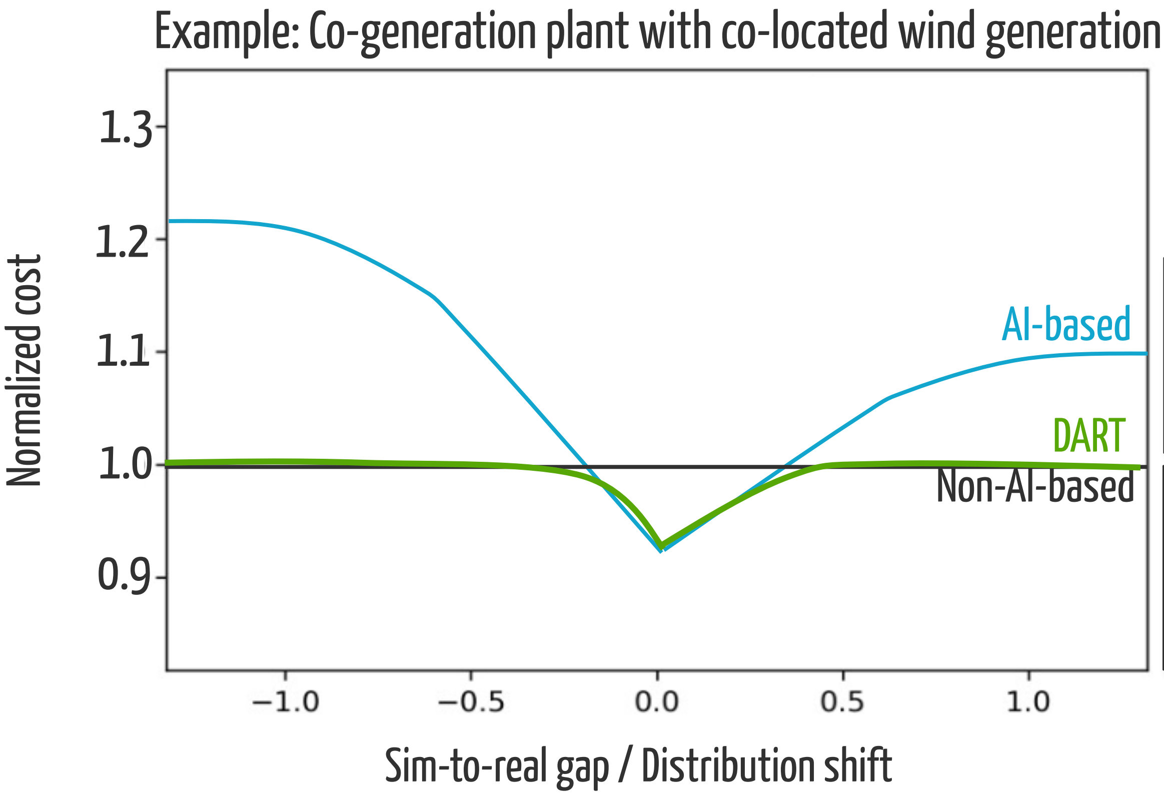


Example: Capacity provisioning with on-site solar & storage
Average cost
3500
3000
2500
2000
1500
−1.0
−0.5
0.0
0.5
1.0
Sim-to-real gap / Distribution shift
AI-based
DART
Non-AI-based



# An example: Co-generation scheduling for the grid

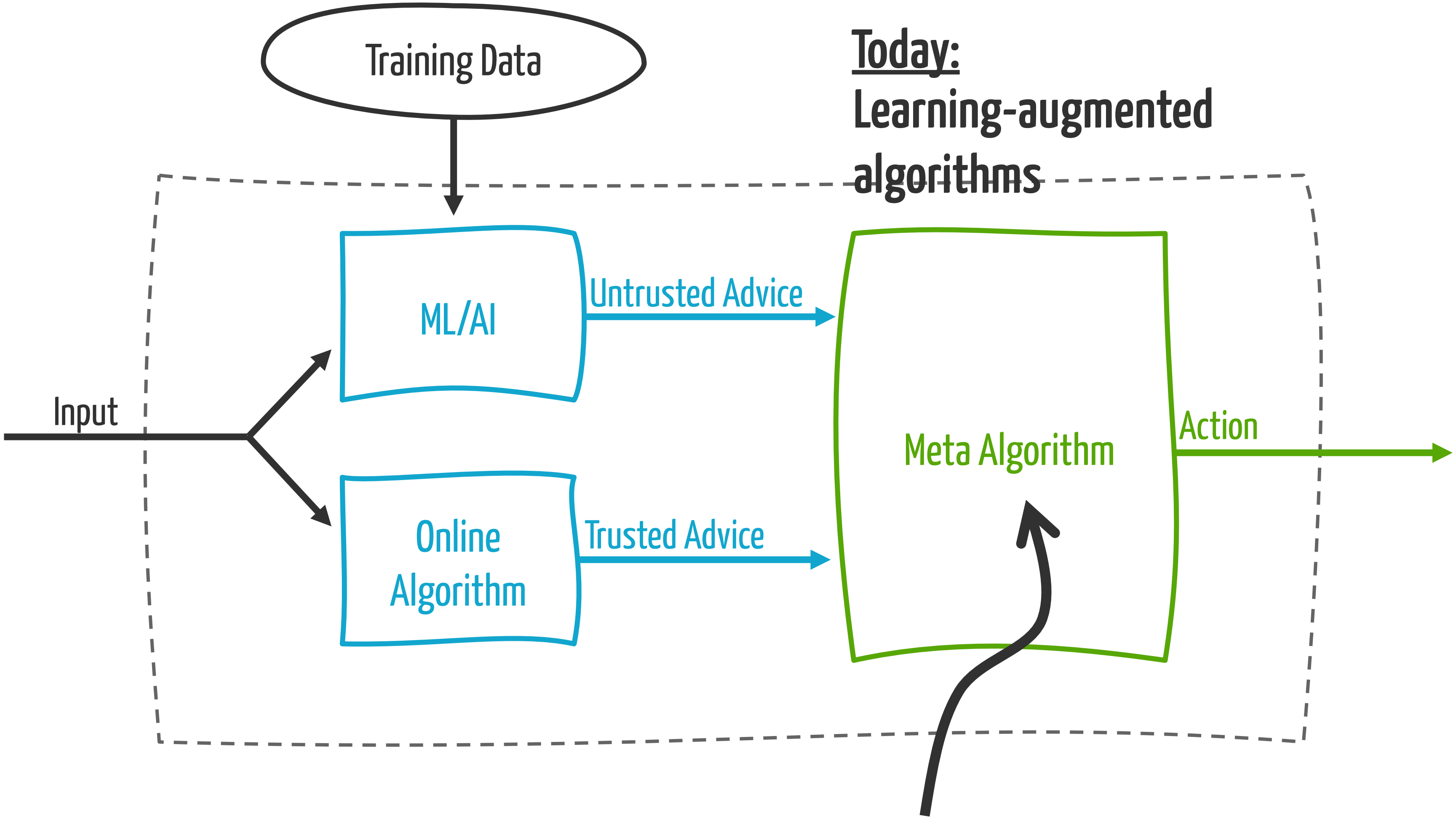

Example: Co-generation plant with co-located wind generation
1.3
1.2
1.1
1.0
0.9
Normalized cost
AI-based
DART
Non-AI-based
−1.0
−0.5
0.0
0.5
1.0
Sim-to-real gap / Distribution shift

This talk: Algorithm design & fundamental limits on the use of learning-augmented algorithms.

Running Example: Convex Body Chasing

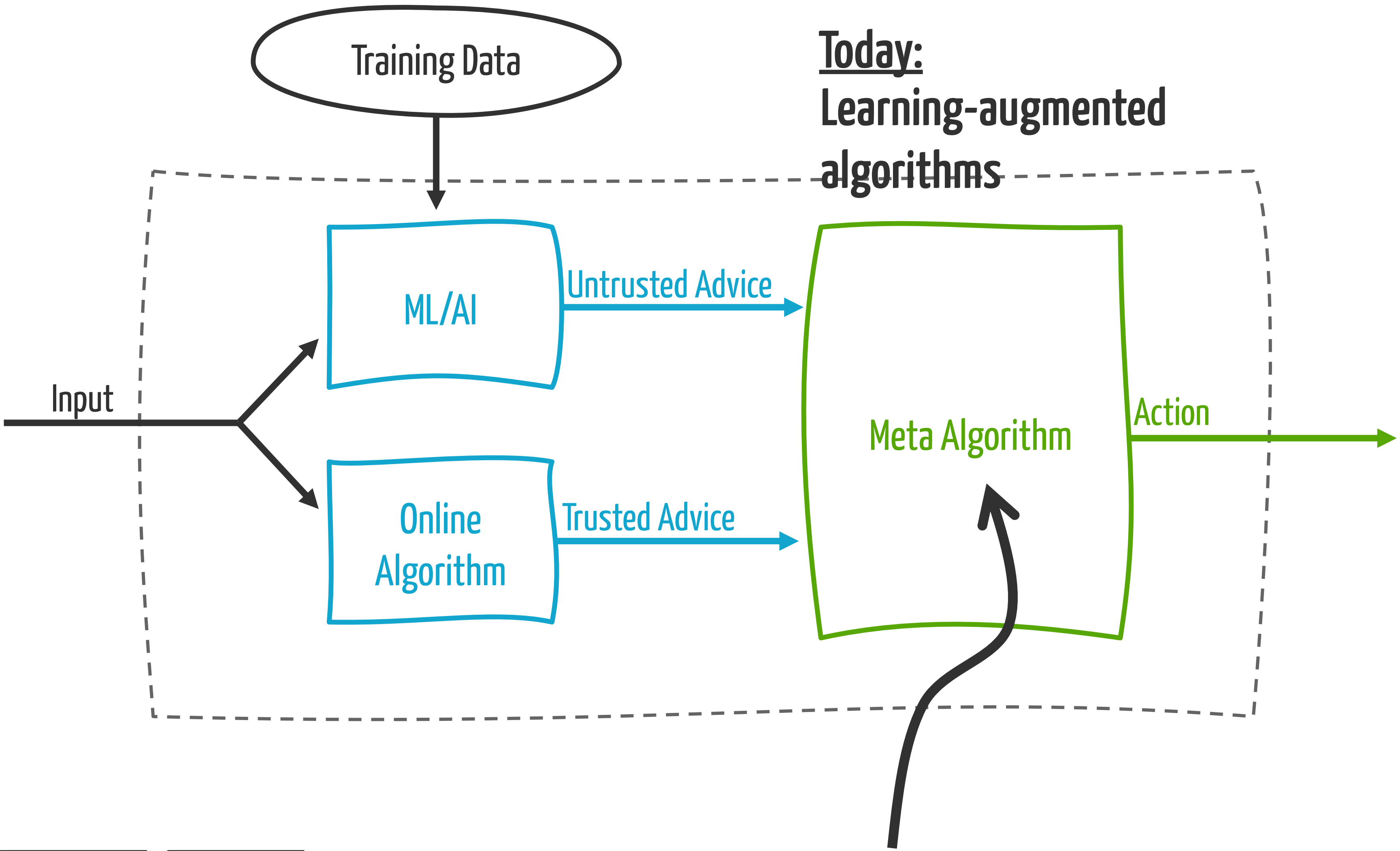
Applications: Carbon-aware data centers, co-generation scheduling, voltage control, drone trajectory tracking, …

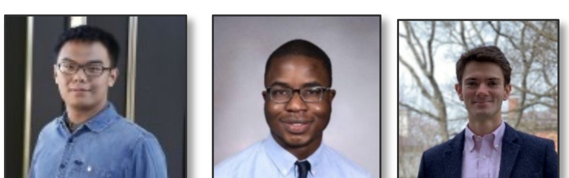
**Today:**
**Learning-augmented algorithms**




**How should advice be used?**
**Switch between them? Combine them?**

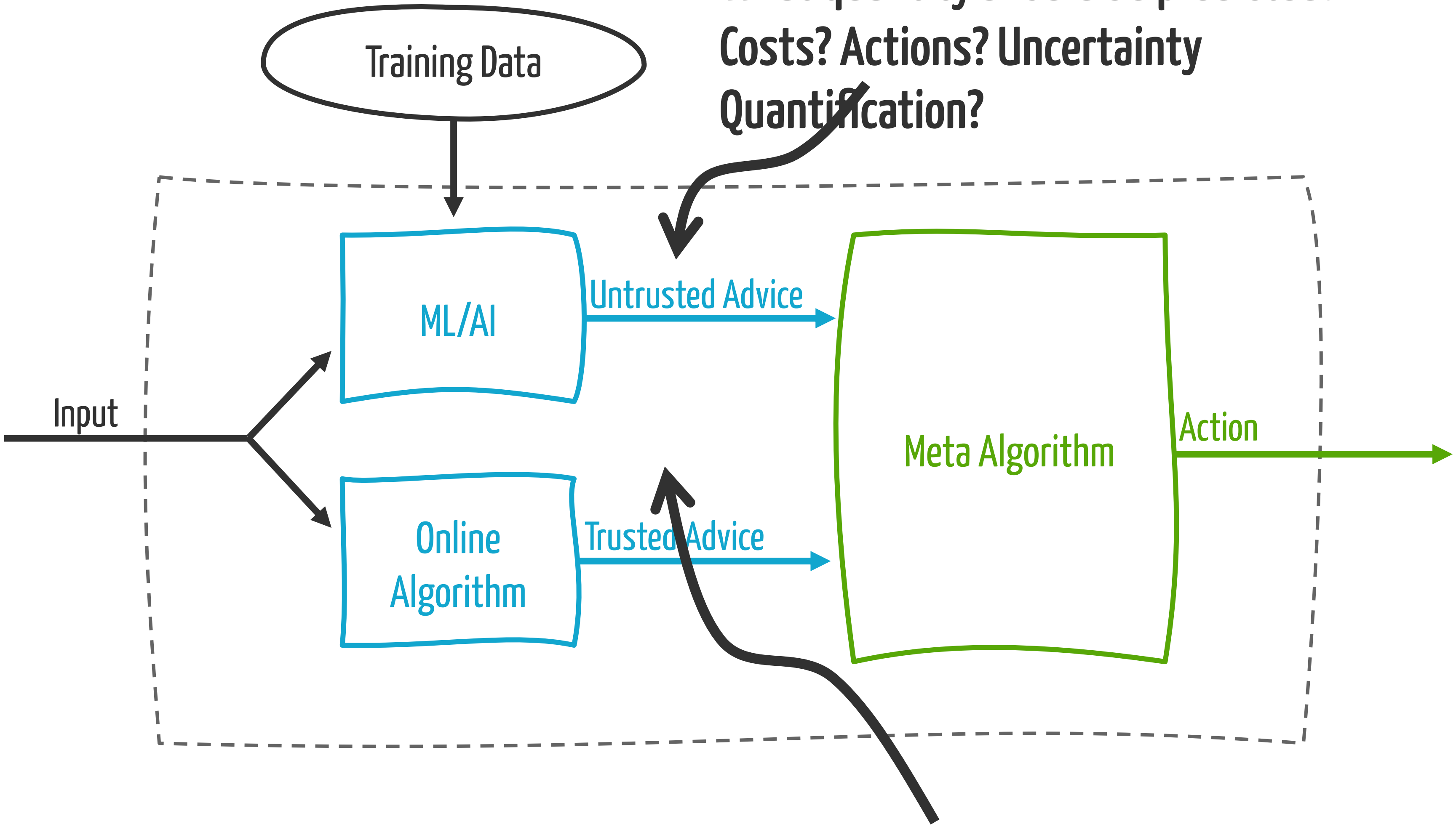
**Today:**
Learning-augmented algorithms

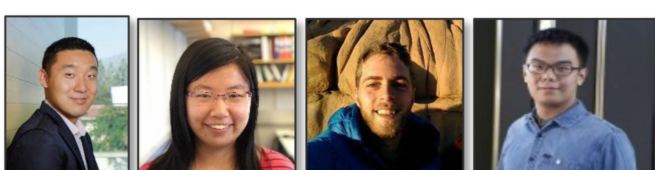


Can we move beyond robustness & consistency?
Average-case? Smoothness? Frugality? Memory-dependence?

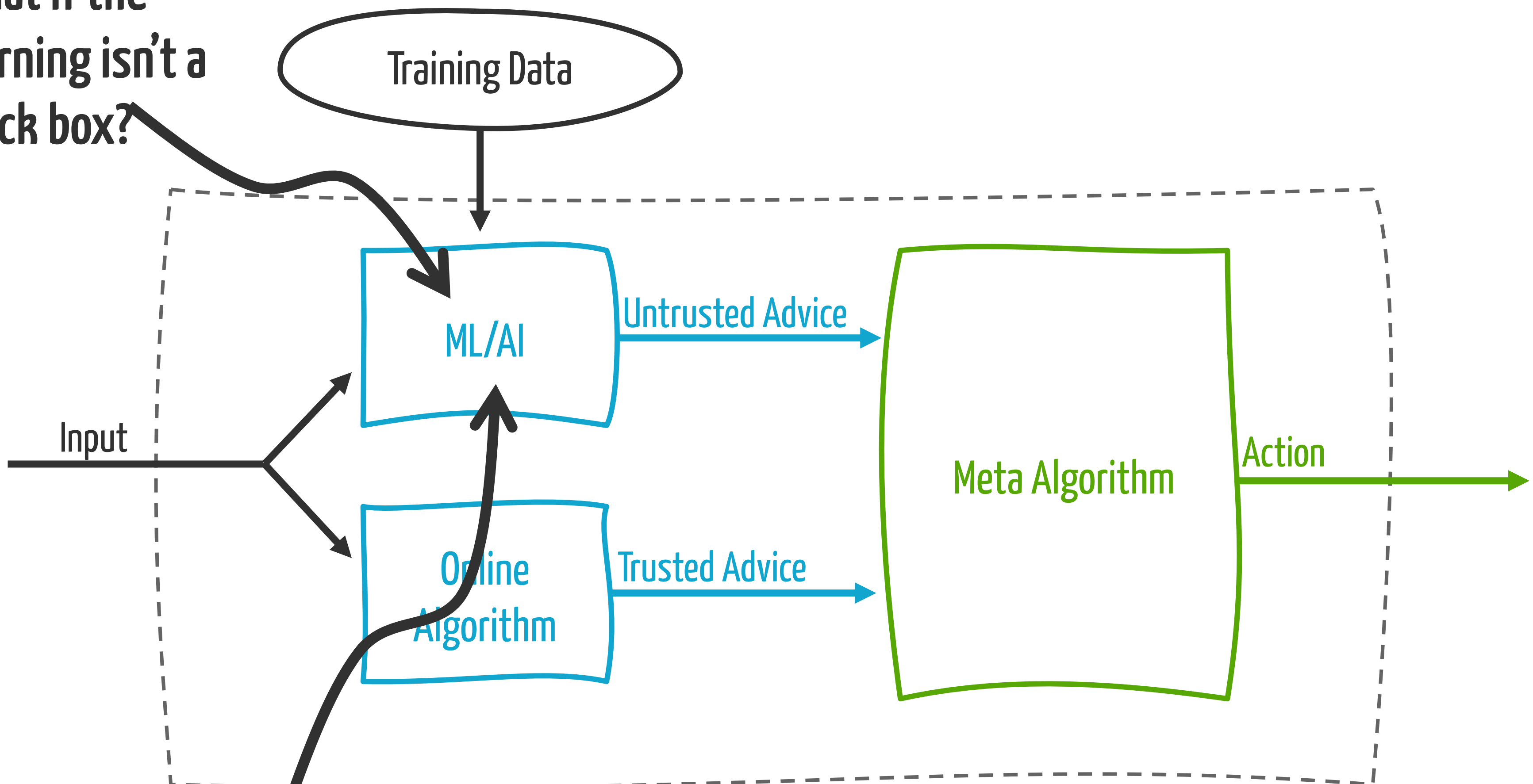
What quantity should be predicted? Costs? Actions? Uncertainty Quantification?



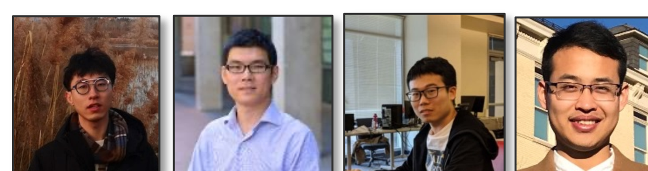



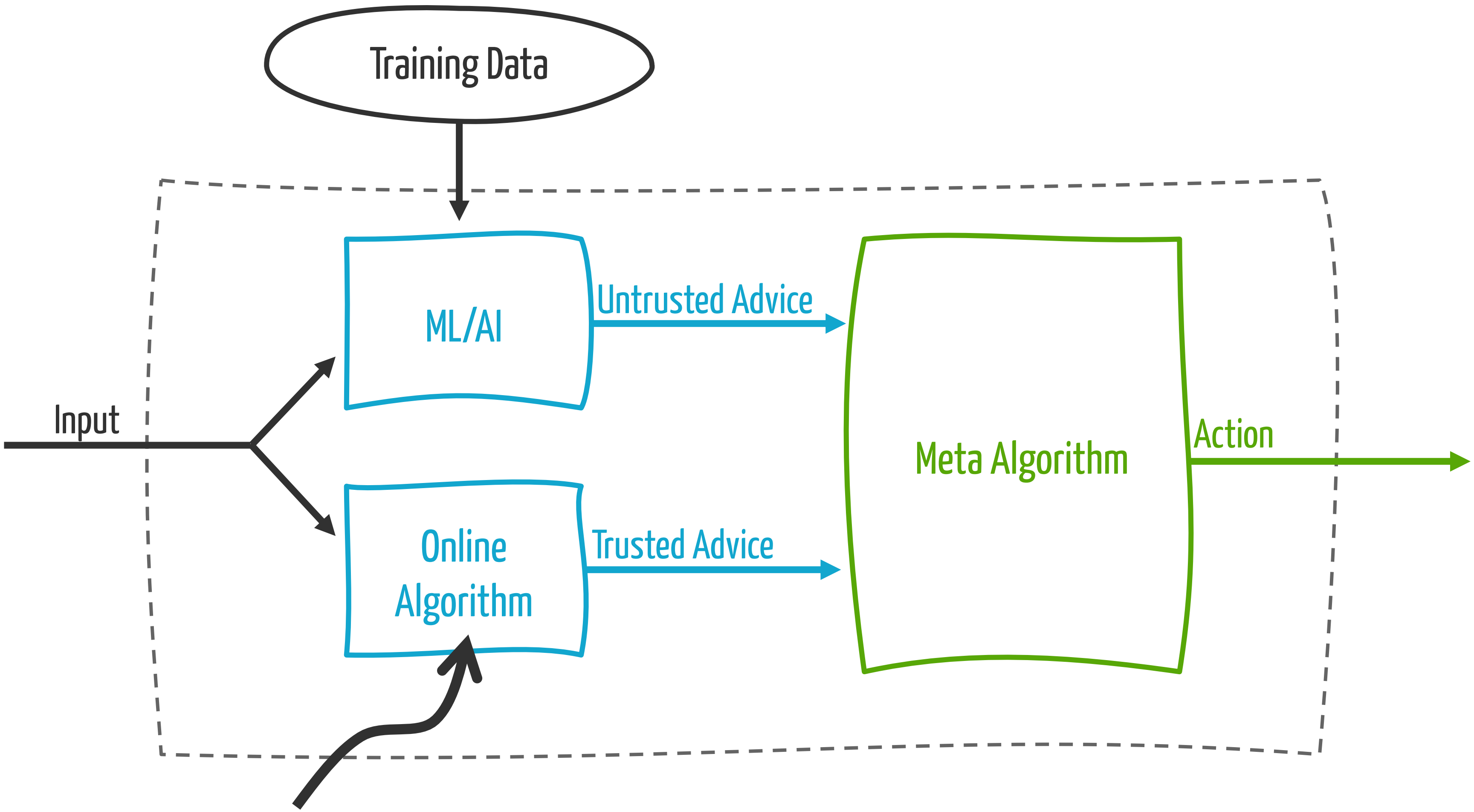
What if there are multiple untrusted/trusted advisors?
What if you're not sure which is the trusted advisor?

What if the learning isn't a black box?

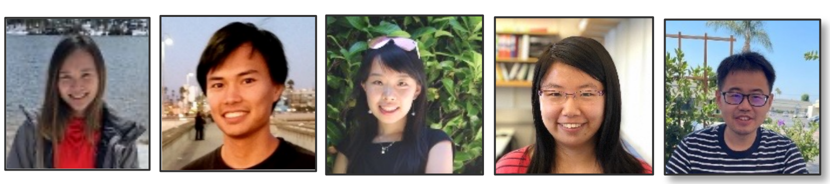



What if the ML model is trained online?

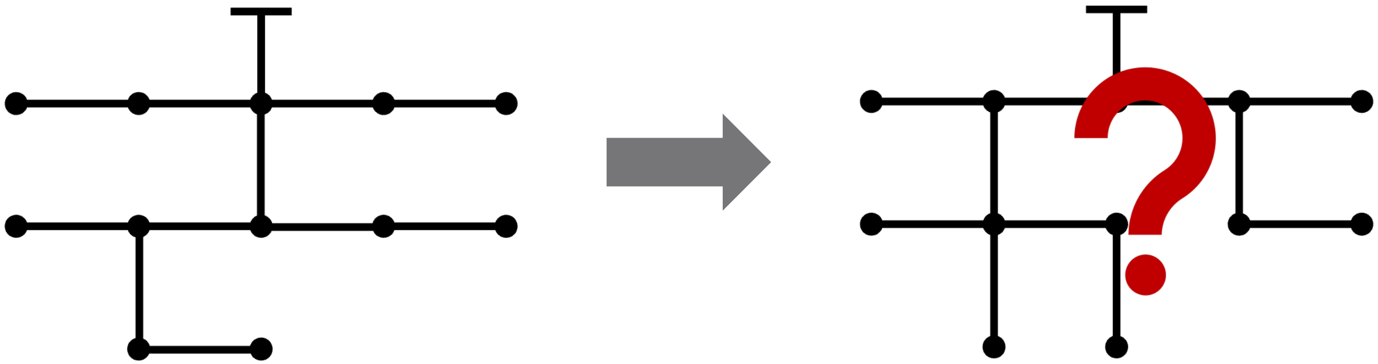




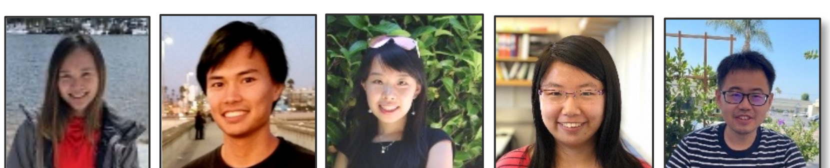

**What if the model needs to be learned?**



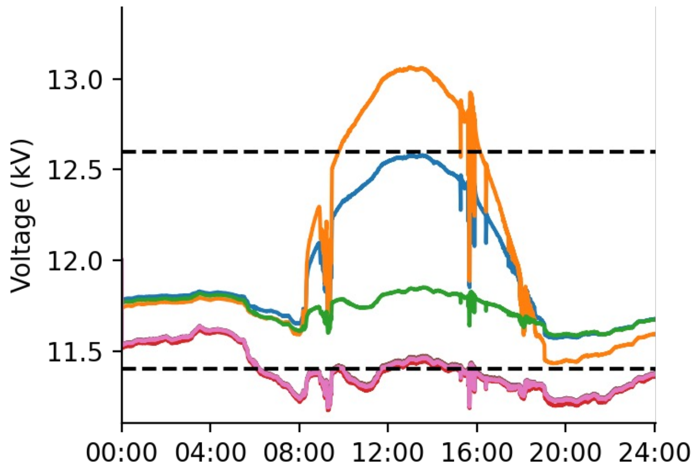
# A teaser: Online voltage control with unknown grid topology



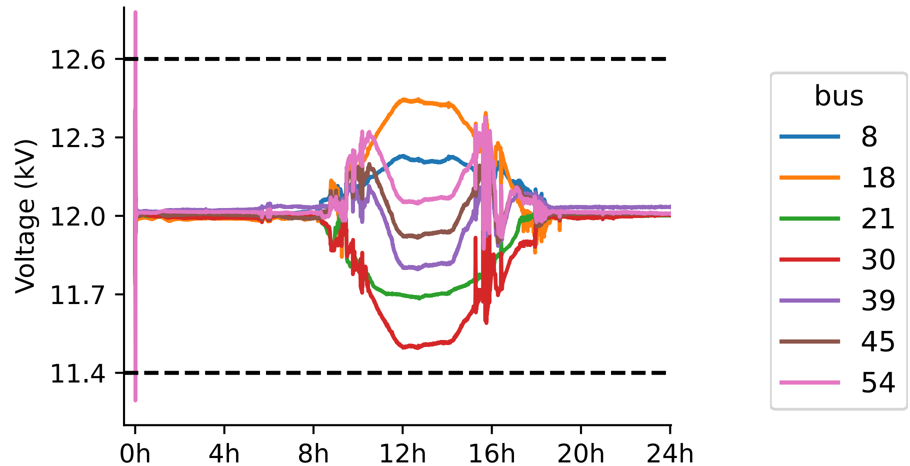
What if the model needs to be learned?



# A teaser: Online voltage control with unknown grid topology

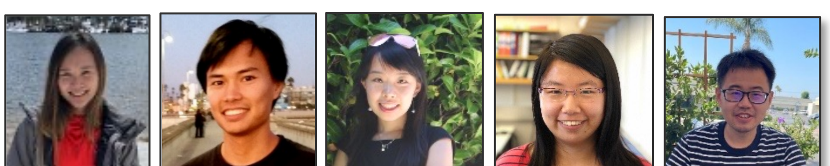



Control based on original model

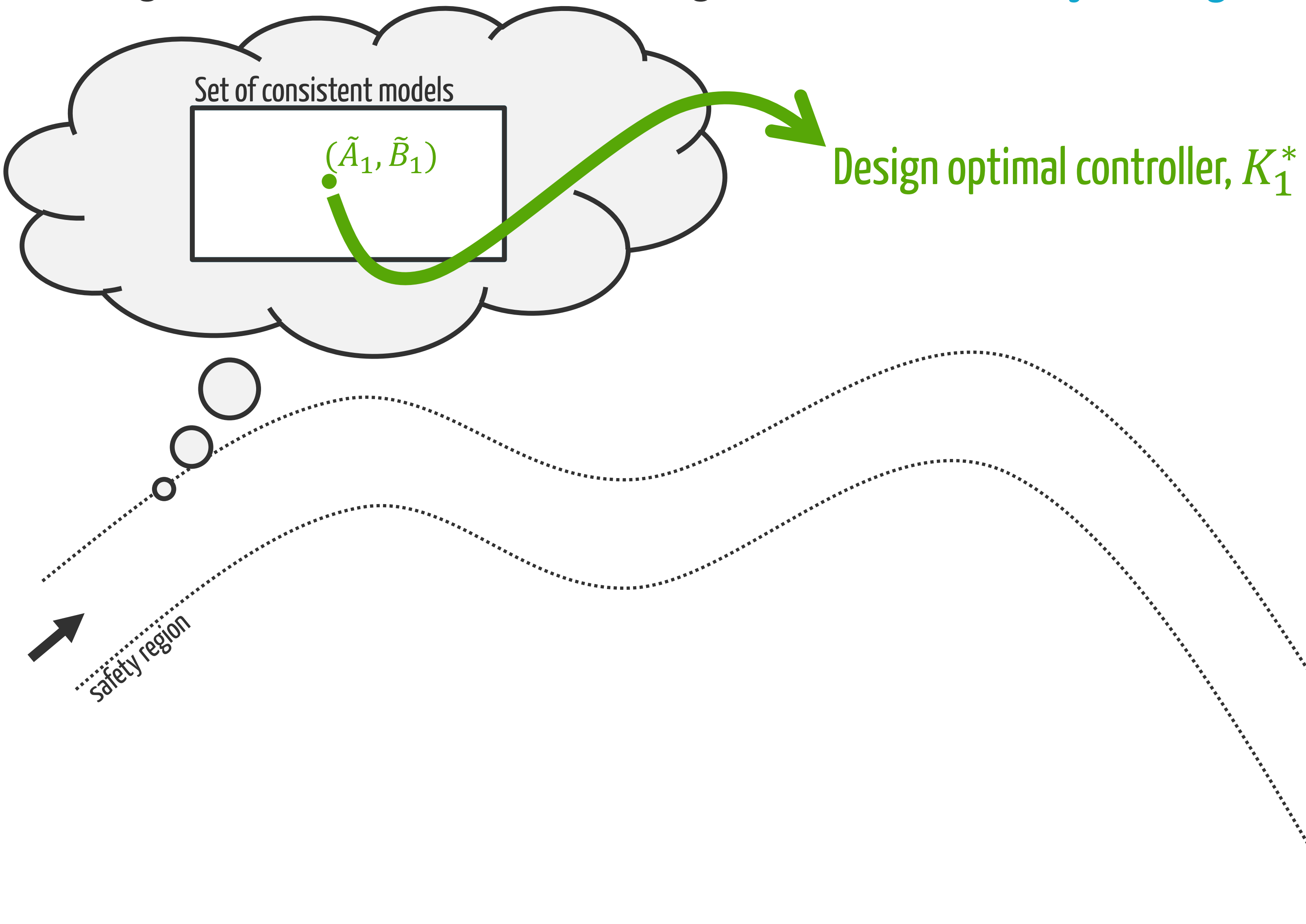


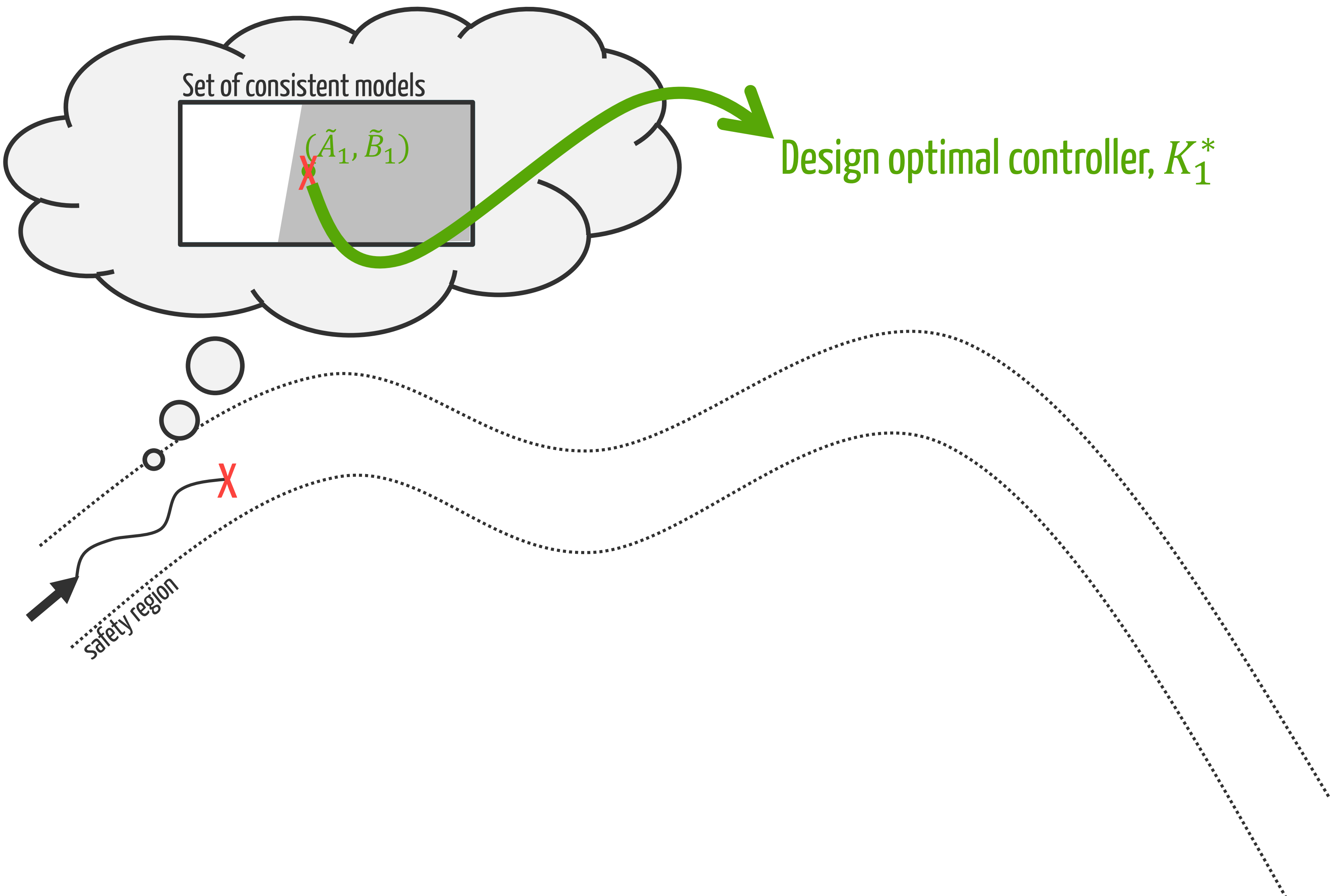

Our algorithm: Consistent Model Chasing

What if the model needs to be learned?

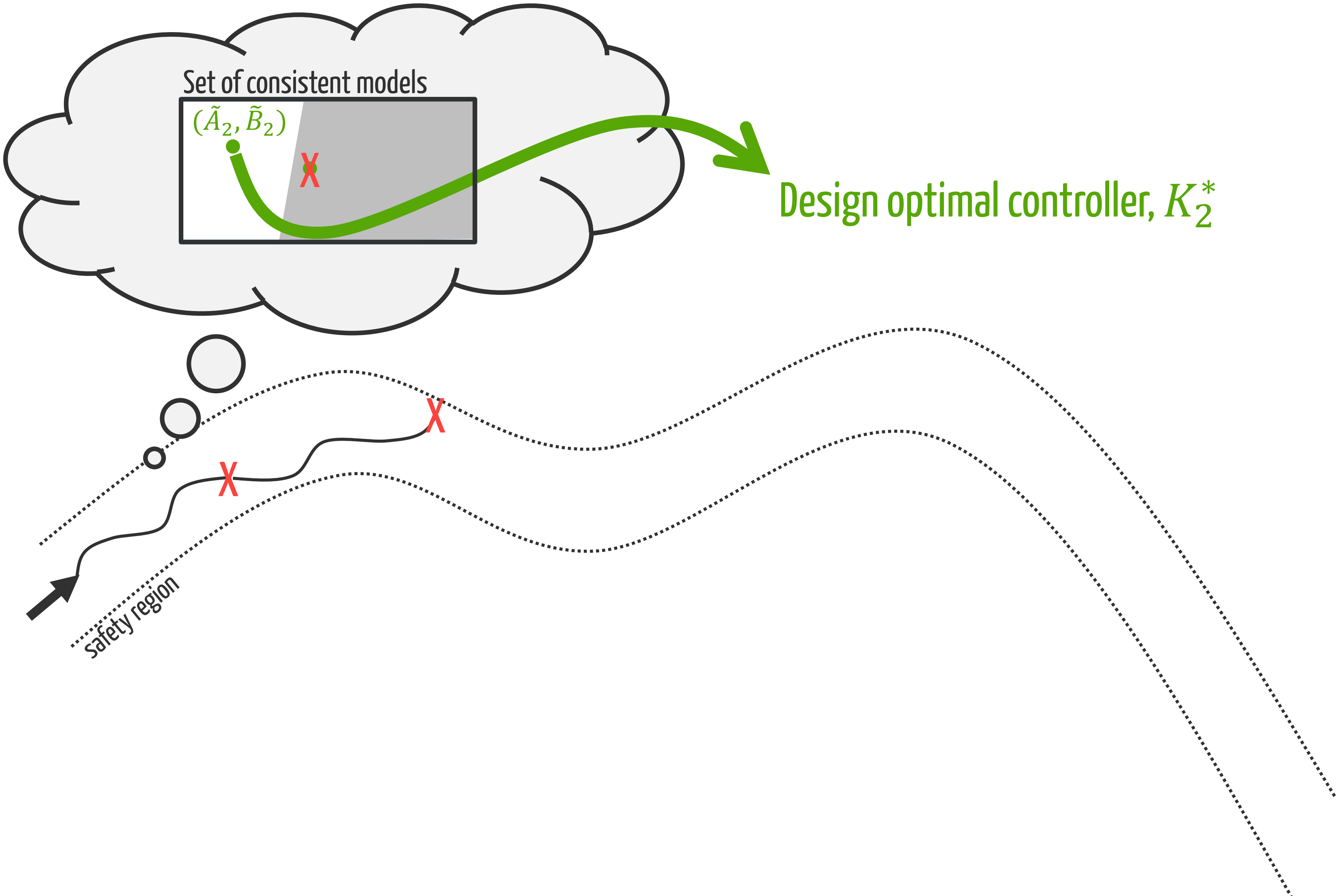


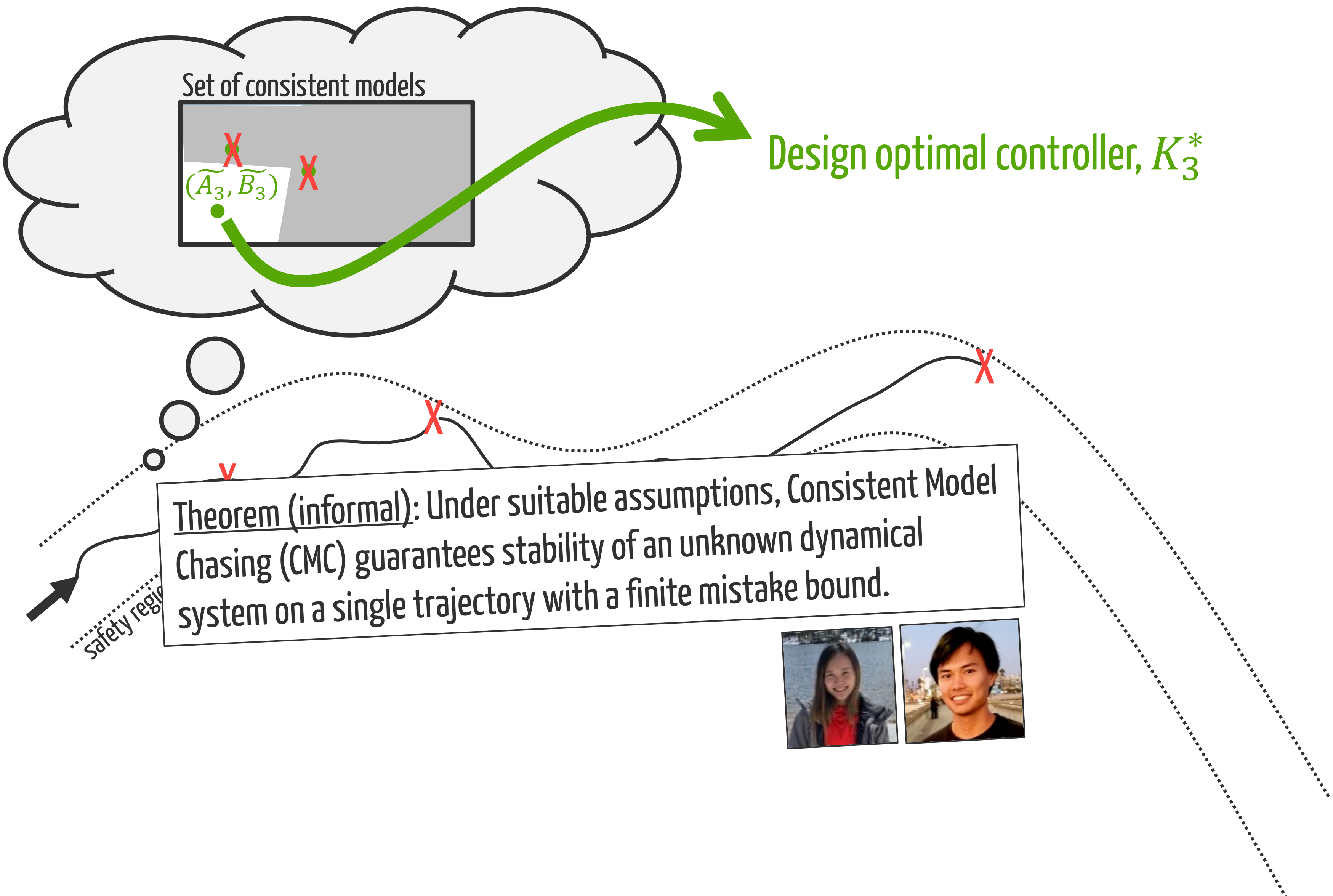
# Our Algorithm: Consistent Model Chasing via Nested Convex Body Chasing



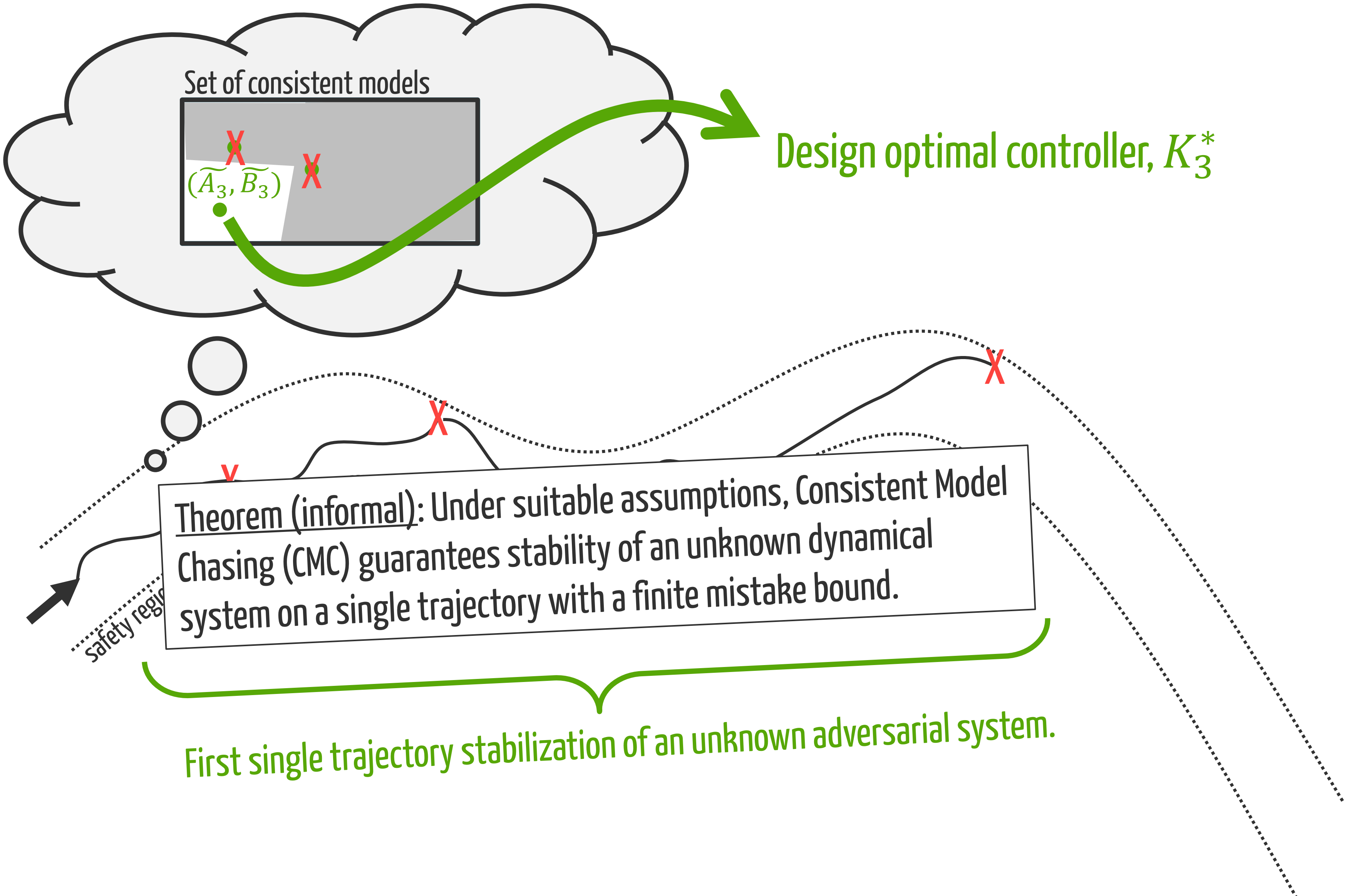


Set of consistent models
$(\tilde{A}_1, \tilde{B}_1)$
Design optimal controller, $K_1^*$
safety region


Set of consistent models
$(\tilde{A}_2, \tilde{B}_2)$
X
Design optimal controller, $K_2^*$
X
X
safety region




<u>Theorem (informal)</u>: Under suitable assumptions, Consistent Model Chasing (CMC) guarantees stability of an unknown dynamical system on a single trajectory with a finite mistake bound.



Set of consistent models
$(\widetilde{A_3}, \widetilde{B_3})$
Design optimal controller, $K_3^*$
Theorem (informal): Under suitable assumptions, Consistent Model Chasing (CMC) guarantees stability of an unknown dynamical system on a single trajectory with a finite mistake bound.
First single trajectory stabilization of an unknown adversarial system.

Holds even in networked systems with communication delay,
adversarial disturbances,
time varying models,
and distributed agents!

Theorem (informal): Under suitable assumptions, Consistent Model Chasing (CMC) guarantees stability of an unknown dynamical system on a single trajectory with a finite mistake bound.

First single trajectory stabilization of an unknown adversarial system.

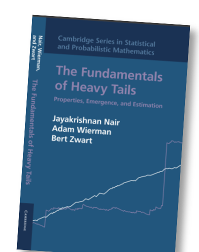
# Learning-Augmented Algorithms for Sustainable Systems

Adam Wierman, Caltech

- A Zeynali, B Sun, M Hajiesmaili, A Wierman. Data-driven Competitive Algorithms for Online Knapsack and Set Cover. AAAI 2021.
- B Sun, R Lee, M Hajiesmaili, A Wierman, D. Tsang. Pareto-Optimal Learning-Augmented Algorithms for Online Conversion Problems. NeurIPS 2021.
- T Li, R Yang, G Qu, G Shi, C Yu, A Wierman, S Low. Robustness and Consistency in Linear Quadratic Control with Untrusted Predictions. Sigmetrics 2022
- C Yeh, J Yu, Y Shi, A Wierman. Robust Online Voltage Control with an Unknown Grid Topology. E-Energy 2022.
- N Christianson, T Handina, A Wierman. Chasing Convex Bodies and Functions with Black-Box Advice. COLT 2022.
- Y Hu, G Qu, A Wierman. On the Sample Complexity of Stabilizing LTI Systems on a Single Trajectory. NeurIPS 2022.
- N Christianson, J Chen, A Wierman. Optimal Robustness-Consistency Tradeoffs for Learning-Augmented Metrical Task Systems. AIStats 2023.
- D Rutten, N Christianson, D Mukherjee, A Wierman. Online Non-convex Optimization with Untrusted Advice. Sigmetrics 2023.
- J Yu, D Ho, A Wierman. Online Stabilization of Unknown Networked Systems with Communication Constraints. Sigmetrics 2023.
- T Li, R Yang, G Qu, Y Lin, A Wierman, S Low. Certifying Black-Box Policies with Stability for Nonlinear Control. IEEE Open J of Control Sys.
- Y Lin, J Preiss, E Anand, Y Li, Y Yue, A Wierman. Online Adaptive Controller Selection in Time Varying Systems. Under submission.
- J Yang, T Li, P Li, A Wierman, S Ren. Learning for Online Competitive Control with Policy Priors. Under submission.
- P Li, J Yang, A Wierman, S Ren. Learning-Augmented Online Convex Optimization in Networks. Under submission.

**Case studies done using SustainGym**

https://chrisyeh96.github.io/sustaingym/




**New book on heavy tails!**