Reliable AI-Augmented Algorithms for Energy

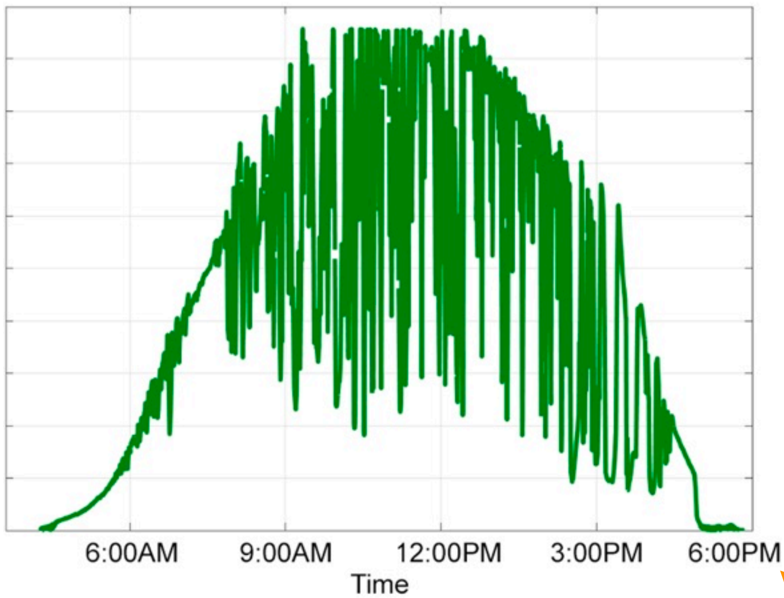
Nico Christianson

eMERGE Seminar - UC Berkeley

September 22, 2025

Uncertainty is the key challenge in energy systems

Renewable Generation



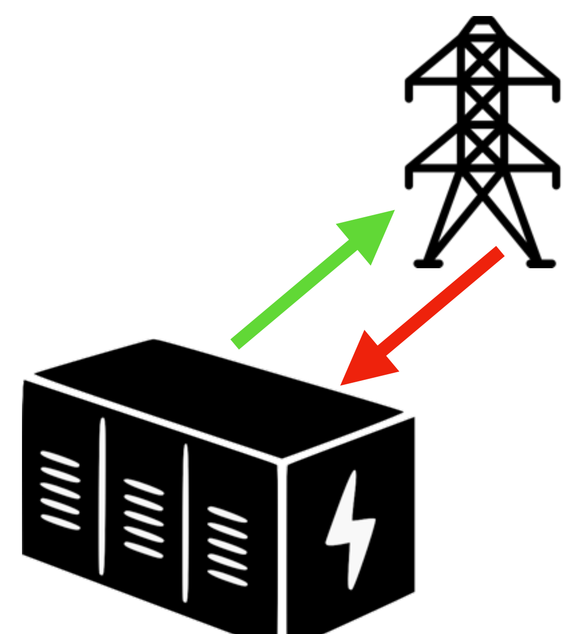
https://eprijournal.com/can-variable-solar-generation-cause-lights-to-flicker/

Consumer demand

Energy Grid/
Resource Operator

Asset outages

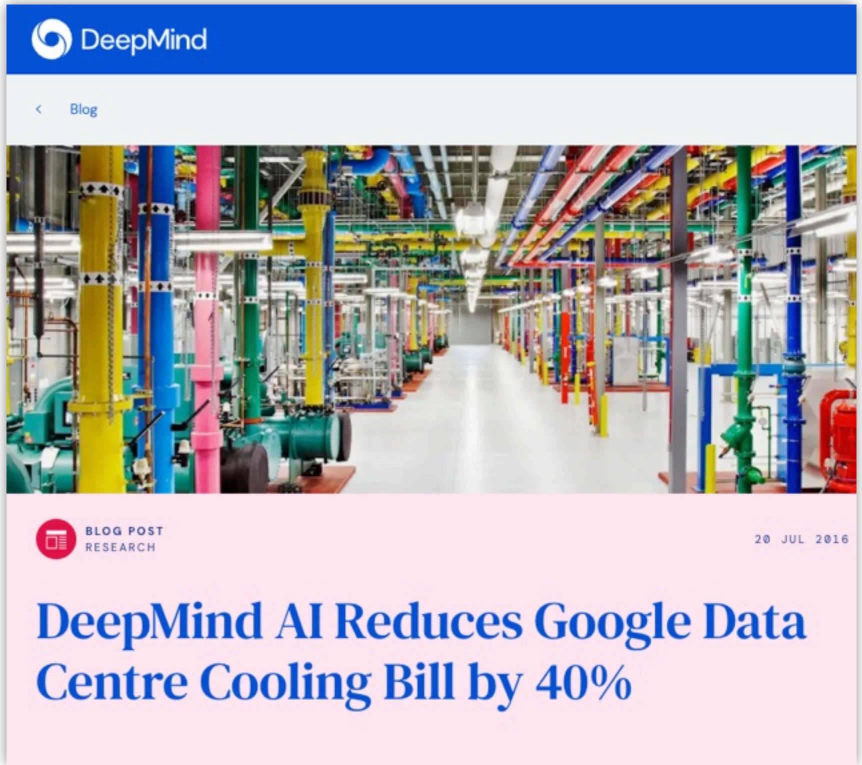
Strategic behavior



AI and machine learning could help!

State-of-the-art performance across many domains:

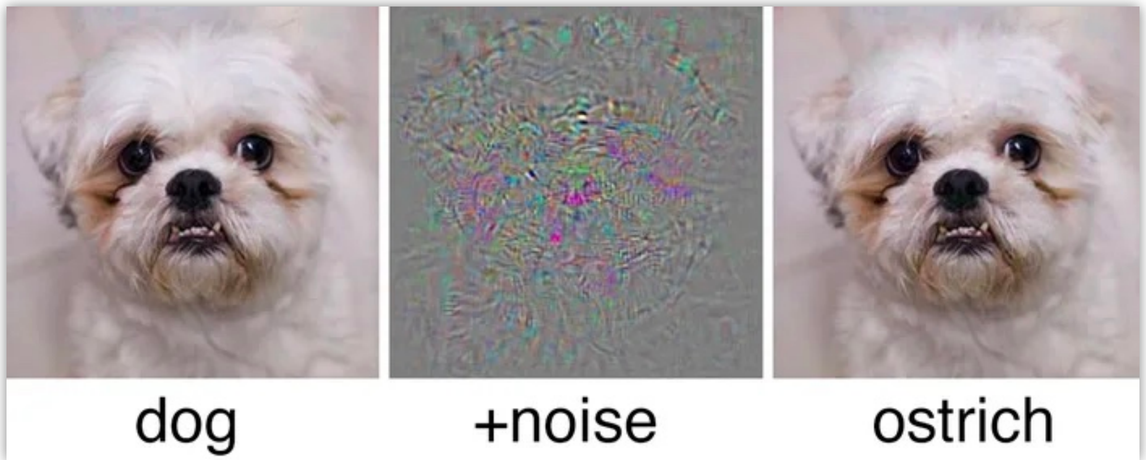
ChatGPT



…but can we trust them?

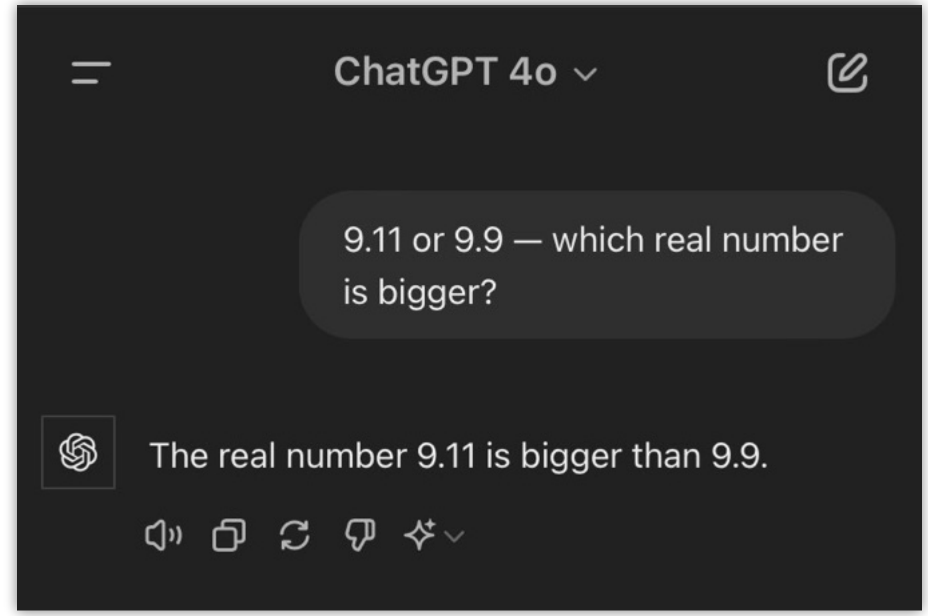
AI/ML lack of guarantees mismatched with high-stakes problems

Adversarial attacks



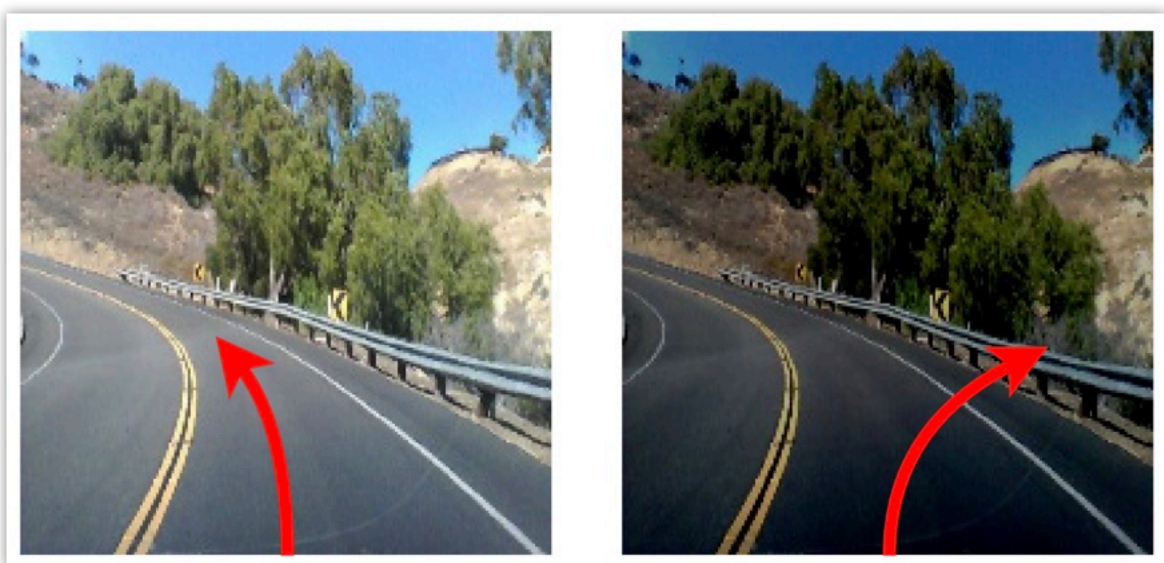
Source: Szegedy et al. 2014

Hallucination



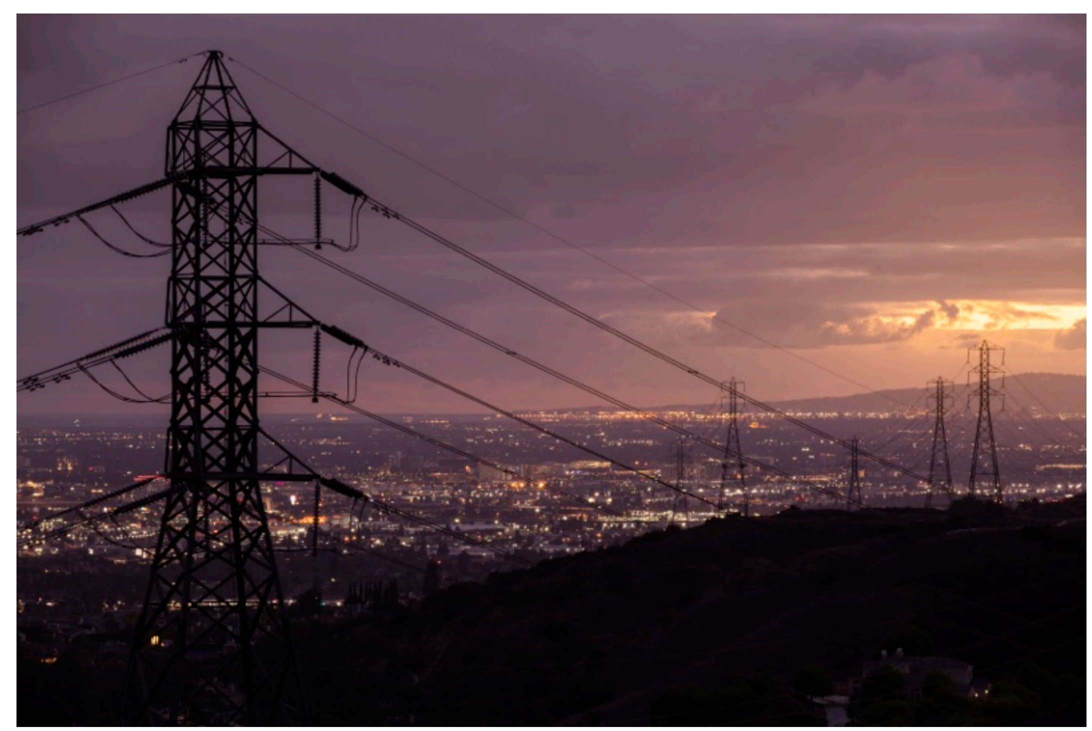
https://twitter.com/goodside/status/1812977388026011764

Distribution shift



Source: Pei et al. 2017

Significant reliability needs



How can we design ***reliable*** AI/ML methods to advance sustainable energy systems?

Robert Giroux / Getty

Overview of my research

Theory ⟷ **Applications & Collaborations**

My papers:

Online/sequential decision-making

Machine Learning

Overview of my research

Theory

Online/sequential decision-making

Machine Learning

Optimization

Energy/sustainability focus

My papers:

C*W*WL	(PSCC ’22)
CSLW	(COLT ’24)
LCZ+	(SIGMETRICS ’24)
CHW	(COLT ’22)
RCMW	(SIGMETRICS ’23)
CSW	(AISTATS ’23)
LCS+	(SIGMETRICS ’24)
LCS+	(ICML ’24)
CLC+	(SIGCOMM ’24)
LCS+	(SIGMETRICS ’25)
SHC+	(ICML ’24)
CYL+	(EDS ‘25)
YLDAC+	(NeurIPS ’23)
Y*C*W+	(under review)
CCL+	(L4DC ’25)
W*C*Z+	(CDC ’23)

*equal contribution

Applications & Collaborations

Carbon-aware computing

University of Massachusetts Amherst

Energy resource operation

BEYOND LIMITS

Video streaming

amazon prime video

Gridmatic

Grid-scale energy storage

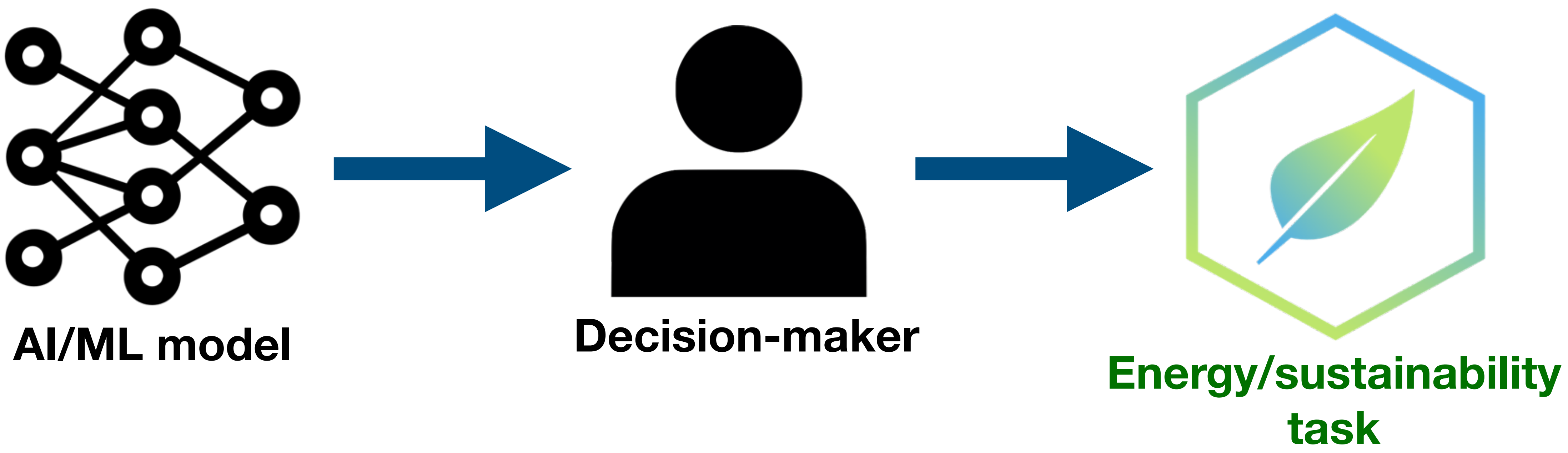
Microsoft

Energy systems

Research theme:

How can we use AI/ML for **reliable** decision-making in energy?

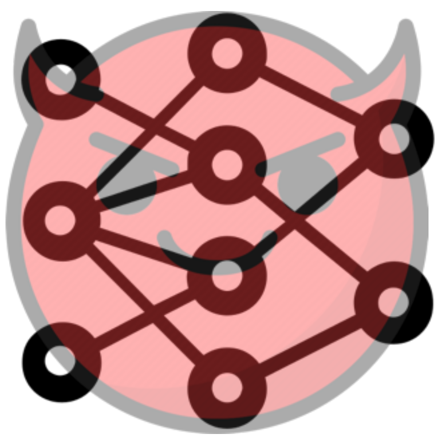
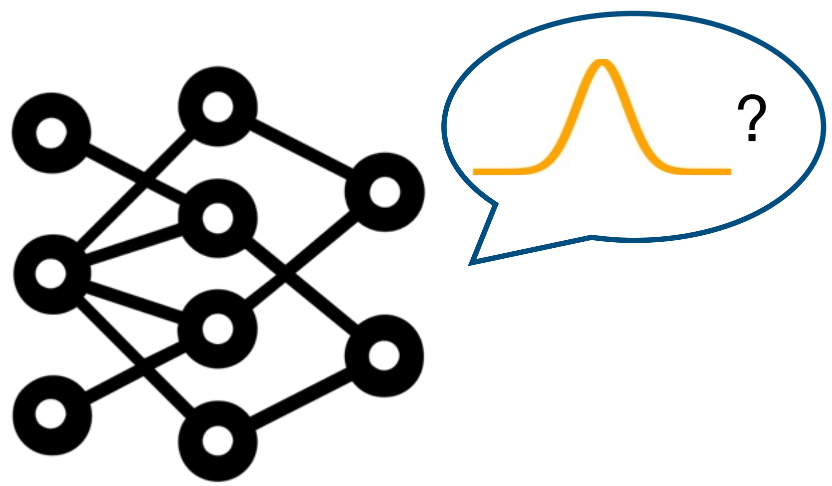
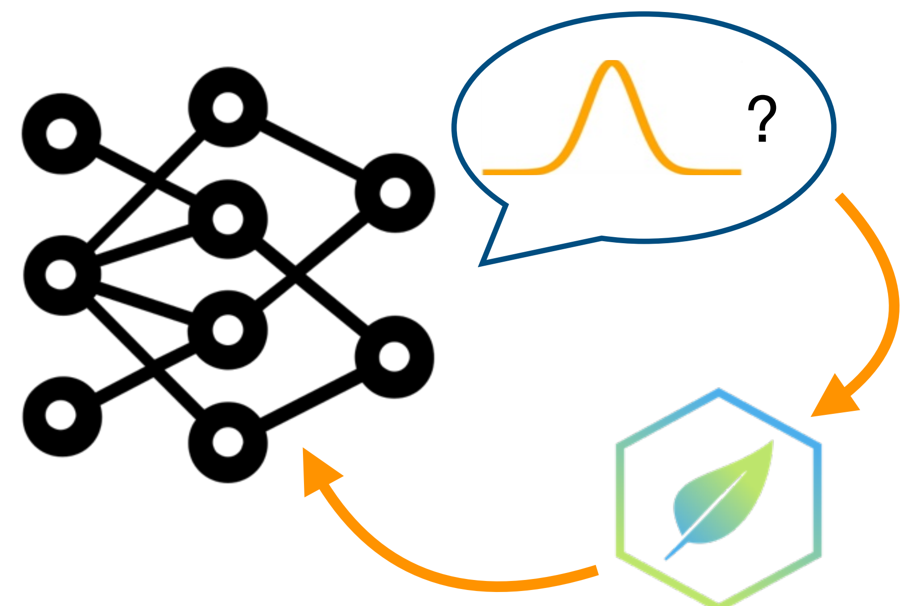
Setting:



This talk:

How can we use AI/ML for **reliable** decision-making in energy, given varying strengths of *a priori* performance guarantees?

More control over guarantees

"Black-box"	"Grey-box"	"End-to-End"
		
No guarantees on model performance	Model provides *some* uncertainty estimates	Trained for specific task with reliability (e.g., UQ) in-the-loop

"Black-box"

Online optimization (COLT '22, SIGMETRICS '23, AISTATS '23)

Online optimization w/ long-term constraints (ICML '24, SIGMETRICS '24 + '25)

⋮

"Grey-box"

Pricing uncertainty in electricity markets (CDC '23)

Algorithms w/ UQ predictions (ICML '24)

Risk-sensitive online algorithms (COLT '24)

⋮

"End-to-End"

Reliable ML for contingency screening (L4DC '25)

End-to-End UQ (under review)

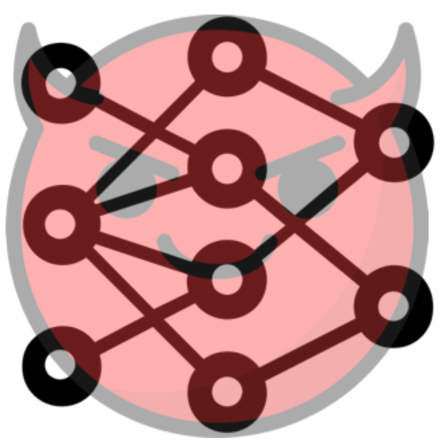
End-to-End risk control (NeurIPS '25)

This talk:

How can we leverage black-box AI/ML while preserving worst-case robustness guarantees?

More control over guarantees

“Black-box”



No guarantees on model performance

“Grey-box”

Model provides *some* uncertainty estimates

“End-to-End”

Trained for specific task with reliability (e.g., UQ) in-the-loop

First half of today’s talk

Motivation: Optimizing cogeneration plant operation

Collaboration with BEYOND LIMITS

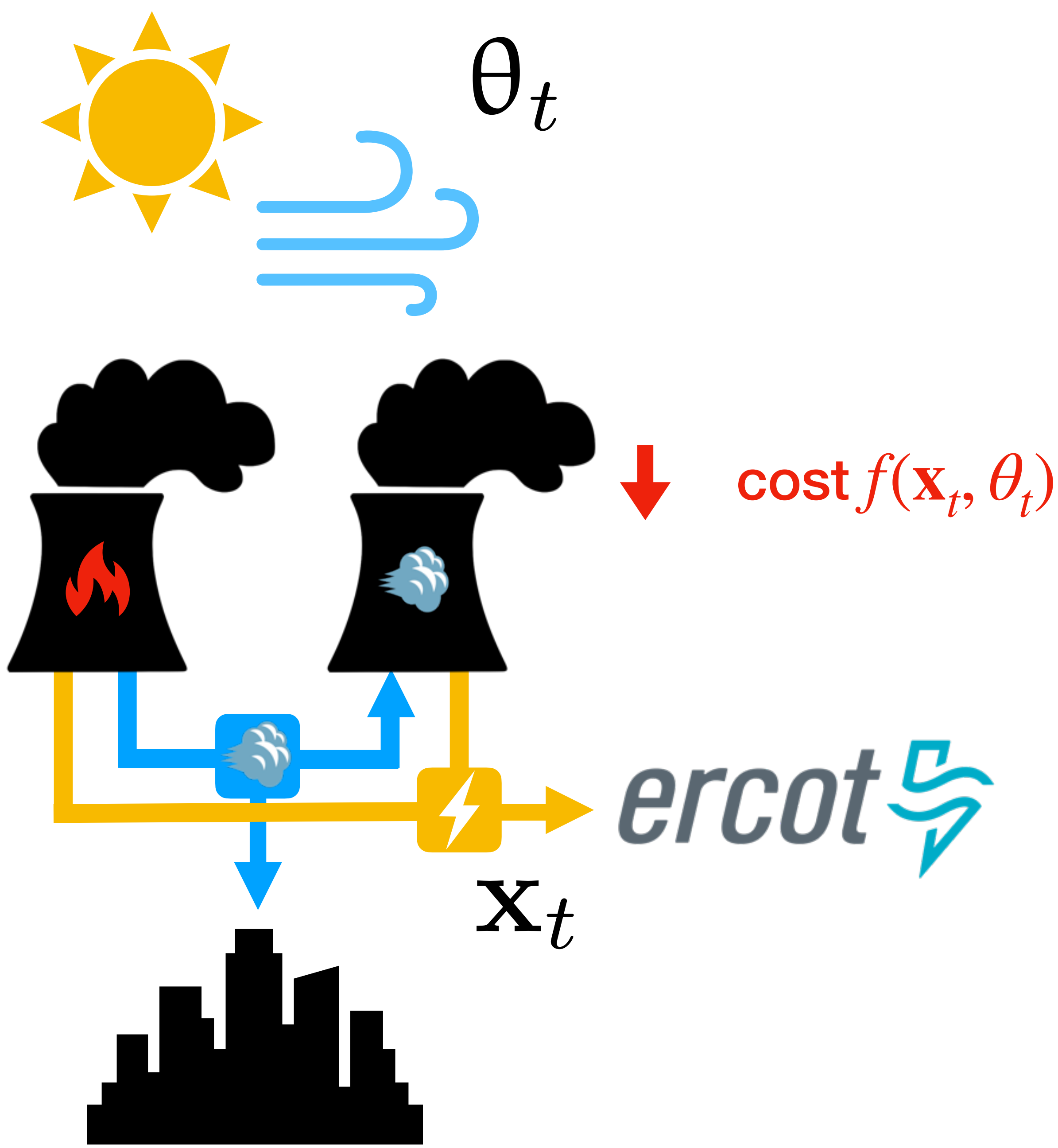
Combined-cycle cogeneration power plant

Uncertainty θ_t

- Ambient conditions
- Electricity demand
- Municipal steam demand

Decisions $\mathbf{x}_t$

- Gas (x3) and steam turbine generation
- Steam production and diversion to steam turbine



Motivation: Optimizing cogeneration plant operation

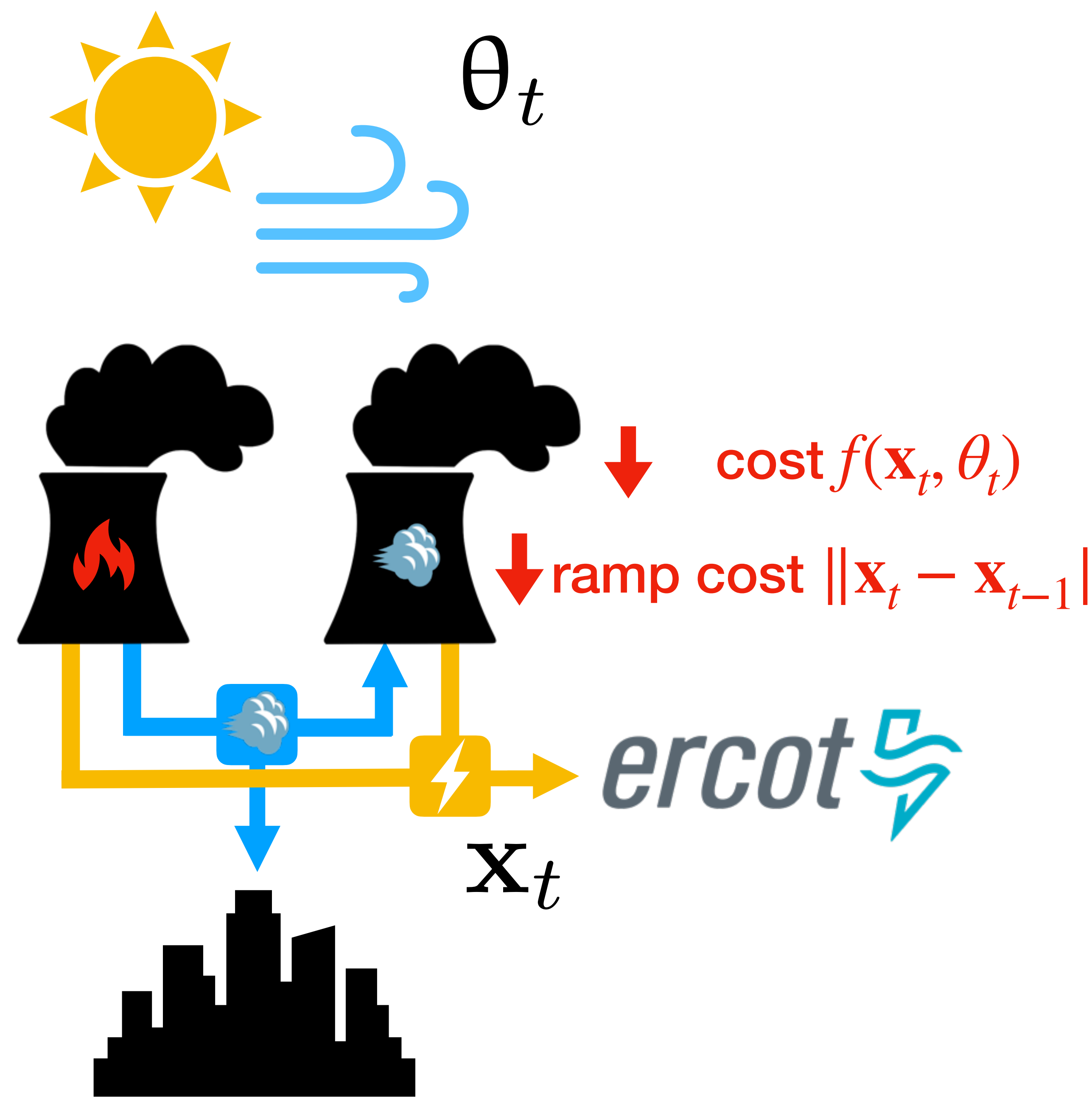
Operation in the high-renewables regime

- Increasing renewables requires more frequent ramping - inefficient
- Common approach is Model Predictive Control (MPC):

$$\mathbf{x}_t \leftarrow \min_{\mathbf{x}} f_t(\mathbf{x}_t, \theta_t) + \sum_{\tau=t+1}^{t+h} f_\tau(\mathbf{x}_\tau, \hat{\theta}_\tau) + \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|$$

Current cost

Forecast



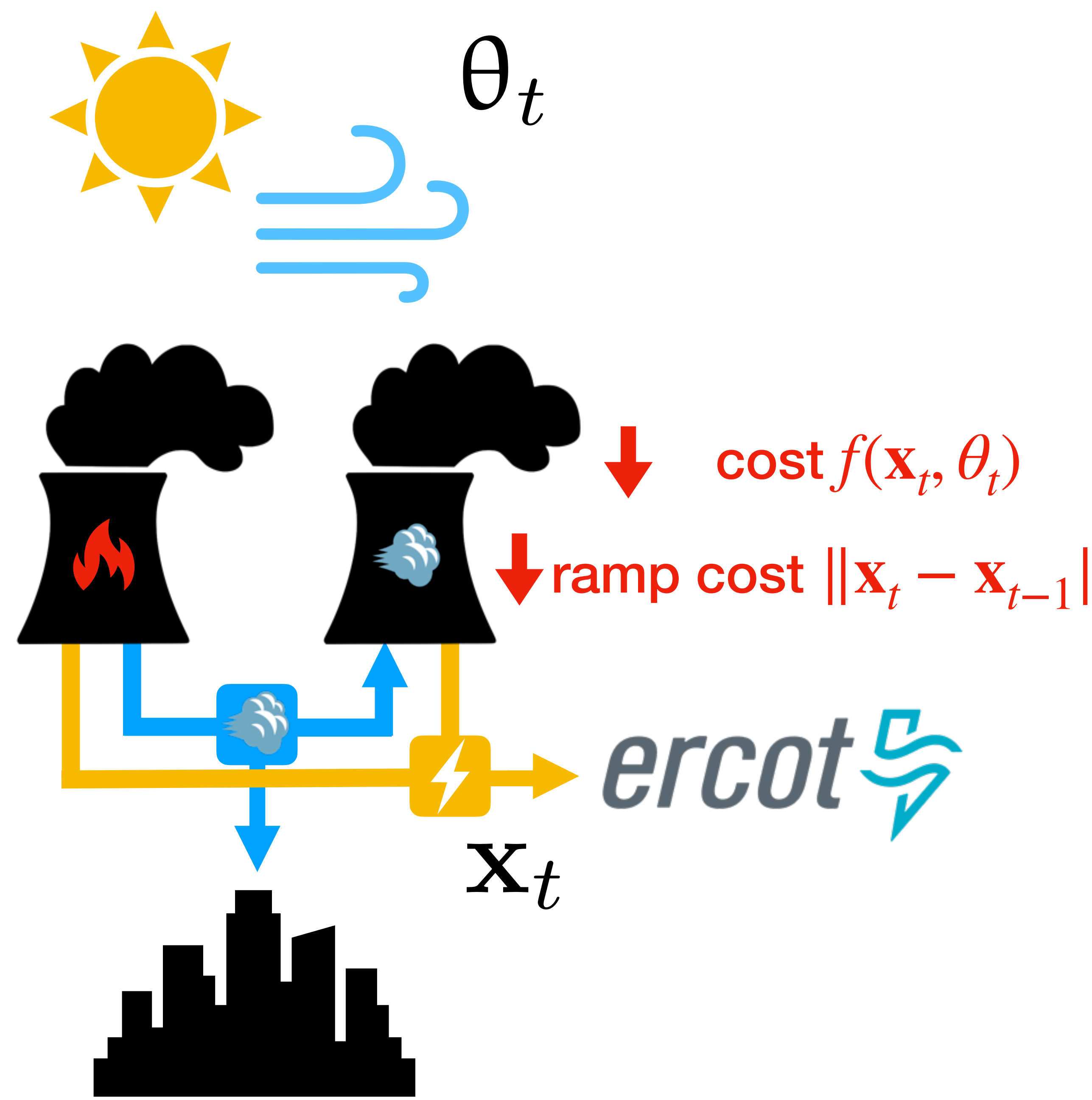
Motivation: Optimizing cogeneration plant operation

Challenge:

- Complex, nonconvex cost $f(\mathbf{x}_t, \theta_t)$
- Intractable to solve MPC with even moderate lookahead h

Beyond Limits was using heuristics - slow, poor quality!

Wanted to use AI/ML to unlock better performance - but needed worst-case guarantees!



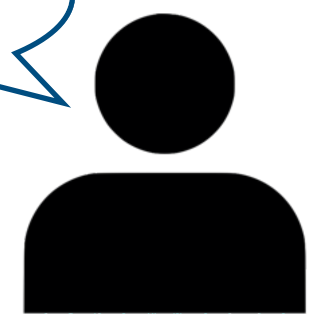
Model: Metrical Task Systems (MTS)

Adversary

$f_t : M \to \mathbb{R}_+$

Metric space (M, d)

$x_t \in M$



Decision-maker

hitting switching

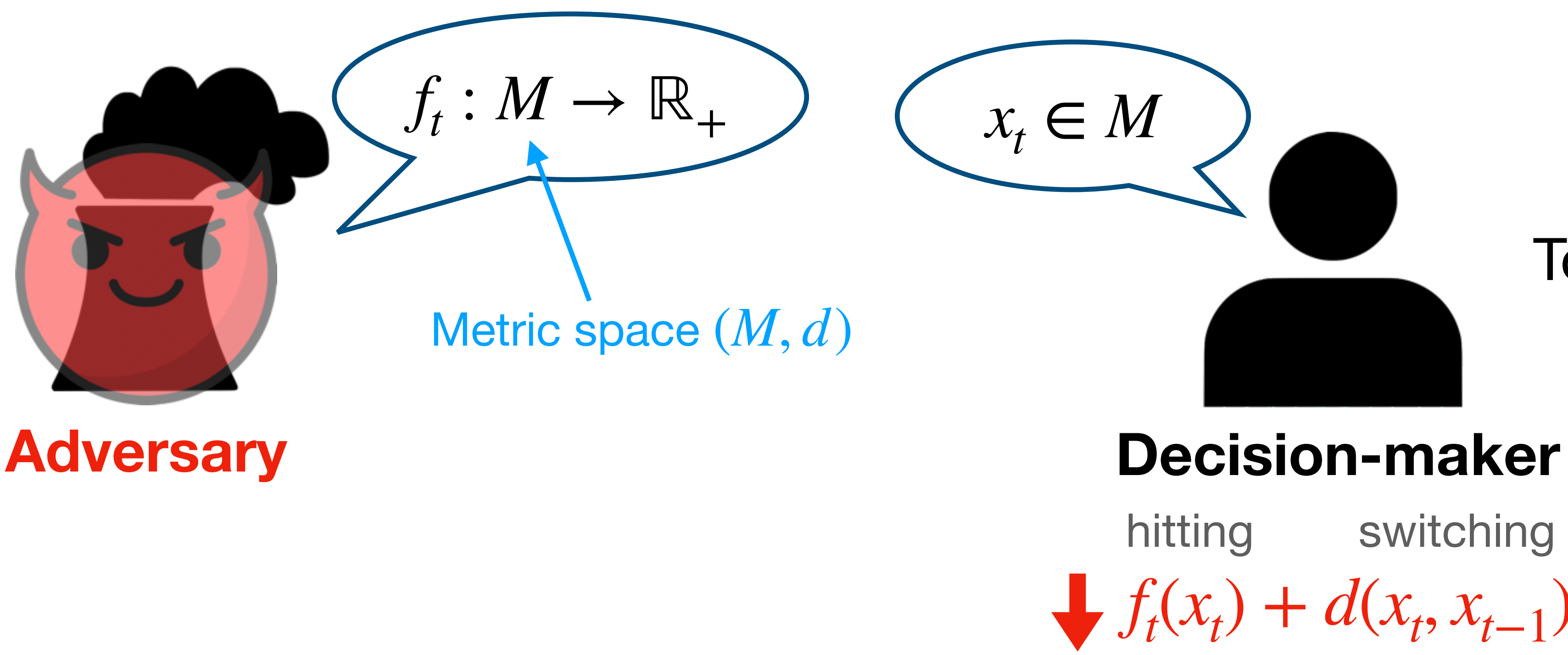
$\downarrow f_t(x_t) + d(x_t, x_{t-1})$

Total cost: $\sum_t f_t(x_t) + d(x_t, x_{t-1})$

Also known as:

- Online optimization with switching costs
- "Smoothed" online optimization
- Function chasing

Model: Metrical Task Systems (MTS)



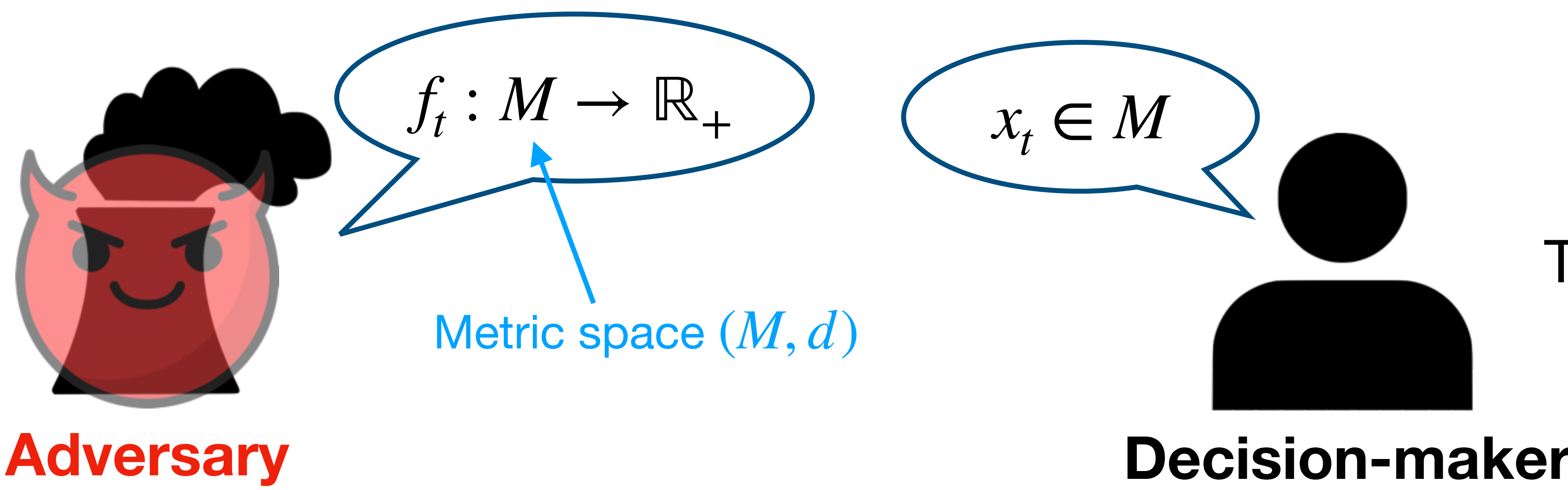
Total cost: $\sum_t f_t(x_t) + d(x_t, x_{t-1})$

Also models other problems like

- datacenter operation[1]
- logistics[2]
- video streaming[3]

1. Lin et al. ’11, ’12; Albers and Quedenfeld ’18, ’21
2. Dehghani et al. ‘17
3. Chen, Lin, **Christianson**, et al. ’24

Model: Metrical Task Systems (MTS)

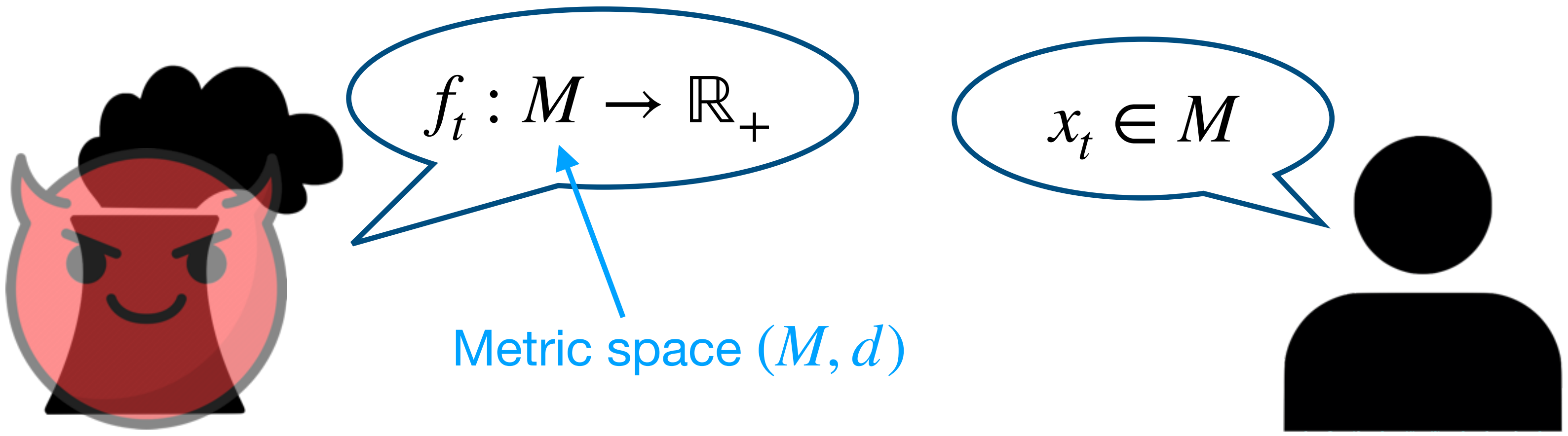


Total cost: $\sum_t f_t(x_t) + d(x_t, x_{t-1})$

Typically want algorithms with small ***competitive ratio:***

$$\mathrm{CR} = \sup_{\{f_t\}} \frac{\mathbb{E}[\mathrm{Cost}(\mathtt{ALG})]}{\mathrm{Cost}(\mathtt{OPT})} = \sup_{\{f_t\}} \frac{\mathbb{E}\left[\sum_t f_t(x_t) + d(x_t, x_{t-1})\right]}{\min_{\{o_t\}} \sum_t f_t(o_t) + d(o_t, o_{t-1})}$$

Model: Metrical Task Systems (MTS)



Adversary

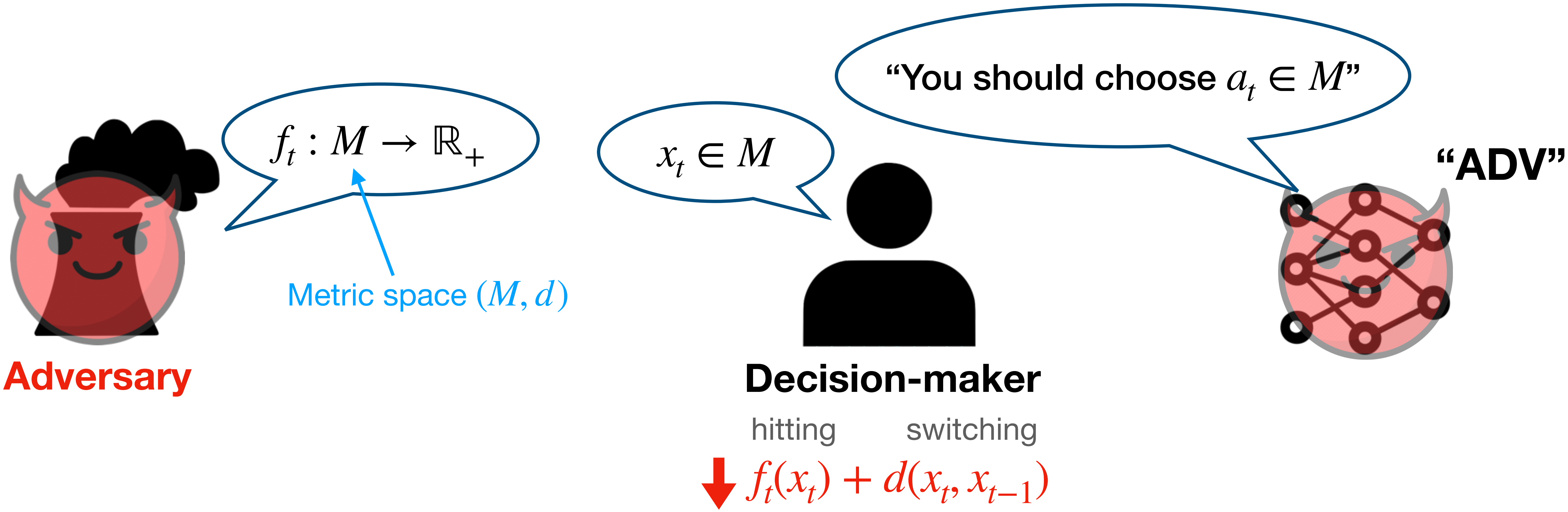
Decision-maker

Total cost: $\sum_t f_t(x_t) + d(x_t, x_{t-1})$

	State-of-the-art competitive ratio	
General n-point metrics	$2n - 1$ (det.)	$\Theta(\log^2 n)$ (rand.)
Convex function chasing ($M = \mathbb{R}^n, f_t$ convex)	$\Theta(n)$	

+ many special cases (specific metrics, function classes, …)

Metrical task systems with AI/ML advice

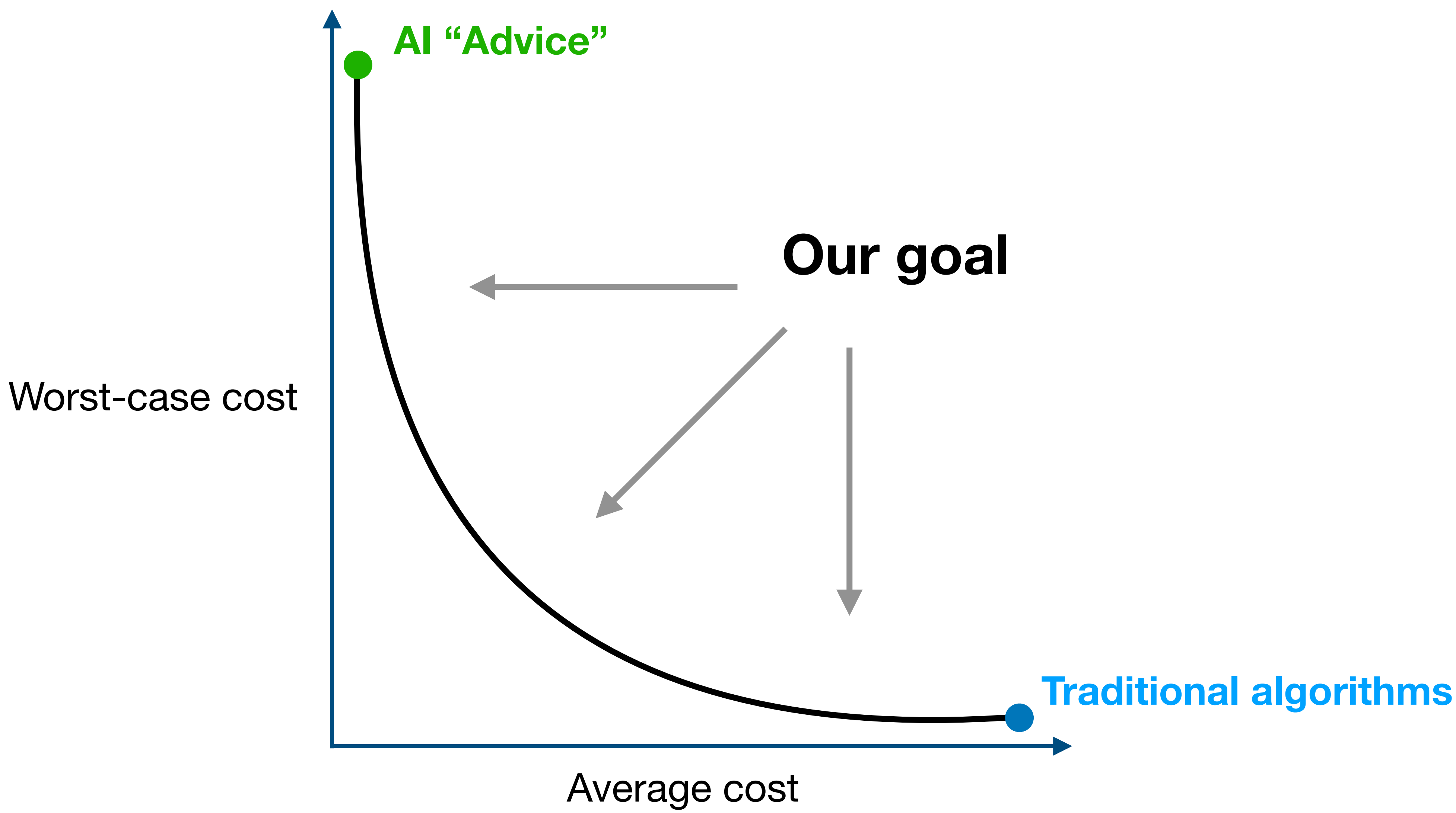


Traditional algorithms for MTS are pessimistic

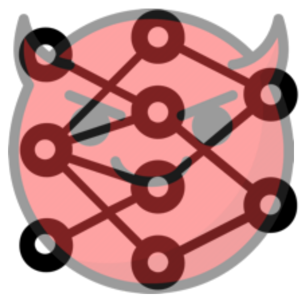
AI/ML can often do better!

Black-box AI - model as possibly adversarial

Goal: Bridge AI/ML and worst-case performance



Our approach:



Black-box advice "ADV"

$a_1, \ldots, a_T \in \mathbb{R}^d$

Robust algorithm "ROB"

$r_1, \ldots, r_T \in \mathbb{R}^d$

Meta-algorithm

Efficient and robust decisions

$x_1, \ldots, x_T \in \mathbb{R}^d$

How should our algorithm behave…

when advice performs well?

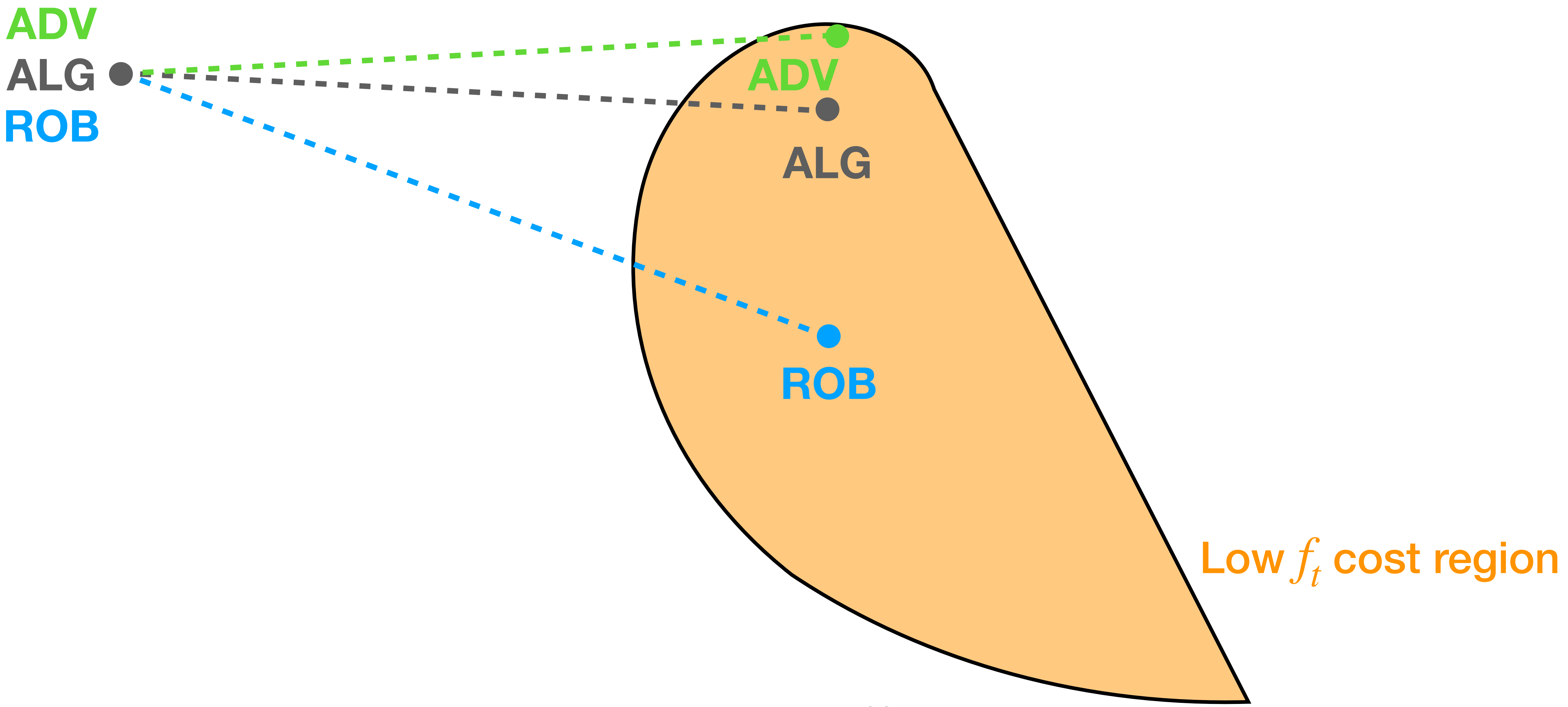
ADV

ALG ●

ROB

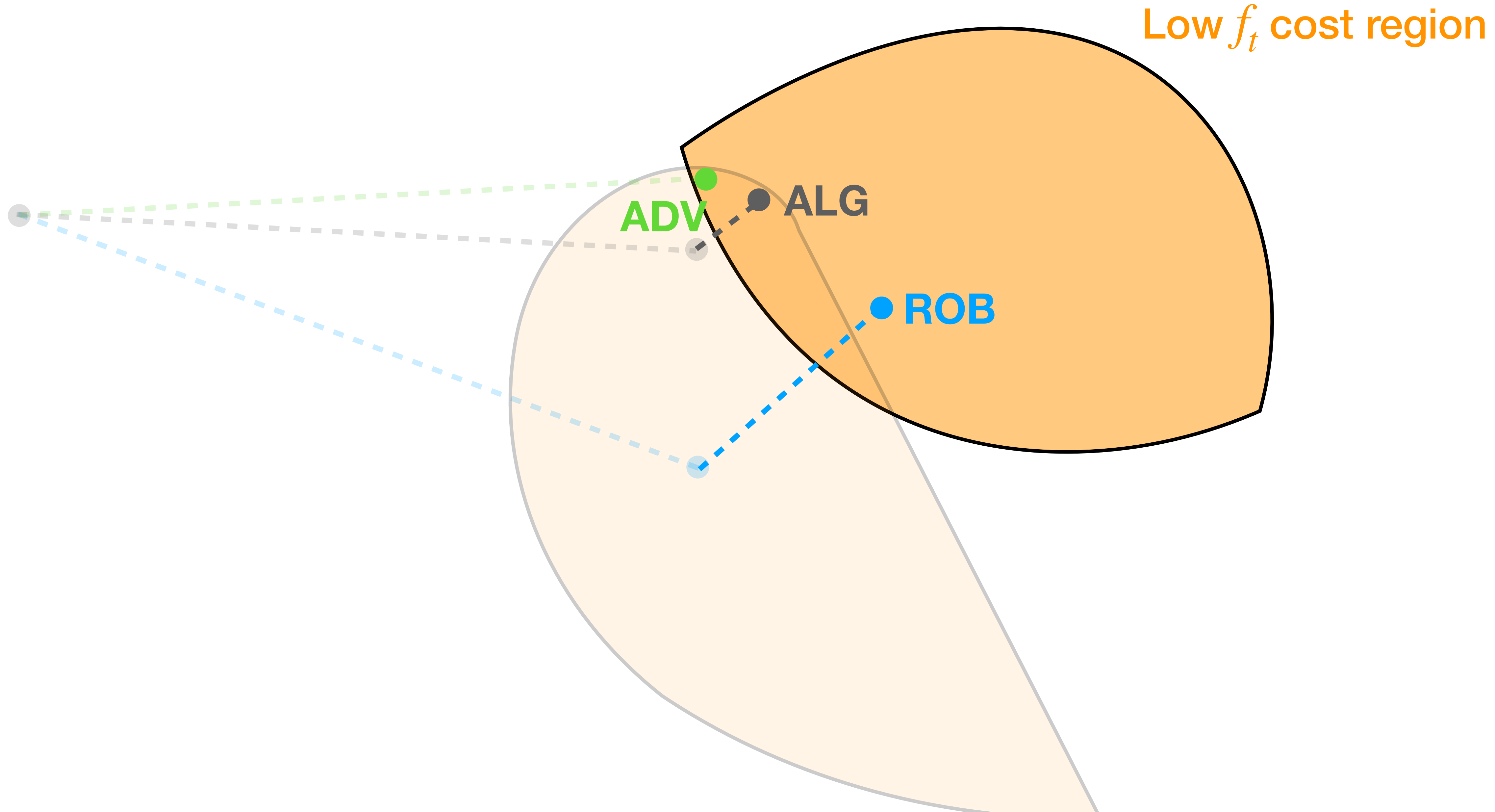
How should our algorithm behave...

when advice performs well?



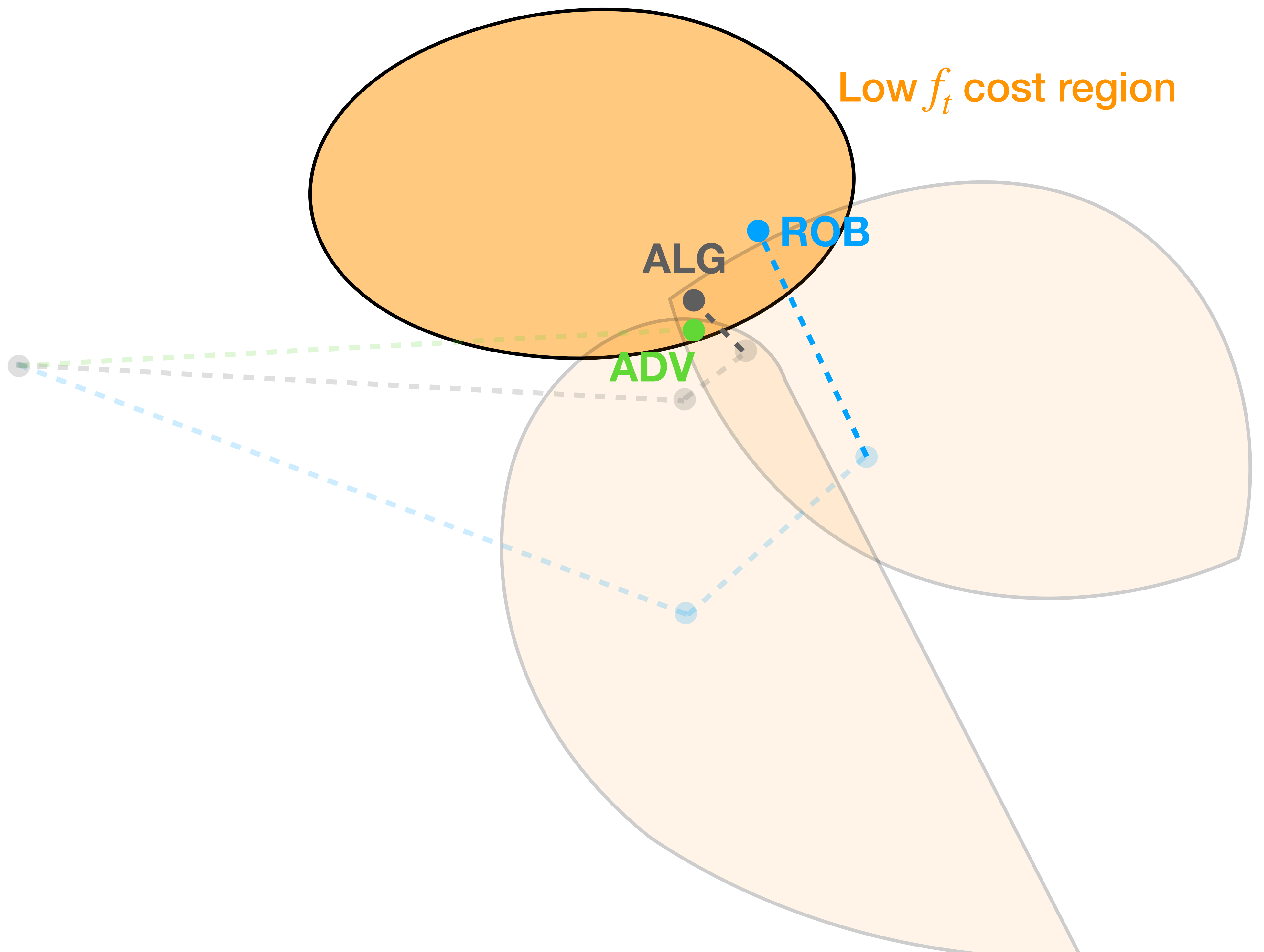
How should our algorithm behave…

when advice performs well?



How should our algorithm behave…

when advice performs well?



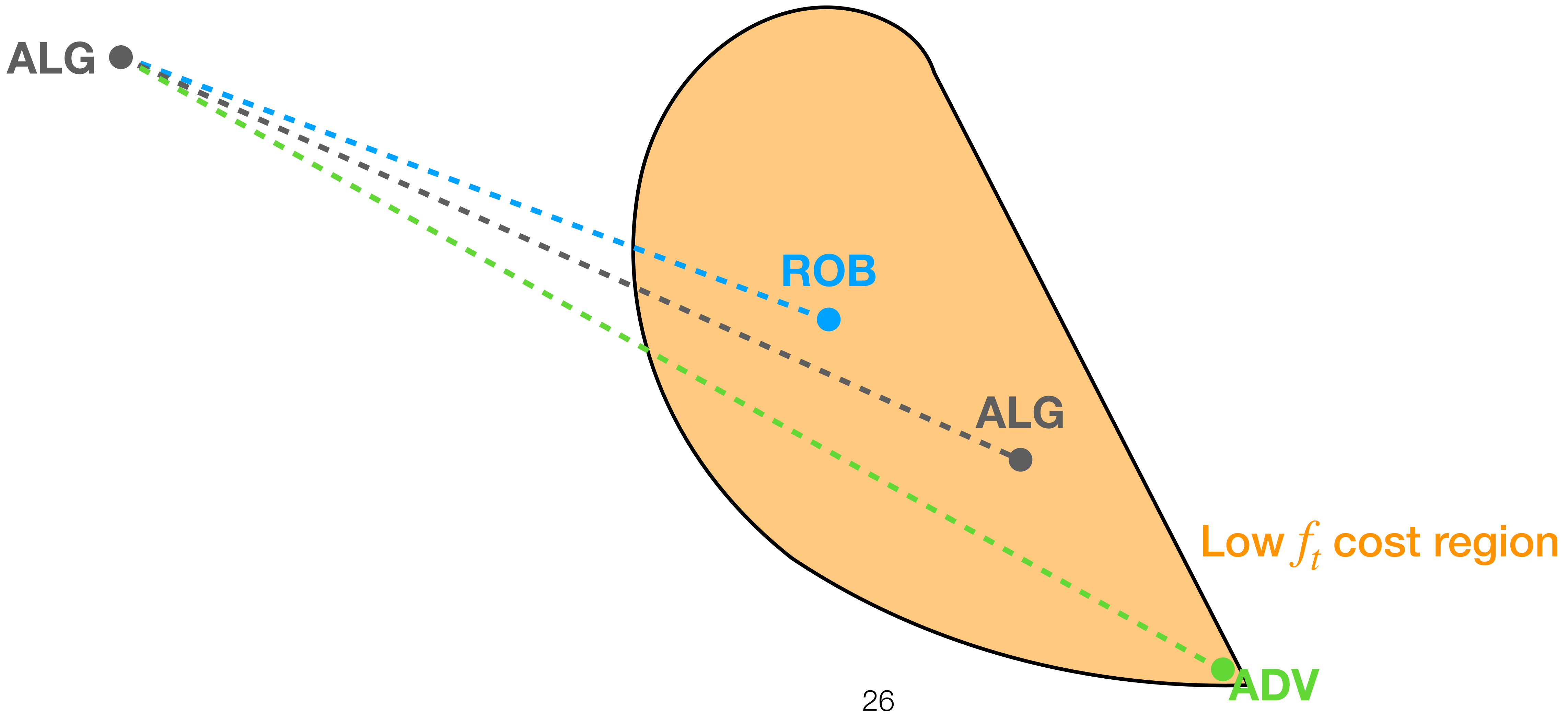
How should our algorithm behave…

when advice performs poorly?

ADV
ALG ●
ROB

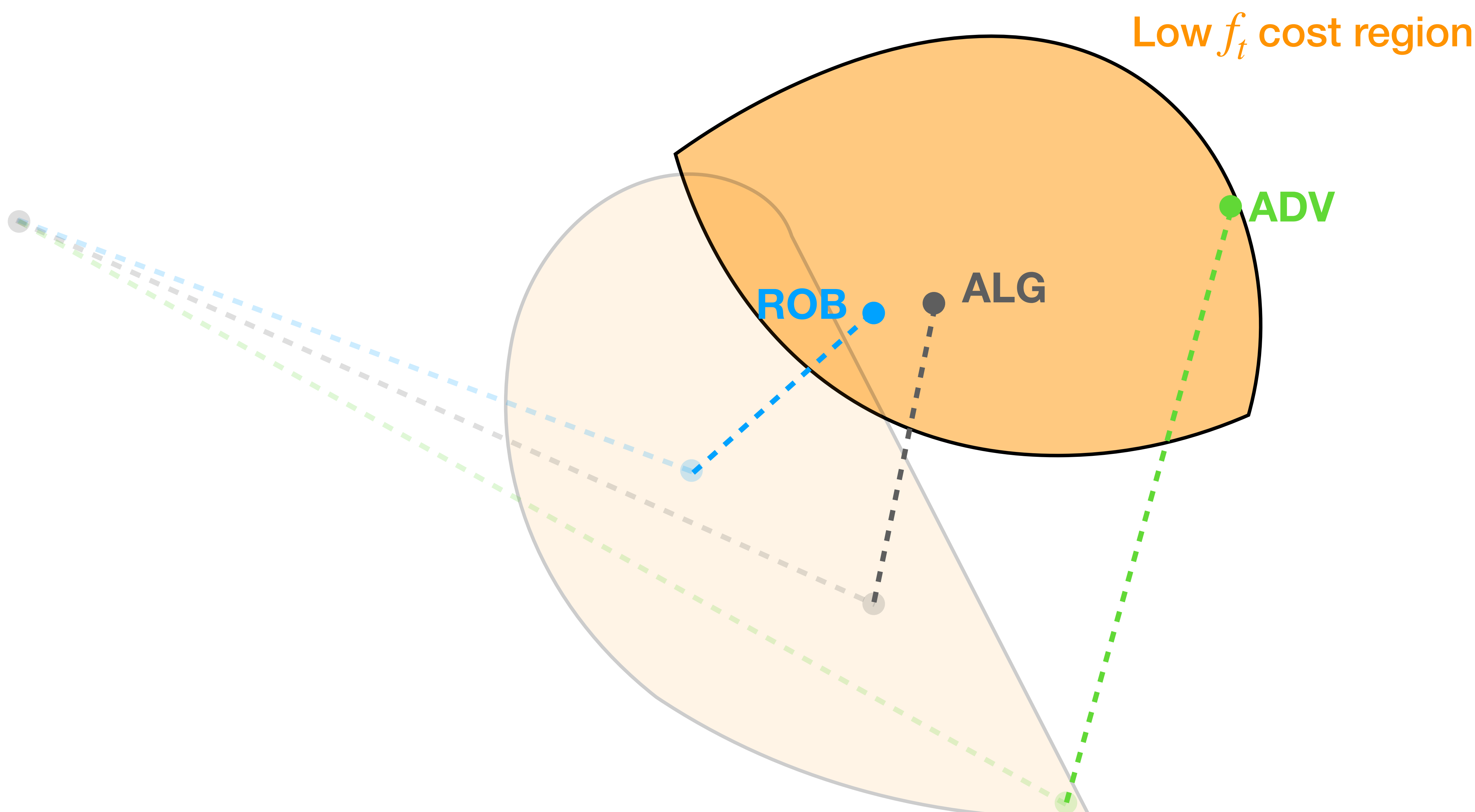
How should our algorithm behave…

when advice performs poorly?



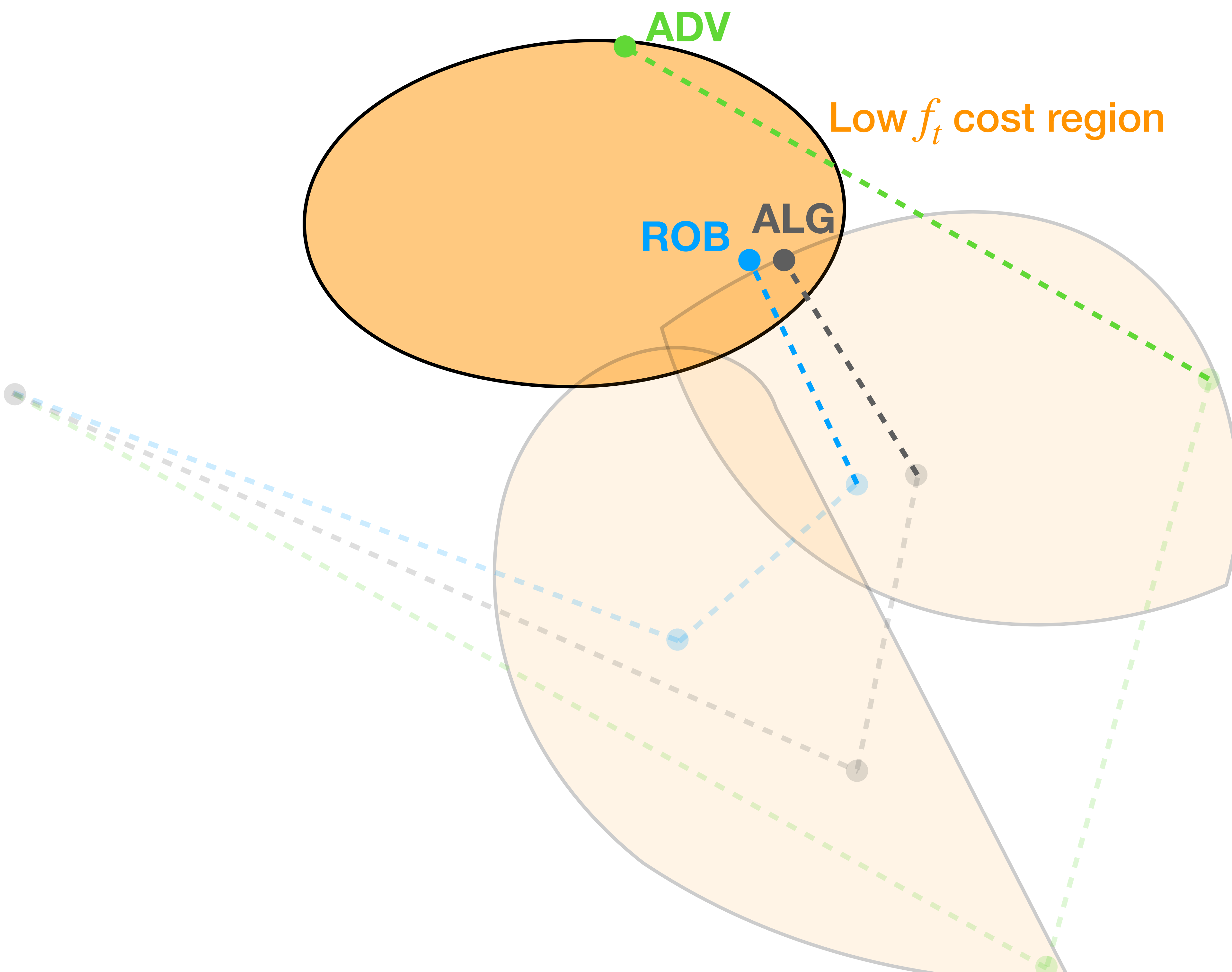
How should our algorithm behave…

when advice performs poorly?



How should our algorithm behave…

when advice performs poorly?



We want a meta-algorithm that is…

Black-box AI "advice"

Consistent: $\mathrm{Cost}(\mathtt{ALG}) \leq (1+\epsilon) \cdot \mathrm{Cost}(\textbf{ADV})$

Hyperparameter $\epsilon > 0$

Robust: $\mathrm{Cost}(\mathtt{ALG}) \leq C(\epsilon) \cdot \mathrm{Cost}(\textbf{ROB})$

Traditional robust algorithm

Significant recent interest designing algorithms with AI/ML advice
- "**algorithms with predictions**" - for online (+ other) problems:

- Caching (Lykouris and Vassilvitskii '18)
- Mechanism design (Agrawal et al. '22, Balkanski et al. '24)
- Linear quadratic control (Li et al. '21)
- **250+ more: https://algorithms-with-predictions.github.io/**

Prior work

Mostly special cases: Lykouris & Vassilvitskii '18; Lindermayr et al. '22; **Christianson** et al. '22; Rutten, **Christianson**, et al. '23

Antoniadis et al. (ICML 2020): off-the-shelf approaches

Doubling idea

- Deterministic 9-consistent, 9-robust algorithm
- Randomized algorithm achieving

Can't take advantage of good AI advice

$$\text{Cost}(\texttt{ALG}) \leq (1+\epsilon)\min\{\text{Cost}(\textbf{ADV}), \text{Cost}(\textbf{ROB})\} + \mathcal{O}(D/\epsilon)$$

Multiplicative weights

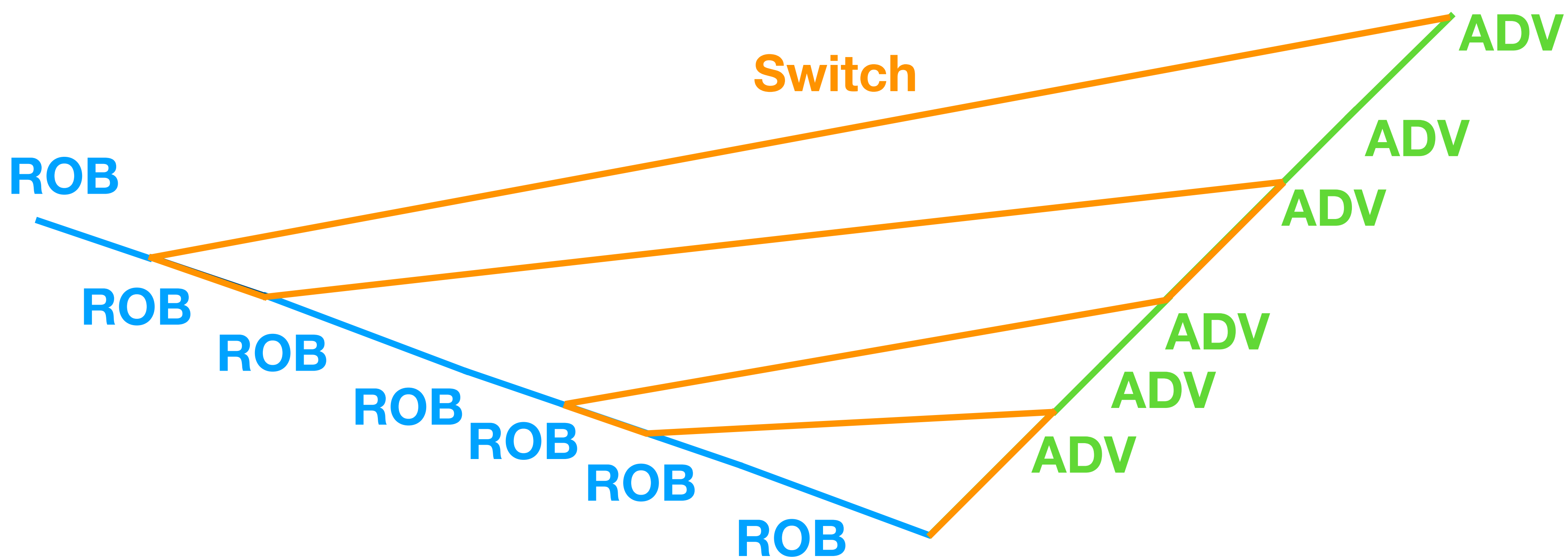
Large additive term on both consistency + robustness

Can we exploit problem structure to do better?

Initial idea: Deterministic switching algorithms

Deterministically switch between ADV and ROB decisions

Can this idea yield good consistency-robustness tradeoffs?



Initial idea: Deterministic switching algorithms

Deterministically switch between ADV and ROB decisions

Can this idea yield good consistency-robustness tradeoffs? **No!**

Theorem. Any deterministic switching algorithm with **finite** robustness has consistency ≥ 3.

Christianson, Handina, and Wierman, COLT '22

*We can do better (but not optimally!) with certain assumptions
- e.g., structural assumptions on f_t or *a priori* diameter bound

Toward the optimal robustness-consistency tradeoff: Randomized algorithms

Choose $x_t \sim p_t$ supported on $\{a_t, r_t\}$

ADV decision

ROB decision

Our idea:

- Set consistency "budget" $(1+\epsilon)$
- Choose p_t to be maximally robust while remaining consistent

$$\mathbb{E}_{x_1 \sim p_1, \ldots, x_t \sim p_t}\left[\sum_{\tau=1}^{t} f_\tau(x_\tau) + d(x_\tau, x_{\tau-1}) + d(a_t, x_t)\right] \leq (1+\epsilon)\,\mathrm{Cost}_{1:t}(\textbf{ADV})$$

How should our algorithm behave?

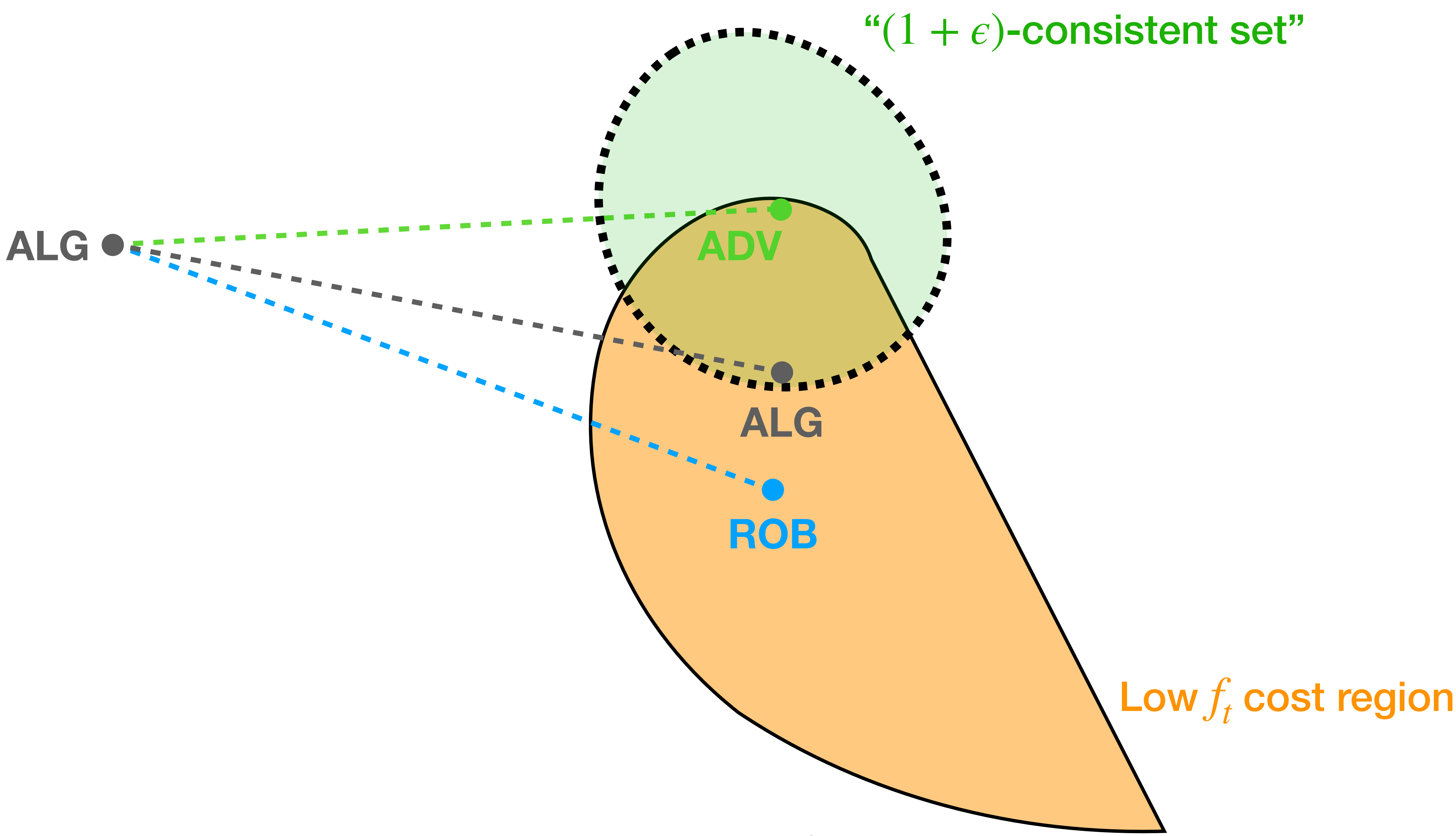
ADV

ALG ●

ROB

How should our algorithm behave?

How should we select p_t to be consistent and robust?



The DART Algorithm

DART = “Distance Adaptive Robust weight Transport”

Algorithm 1: DART(ADV, ROB; ϵ)

Input: Algorithms ADV, ROB; parameter $\epsilon > 0$

Output: Distributions $p_1, \dots, p_T \in \Delta(X)$ chosen online

$\lambda_0 \leftarrow 0$

for $t = 1, 2, \dots, T$ **do**

Observe f_t, $a_t :=$ ADV$_t$, and $r_t :=$ ROB$_t$

if $\mathrm{Cost}_{1:t}(\mathrm{ROB}) \geq \frac{\epsilon}{4} \cdot \mathrm{Cost}_{1:t}(\mathrm{ADV})$ **then**

$\lambda_t \leftarrow 1$

else

$\lambda_t \leftarrow \max\left\{\lambda_{t-1} - \frac{\frac{\epsilon}{2}\mathrm{Cost}_t(\mathrm{ADV}) + (1-\lambda_{t-1})f_t(a_t)}{2d(a_{t-1}, r_{t-1})}, 0\right\}$

$p_t \leftarrow \lambda_t \delta_{a_t} + (1-\lambda_t)\delta_{r_t}$

end

When ADV performs well, follow it exactly ($\lambda_t = 1$)

Otherwise, decrease λ_t (move toward ROB)

Choose a_t with prob. λ_t, r_t with prob. $1 - \lambda_t$

DART: Performance Bound

Theorem. For any MTS and any $\epsilon > 0$, DART is $(1 + \epsilon)$-consistent and $2^{\mathcal{O}(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Proof idea: consistency comes by design;
for robustness, must bound how far astray "bad" advice can lead you

Exponential tradeoff between robustness and consistency - is this necessary?

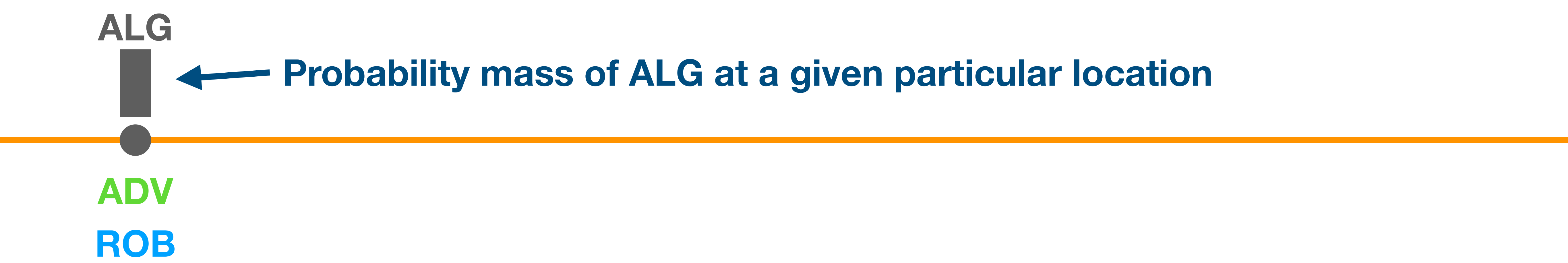
Fundamental limit on robustness-consistency

DART achieves the optimal tradeoff:

Theorem. Any $(1+\epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Proof idea: ADV doubles its distance from ROB every step,
ALG can send at most $\mathcal{O}(\epsilon)$ of its mass back while staying $(1+\epsilon)$-consistent



Fundamental limit on robustness-consistency

Theorem. Any $(1+\epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS ‘23

Proof idea: ADV doubles its distance from ROB every step, ALG can send at most $\mathcal{O}(\epsilon)$ of its mass back while staying $(1+\epsilon)$-consistent

Fundamental limit on robustness-consistency

Theorem. Any $(1 + \epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Proof idea: ADV doubles its distance from ROB every step,
ALG can send at most $\mathcal{O}(\epsilon)$ of its mass back while staying $(1 + \epsilon)$-consistent

Fundamental limit on robustness-consistency

Theorem. Any $(1+\epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Proof idea: ADV doubles its distance from ROB every step,
ALG can send at most $\mathcal{O}(\epsilon)$ of its mass back while staying $(1+\epsilon)$-consistent

Fundamental limit on robustness-consistency

Theorem. Any $(1+\epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Proof idea: ADV doubles its distance from ROB every step,
ALG can send at most $\mathcal{O}(\epsilon)$ of its mass back while staying $(1+\epsilon)$-consistent

After $\mathcal{O}(1/\epsilon)$ steps, ADV has moved distance $2^{\mathcal{O}(1/\epsilon)}$

Fundamental limit on robustness-consistency

Theorem. Any $(1+\epsilon)$-consistent randomized algorithm for MTS is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Synthetic lower bound; similar pessimistic result for other cases?

Yes! Same lower bound holds for convex hitting costs - convex function chasing in ℓ^1

Theorem. Any $(1+\epsilon)$-consistent algorithm for convex function chasing is at least $2^{\Omega(1/\epsilon)}$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Beyond the worst-case exponential tradeoff

With certain structure, DART can perform even better:

Theorem. If $d(a_t, r_t) \leq D$ for all time, then for any $\epsilon > 0$, DART is $(1+\epsilon)$-consistent and

$$\mathrm{Cost}(\text{DART}) \leq \mathcal{O}(1/\epsilon)\mathrm{Cost}(\text{ROB}) + \mathcal{O}(D/\epsilon)$$

Christianson, Shen, and Wierman, AISTATS '23

Same algorithm, doesn't need D a priori! Follows by a specialized analysis

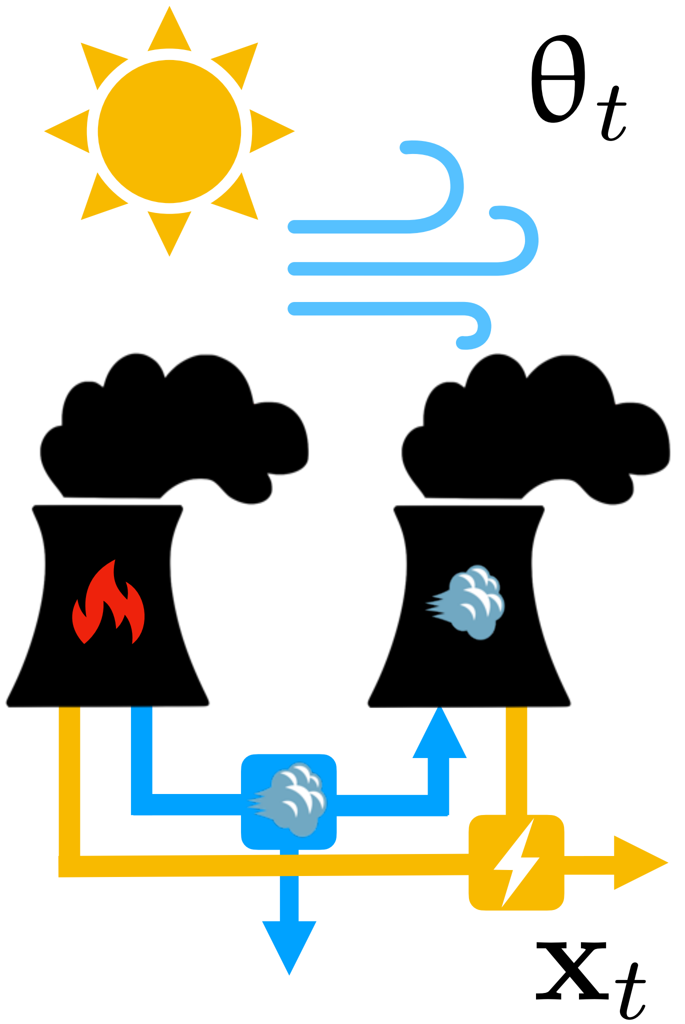
Corollary. For the k-server problem, for any $\epsilon > 0$, DART is $(1+\epsilon)$-consistent and $\mathcal{O}(k/\epsilon)$-robust.

Christianson, Shen, and Wierman, AISTATS '23

Follows because for k-server, $D \leq k \cdot \mathrm{Cost}(\mathtt{OPT})$

Cogeneration experiments

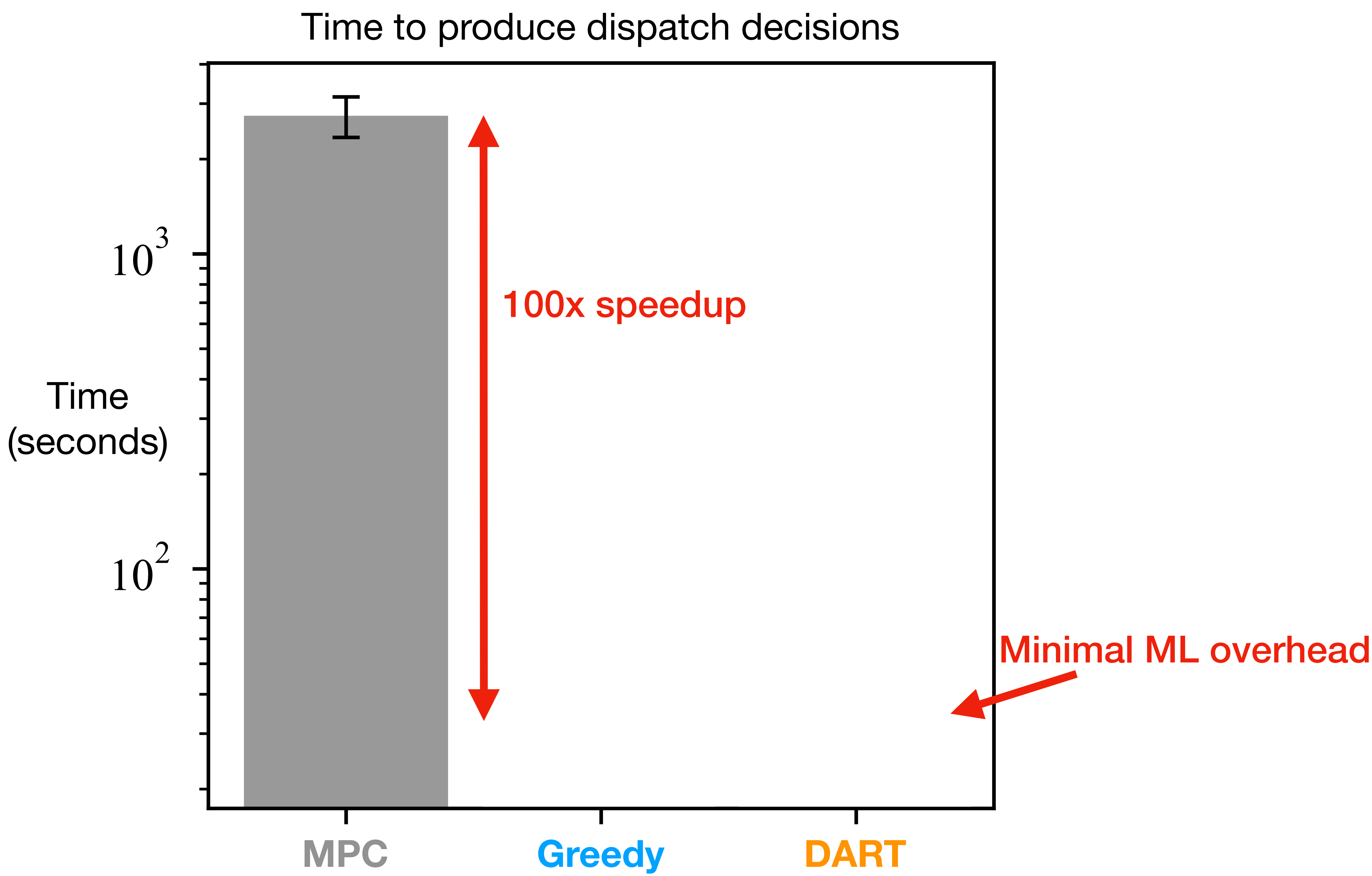
- Real-world cogeneration plant model*
- Ramp costs + nonconvex fuel costs f_t
- Beyond Limits wanted to use forecasts; Model Predictive Control (MPC) too slow due to nonconvexity!
- ML trained as ADV to choose dispatch $\mathbf{x}_t$ using lookahead predictions $\hat{\theta}_{t+j}$
- Combined with greedy baseline - ROB simply minimizes f_t



***Available in SustainGym (NeurIPS '23); website here:**

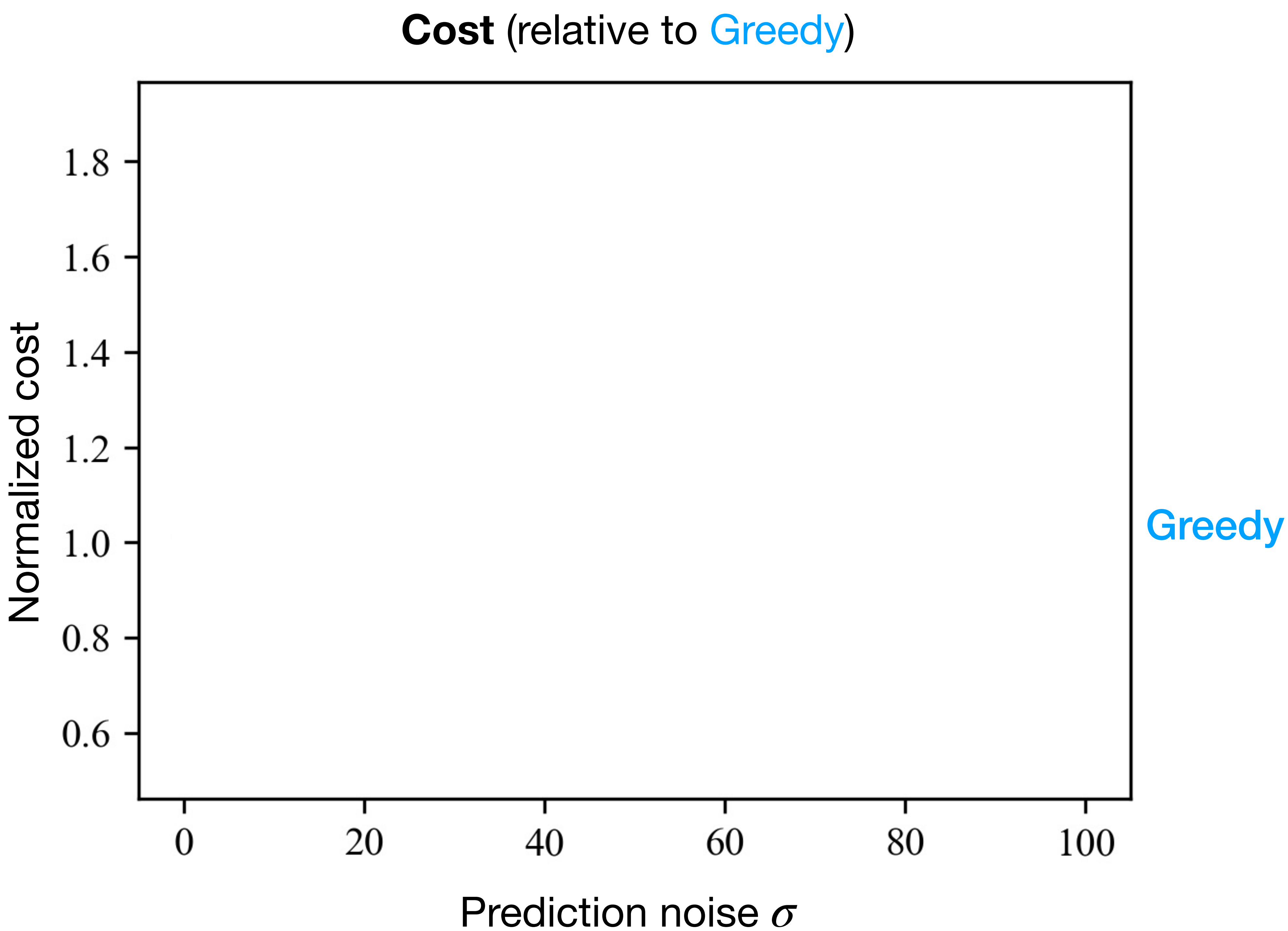
Yeh, Li, Datta, Arroyo, **Christianson**, et al. (NeurIPS '23)

ML + DART yields significant computational speedups



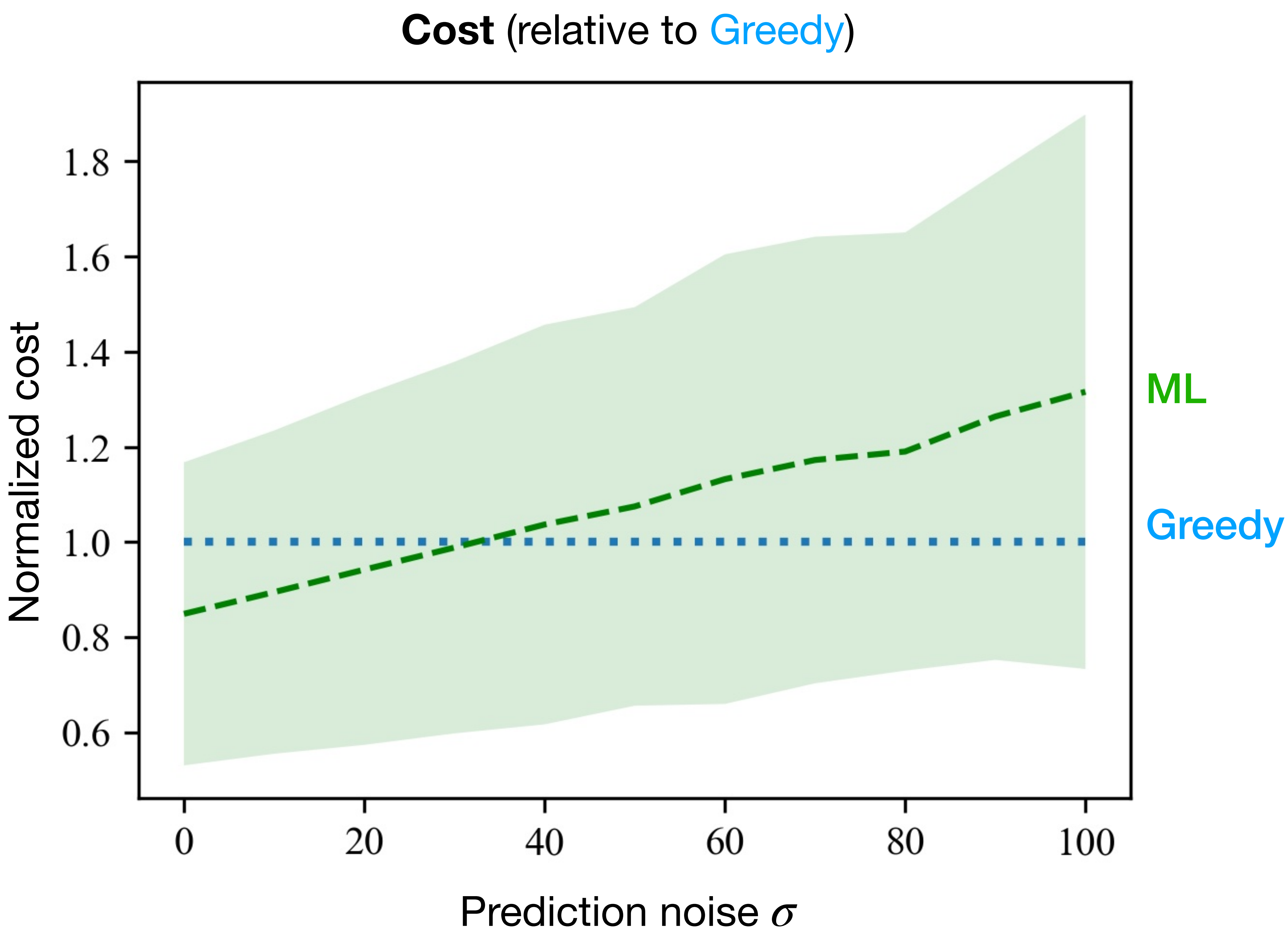
DART performs better than both ML *and* Greedy

Significant robustness in the face of distribution shift:



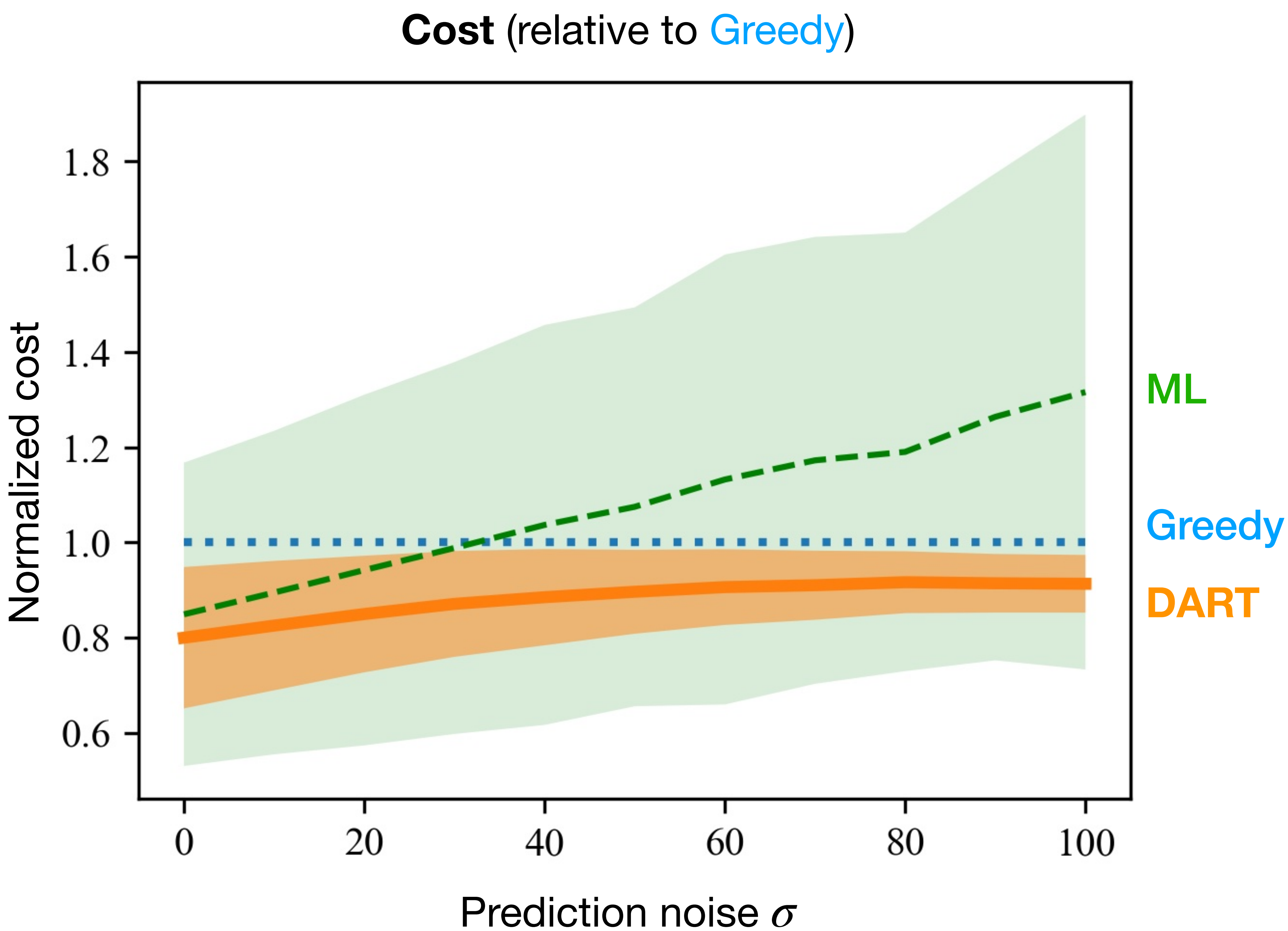
DART performs better than both ML *and* Greedy

Significant robustness in the face of distribution shift:



DARTperforms better than both ML *and* Greedy

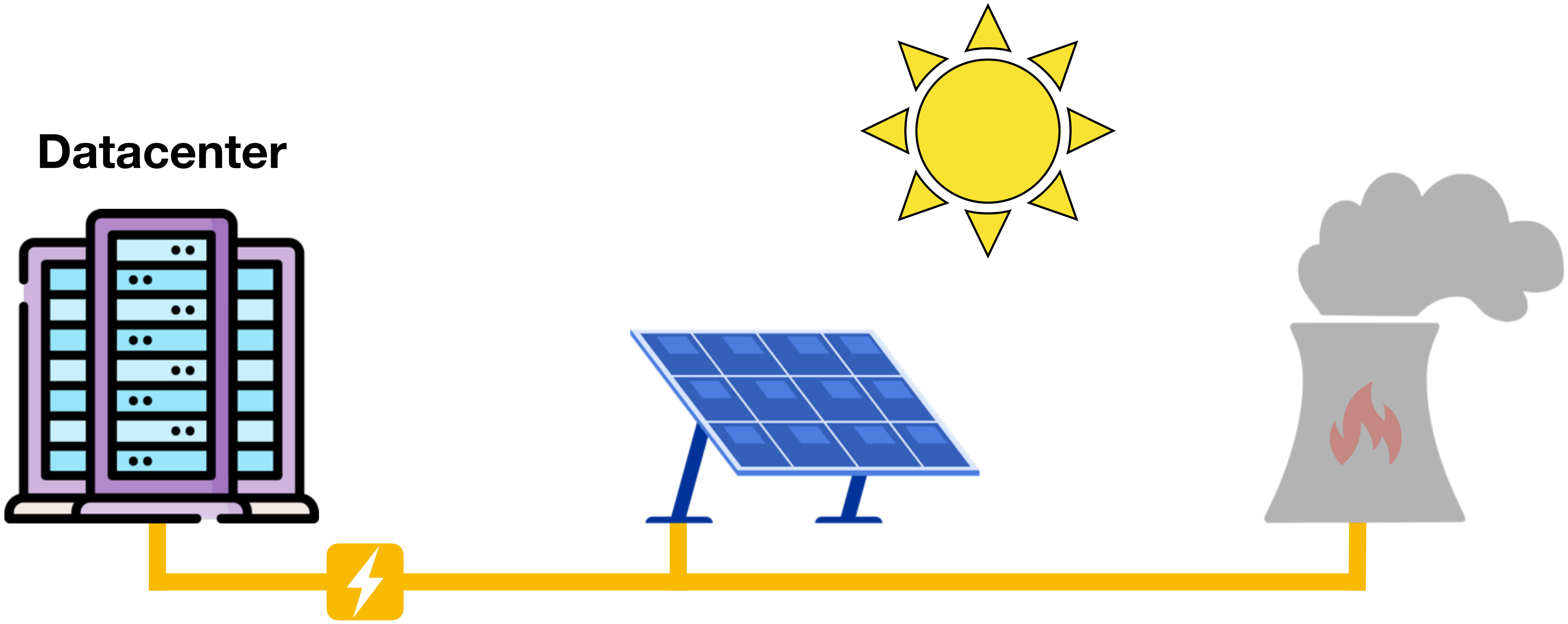
Significant robustness in the face of distribution shift:



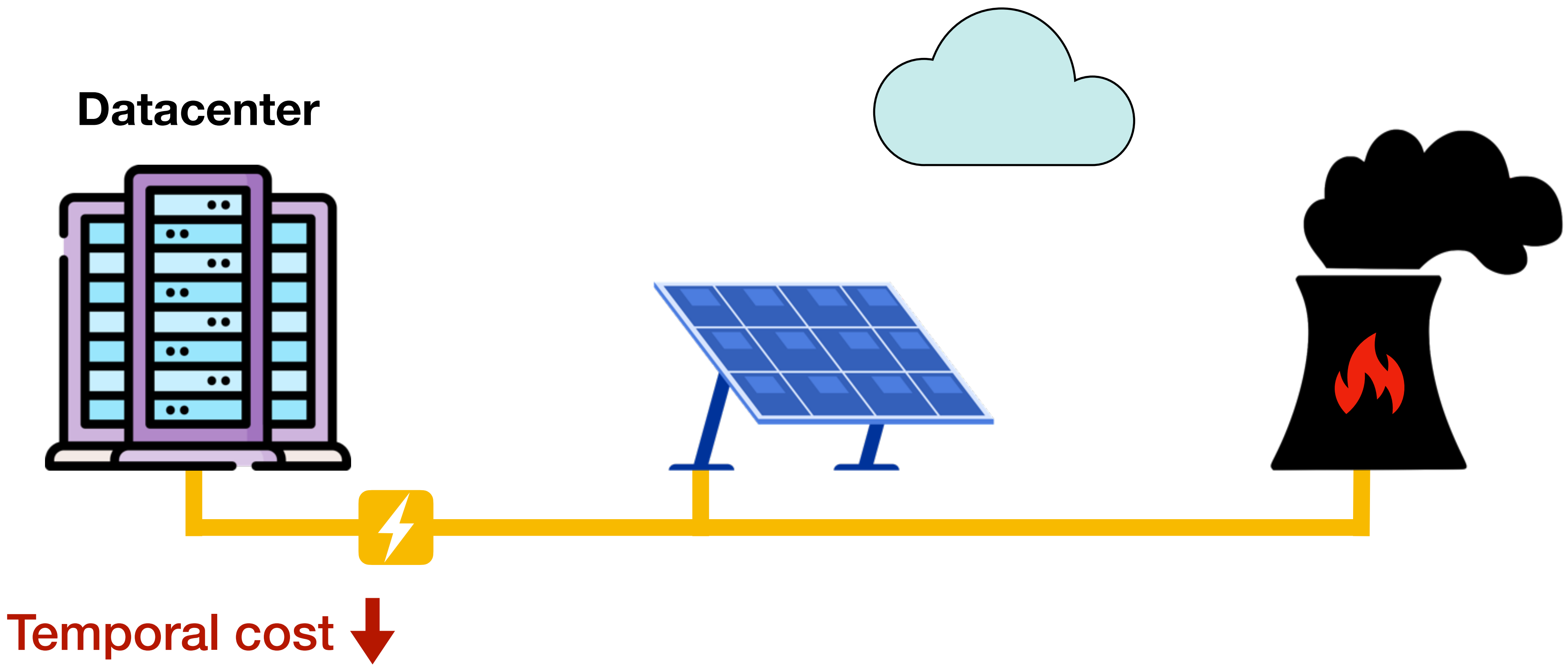
DART delivers best-of-both-worlds performance for MTS and cogeneration operation…

…where else can black-box AI/ML advice enable better performance for energy + sustainability problems?

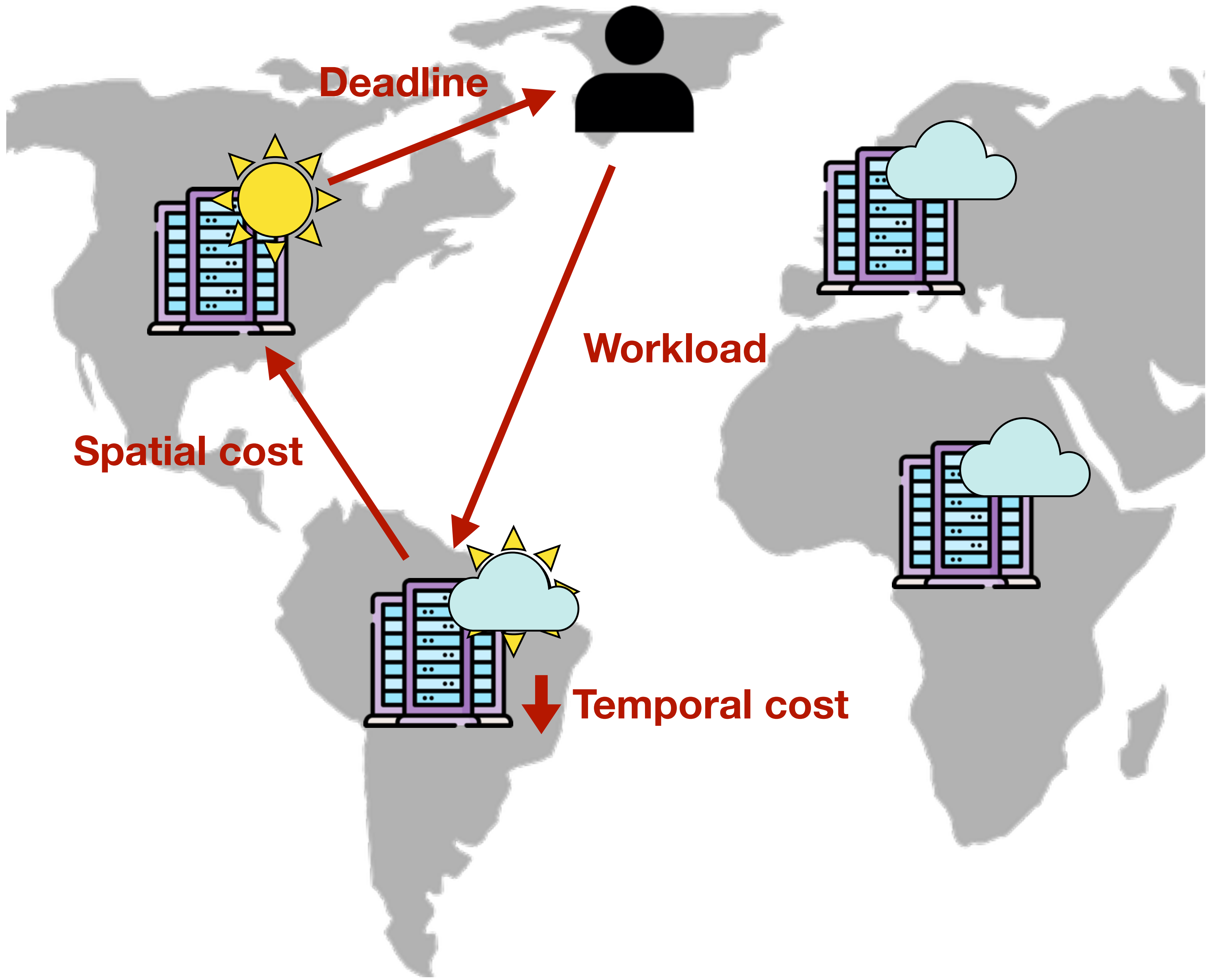
Carbon-aware *temporal* load shifting



Carbon-aware *temporal* load shifting



Carbon-aware *spatiotemporal* load shifting



Carbon-aware *spatiotemporal* load shifting

Model as online optimization with switching costs and deadline:

Carbon cost

Multi-scale switching cost

$$\min_{x_1,\ldots,x_T\in\mathcal{X}} \quad \sum_{t=1}^{T} f_t(x_t) + d(x_t, x_{t-1})$$

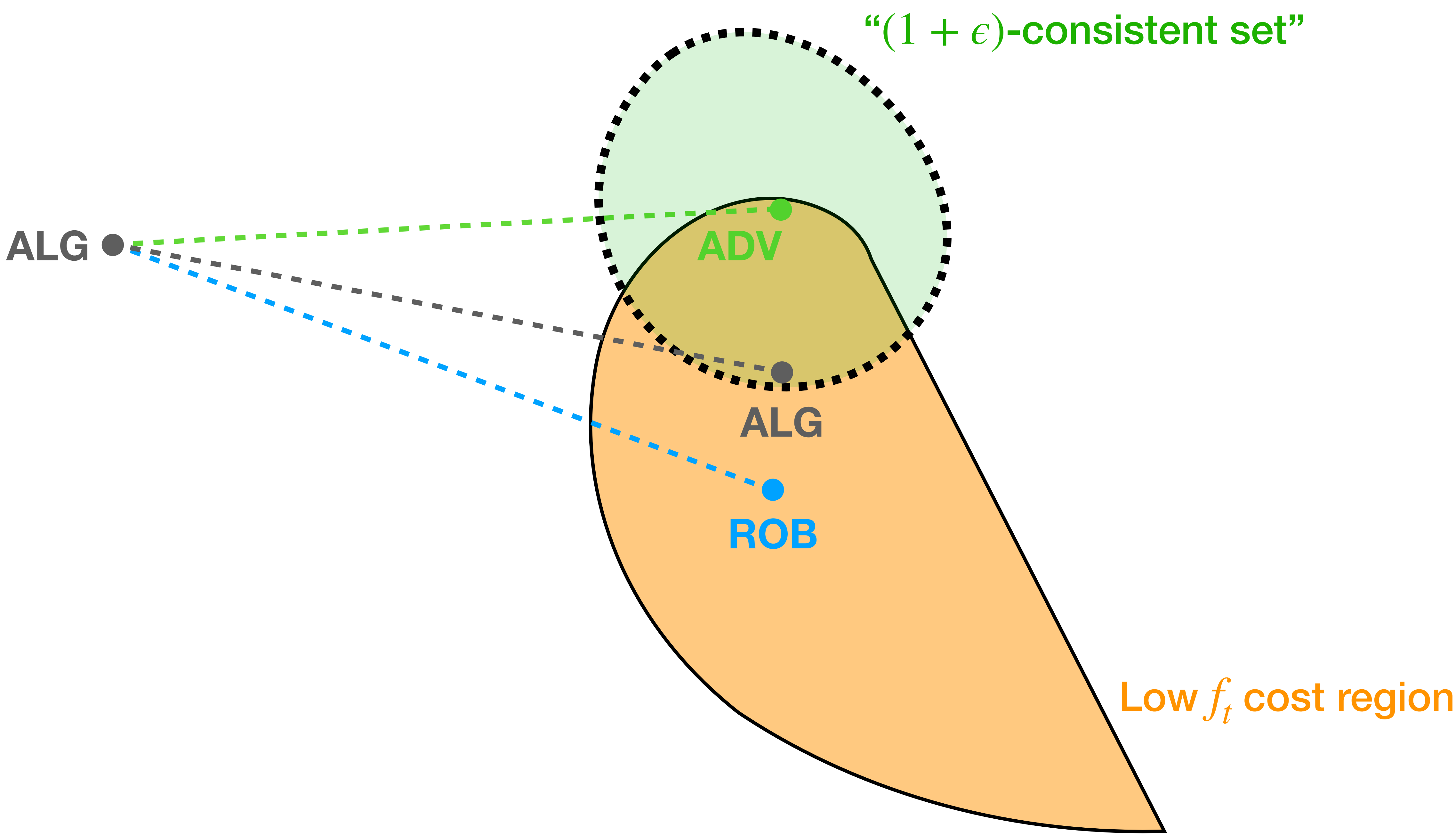
$$\text{s.t.} \quad \sum_{t=1}^{T} c(x_t) \geq 1$$

Deadline constraint

Challenging multi-scale structure + hard deadline —
can we robustly leverage black-box AI/ML advice here?

Revisiting the “consistent set” idea

How should we select x_t to be consistent and robust?



Revisiting the “consistent set” idea with ST-CLIP

Consistency-constrained pseudo-cost minimization

Algorithm 2 ST-CLIP (spatiotemporal consistency-limited pseudo-cost minimization) for SOAD

---

input: Consistency parameter ε, constraint function $c(\cdot)$, pseudo-cost $\psi^{(\varepsilon)}(\cdot)$, starting OFF state $s \in \mathcal{S}$.
initialize: $z^{(0)} = 0$; $\rho^{(0)} = 0$; $A^{(0)} = 0$; $\text{SC}_0 = 0$; $\text{ADV}_0 = 0$; $\mathbf{k}_0 = \Phi\delta_s$; $\mathbf{p}_0 = \mathbf{a}_0 = \delta_s$.
while cost function $f_t(\cdot)$ is revealed, untrusted advice $\mathbf{a}_t$ is revealed, and $z^{(t-1)} < 1$ **do**

Update advice cost $\text{ADV}_t \leftarrow \text{ADV}_{t-1} + f_t(\mathbf{a}_t) + \mathbb{W}^1(\mathbf{a}_t, \mathbf{a}_{t-1})$ and advice utilization $A^{(t)} \leftarrow A^{(t-1)} + c(\mathbf{a}_t)$.
Solve **constrained** pseudo-cost minimization problem:

$$\mathbf{k}_t = \underset{\mathbf{k} \in K : \bar{c}(\mathbf{k}) \leq 1 - z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k} - \mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)} + \bar{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du, \tag{6}$$

such that $\mathbf{p} \leftarrow \Phi^{-1}\mathbf{k}$ and,

$$\text{SC}_{t-1} + f_t(\mathbf{p}) + \mathbb{W}^1(\mathbf{p}, \mathbf{p}_{t-1}) + \mathbb{W}^1(\mathbf{p}, \mathbf{a}_t) + \tau c(\mathbf{a}_t) + (1 - z^{(t-1)} - c(\mathbf{p}))L + \max\{A^{(t)} - z^{(t-1)} - c(\mathbf{p}), 0\}(U - L) \leq (1 + \varepsilon)[\text{ADV}_t + \tau c(\mathbf{a}_t) + (1 - A^{(t)})L]. \tag{7}$$

Update running cost $\text{SC}_t \leftarrow \text{SC}_{t-1} + f_t(\mathbf{p}_t) + \mathbb{W}^1(\mathbf{p}_t, \mathbf{p}_{t-1})$ and utilization $z^{(t)} \leftarrow z^{(t-1)} + c(\mathbf{p}_t)$.
Solve **unconstrained** pseudo-cost minimization problem:

$$\tilde{\mathbf{k}}_t = \underset{\mathbf{k} \in K : \bar{c}(\mathbf{k}) \leq 1 - z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k} - \mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)} + \bar{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du \tag{8}$$

Update *robust pseudo-utilization* $\rho^{(t)} \leftarrow \rho^{(t-1)} + \min\{\bar{c}(\tilde{\mathbf{k}}_t), c(\mathbf{p}_t)\}$.

Revisiting the “consistent set” idea with ST-CLIP

Algorithm 2 ST-CLIP (spatiotemporal consistency-limited pseudo-cost minimization) for SOAD

input: Consistency parameter ε, constraint function $c(\cdot)$, pseudo-cost $\psi^{(\varepsilon)}(\cdot)$, starting `OFF` state $s \in \mathcal{S}$.
initialize: $z^{(0)} = 0$; $\rho^{(0)} = 0$; $A^{(0)} = 0$; $\mathrm{SC}_0 = 0$; $\textsc{Adv}_0 = 0$; $\mathbf{k}_0 = \Phi\delta_s$; $\mathbf{p}_0 = \mathbf{a}_0 = \delta_s$.
while cost function $f_t(\cdot)$ is revealed, untrusted advice $\mathbf{a}_t$ is revealed, and $z^{(t-1)} < 1$ **do**
Update advice cost $\textsc{Adv}_t \leftarrow \textsc{Adv}_{t-1} + f_t(\mathbf{a}_t) + \mathbb{W}^1(\mathbf{a}_t, \mathbf{a}_{t-1})$ and advice utilization $A^{(t)} \leftarrow A^{(t-1)} + c(\mathbf{a}_t)$.
Solve **constrained** pseudo-cost minimization problem:

pseudo-cost minimization

$$\mathbf{k}_t = \underset{\mathbf{k}\in K:\overline{c}(\mathbf{k})\le 1-z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k}-\mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)}+\overline{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du, \tag{6}$$

such that $\mathbf{p} \leftarrow \Phi^{-1}\mathbf{k}$ and,

$$\begin{aligned} &\mathrm{SC}_{t-1} + f_t(\mathbf{p}) + \mathbb{W}^1(\mathbf{p}, \mathbf{p}_{t-1}) + \mathbb{W}^1(\mathbf{p}, \mathbf{a}_t) + \tau c(\mathbf{a}_t) + (1 - z^{(t-1)} - c(\mathbf{p}))L + \max\{A^{(t)} - z^{(t-1)} - c(\mathbf{p}), 0\}(U-L) \\ &\qquad \le (1+\varepsilon)[\textsc{Adv}_t + \tau c(\mathbf{a}_t) + (1 - A^{(t)})L]. \end{aligned} \tag{7}$$

Update running cost $\mathrm{SC}_t \leftarrow \mathrm{SC}_{t-1} + f_t(\mathbf{p}_t) + \mathbb{W}^1(\mathbf{p}_t, \mathbf{p}_{t-1})$ and utilization $z^{(t)} \leftarrow z^{(t-1)} + c(\mathbf{p}_t)$.
Solve **unconstrained** pseudo-cost minimization problem:

$$\tilde{\mathbf{k}}_t = \underset{\mathbf{k}\in K:\overline{c}(\mathbf{k})\le 1-z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k}-\mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)}+\overline{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du \tag{8}$$

Update *robust pseudo-utilization* $\rho^{(t)} \leftarrow \rho^{(t-1)} + \min\{\overline{c}(\tilde{\mathbf{k}}_t), c(\mathbf{p}_t)\}$.

Revisiting the "consistent set" idea with ST-CLIP

Algorithm 2 ST-CLIP (spatiotemporal consistency-limited pseudo-cost minimization) for SOAD

input: Consistency parameter ε, constraint function $c(\cdot)$, pseudo-cost $\psi^{(\varepsilon)}(\cdot)$, starting OFF state $s \in \mathcal{S}$.
initialize: $z^{(0)} = 0$; $\rho^{(0)} = 0$; $A^{(0)} = 0$; $\text{SC}_0 = 0$; $\textsc{Adv}_0 = 0$; $\mathbf{k}_0 = \Phi\delta_s$; $\mathbf{p}_0 = \mathbf{a}_0 = \delta_s$.
while cost function $f_t(\cdot)$ is revealed, untrusted advice $\mathbf{a}_t$ is revealed, and $z^{(t-1)} < 1$ **do**

Update advice cost $\textsc{Adv}_t \leftarrow \textsc{Adv}_{t-1} + f_t(\mathbf{a}_t) + \mathbb{W}^1(\mathbf{a}_t, \mathbf{a}_{t-1})$ and advice utilization $A^{(t)} \leftarrow A^{(t-1)} + c(\mathbf{a}_t)$.
Solve **constrained** pseudo-cost minimization problem:

$$\mathbf{k}_t = \underset{\mathbf{k} \in K : \bar{c}(\mathbf{k}) \le 1 - z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k} - \mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)} + \bar{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du, \tag{6}$$

such that $\mathbf{p} \leftarrow \Phi^{-1}\mathbf{k}$ and,

consistency constraint

$$\text{SC}_{t-1} + f_t(\mathbf{p}) + \mathbb{W}^1(\mathbf{p}, \mathbf{p}_{t-1}) + \mathbb{W}^1(\mathbf{p}, \mathbf{a}_t) + \tau c(\mathbf{a}_t) + (1 - z^{(t-1)} - c(\mathbf{p}))L + \max\{A^{(t)} - z^{(t-1)} - c(\mathbf{p}), 0\}(U - L) \\ \le (1+\varepsilon)\left[\textsc{Adv}_t + \tau c(\mathbf{a}_t) + (1 - A^{(t)})L\right]. \tag{7}$$

Update running cost $\text{SC}_t \leftarrow \text{SC}_{t-1} + f_t(\mathbf{p}_t) + \mathbb{W}^1(\mathbf{p}_t, \mathbf{p}_{t-1})$ and utilization $z^{(t)} \leftarrow z^{(t-1)} + c(\mathbf{p}_t)$.
Solve **unconstrained** pseudo-cost minimization problem:

$$\tilde{\mathbf{k}}_t = \underset{\mathbf{k} \in K : \bar{c}(\mathbf{k}) \le 1 - z^{(t-1)}}{\arg\min} \overline{f}_t(\mathbf{k}) + \|\mathbf{k} - \mathbf{k}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{\rho^{(t-1)}}^{\rho^{(t-1)} + \bar{c}(\mathbf{k})} \psi^{(\varepsilon)}(u)\, du \tag{8}$$

Update *robust pseudo-utilization* $\rho^{(t)} \leftarrow \rho^{(t-1)} + \min\{\bar{c}(\tilde{\mathbf{k}}_t), c(\mathbf{p}_t)\}$.

ST-CLIP: Performance Bound

Number of datacenters/locations

Theorem. **ST-CLIP** achieves $(1+\varepsilon)$-consistency and $\mathcal{O}(\log n)\gamma^{(\varepsilon)}$-robustness, where $\gamma^{(\varepsilon)}$ is the solution to a certain transcendental equation.

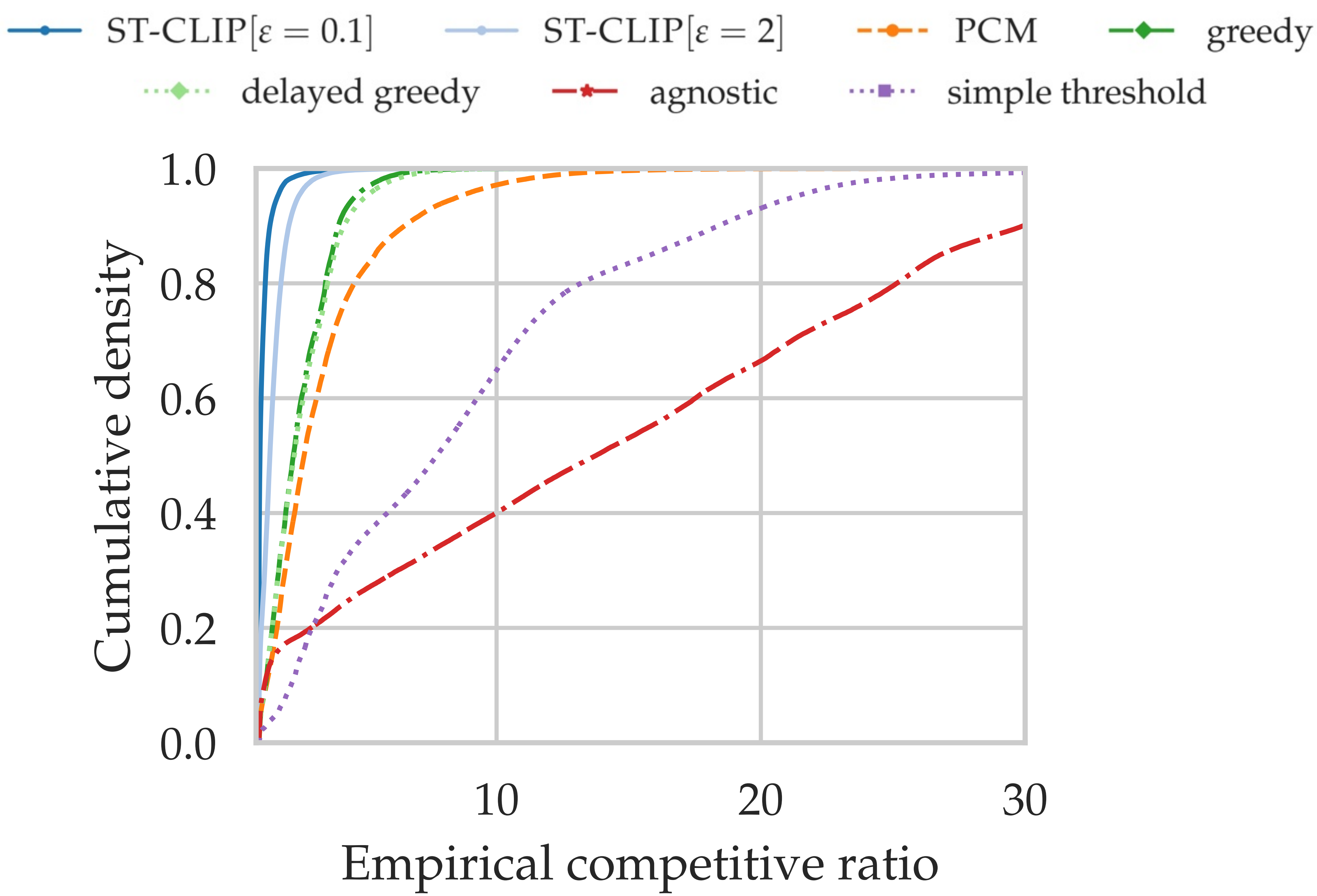
Lechowicz, **Christianson**, et al., SIGMETRICS '25

Theorem. **ST-CLIP**'s robustness-consistency tradeoff is **optimal** up to the $\mathcal{O}(\log n)$ factor.

Lechowicz, **Christianson**, et al., SIGMETRICS '25

ST-CLIP: Experimental Results

Case study on carbon-aware spatiotemporal load balancing with Google cluster data

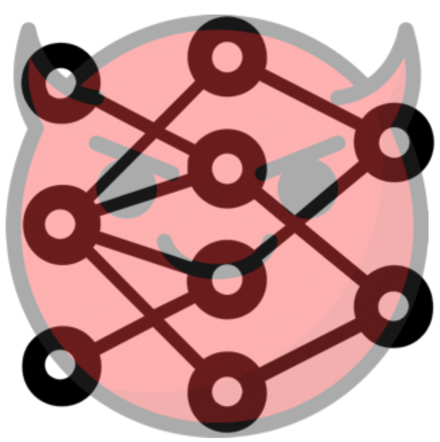


Interlude

Robust and consistent algorithms let us leverage **black-box AI/ML** with **worst-case guarantees**

More control over guarantees

“Black-box”



No guarantees on model performance

“Grey-box”

Model provides *some* uncertainty estimates

“End-to-End”

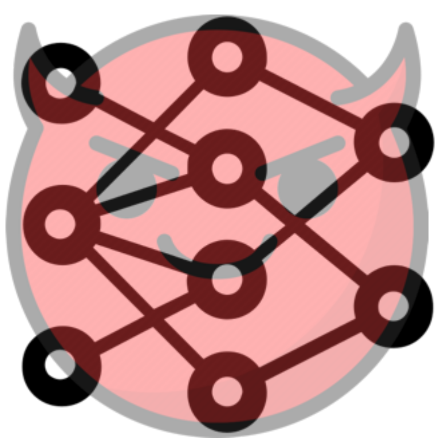
Trained for specific task with reliability (e.g., UQ) in-the-loop

Interlude

How can we design new **algorithms** and **AI/ML training procedures** that prioritize these criteria?

More control over guarantees

“Black-box”



No guarantees on model performance

“Grey-box”

Model provides *some* uncertainty estimates

“End-to-End”

Trained for specific task with reliability (e.g., UQ) in-the-loop

Many considerations (risk, uncertainty, constraints…) when deploying AI/ML in the real world

Interlude

How can we train ML models *End-to-End* with uncertainty quantification (UQ) + other reliability criteria?

More control over guarantees

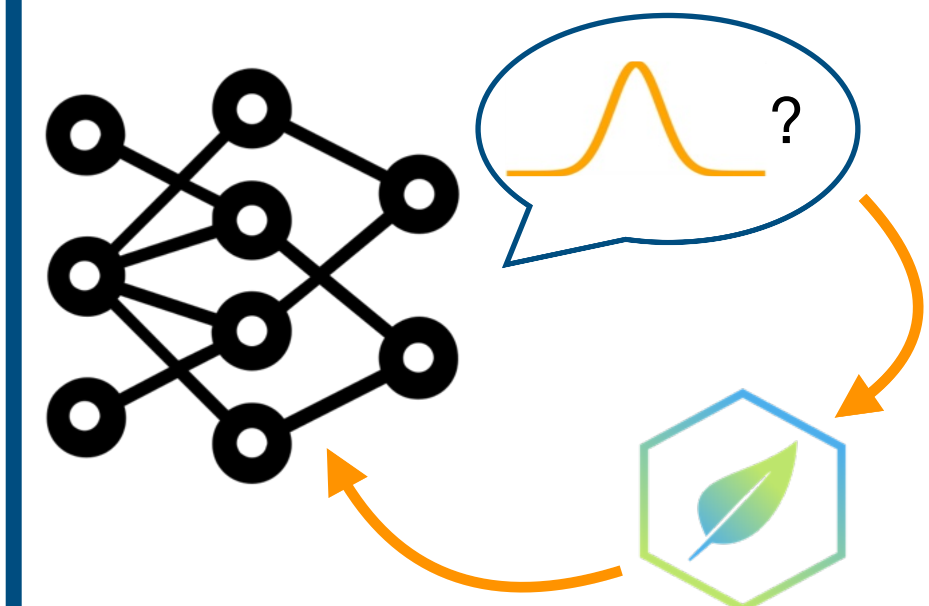
“Black-box”

No guarantees on model performance

“Grey-box”

Model provides *some* uncertainty estimates

“End-to-End”

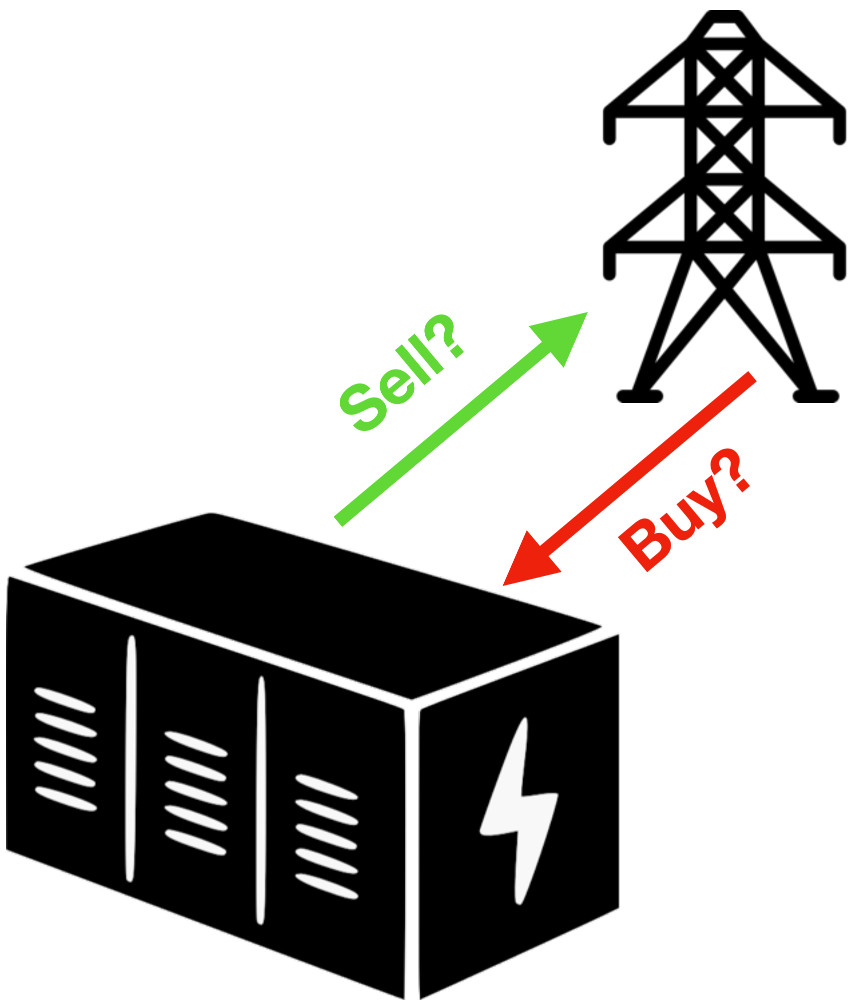


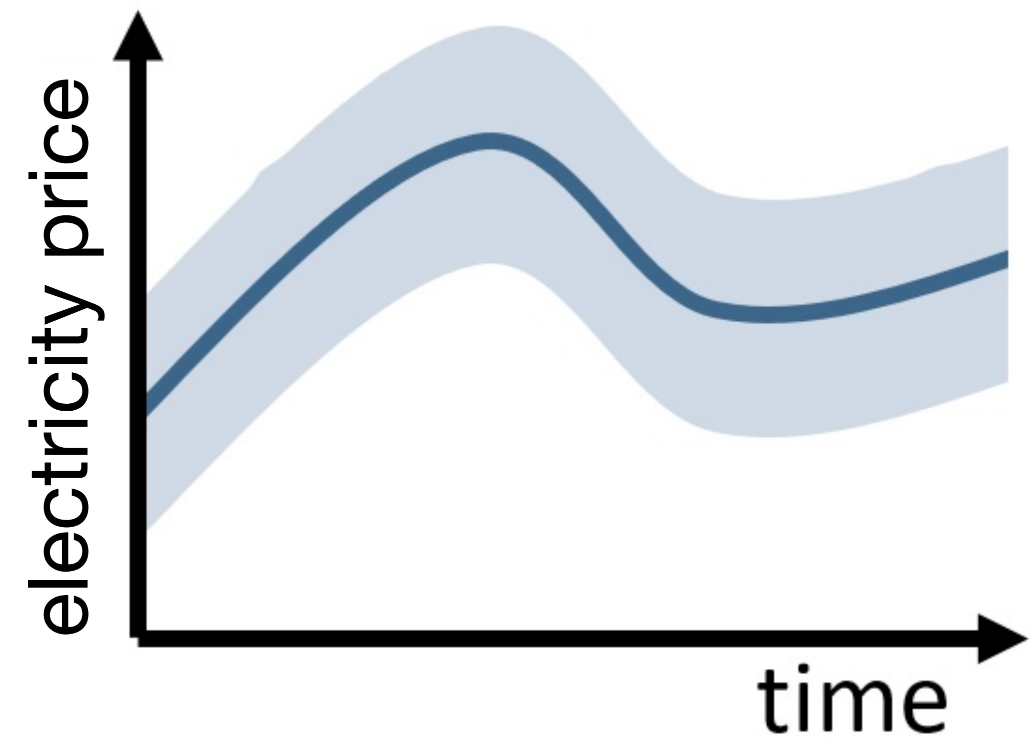
Trained for specific task with reliability (e.g., UQ) in-the-loop

Second half of today’s talk

Motivation: Grid-scale energy storage operation

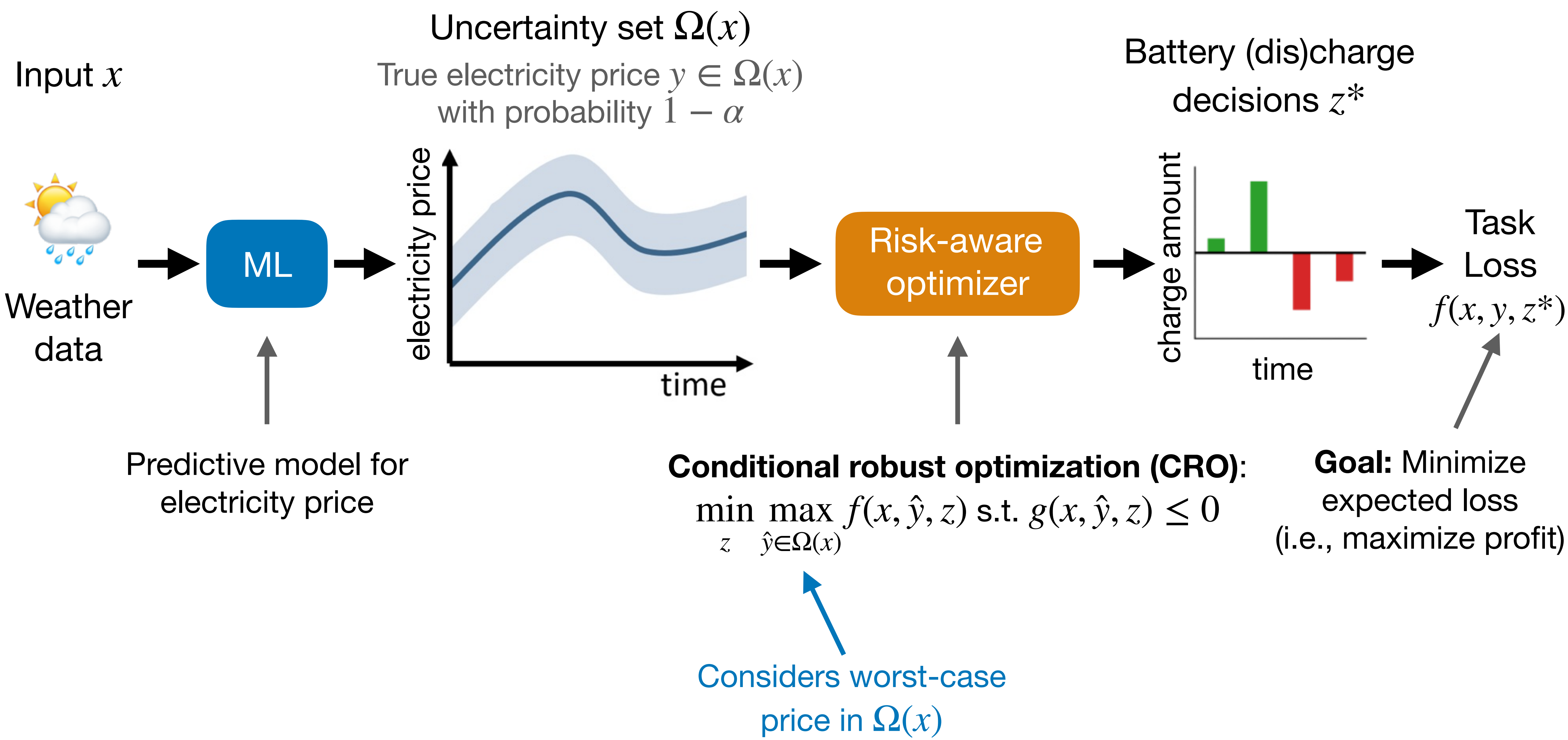
Storage operators must manage uncertainty/risk to avoid losses





Motivation: Grid-scale energy storage operation

Storage operators must manage uncertainty/risk to avoid losses



Typical approach: *“Estimate, then Optimize”*

1. Estimate uncertainty set from data

Input x → ML → Uncertainty set $\Omega(x)$ → Risk-aware optimizer → Decisions z^* → Task Loss $f(x, y, z^*)$

Restrictive uncertainty representations:

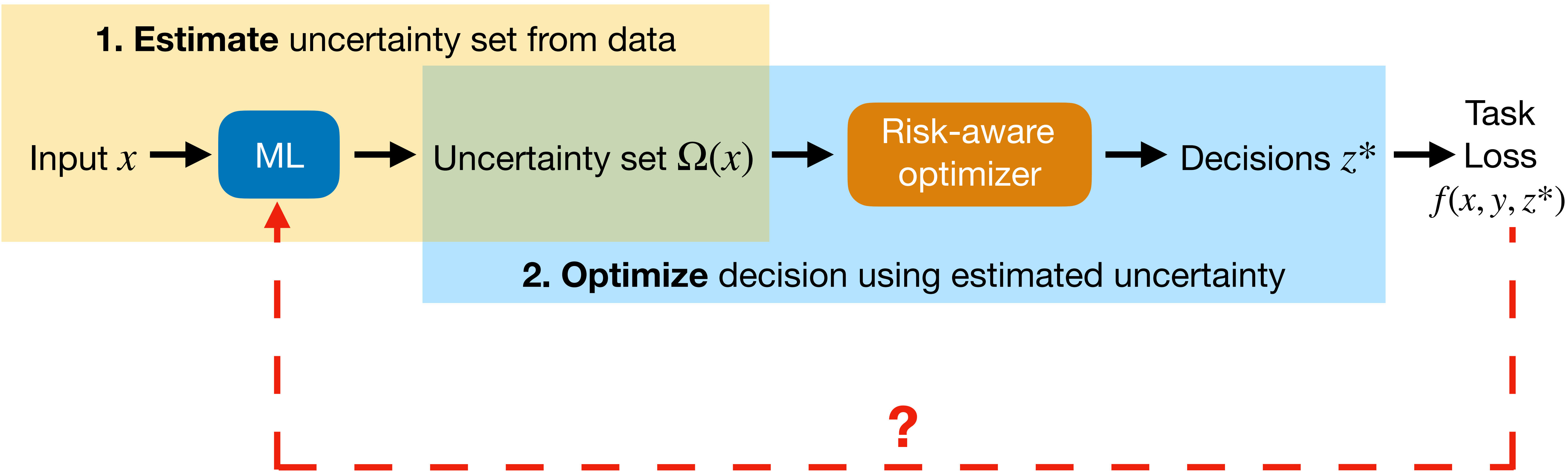
- Box
- Ellipsoidal

+ calibration (with, e.g., conformal prediction)

Key challenges:

1. Can we learn more *expressive* uncertainty sets - e.g., general convex sets?

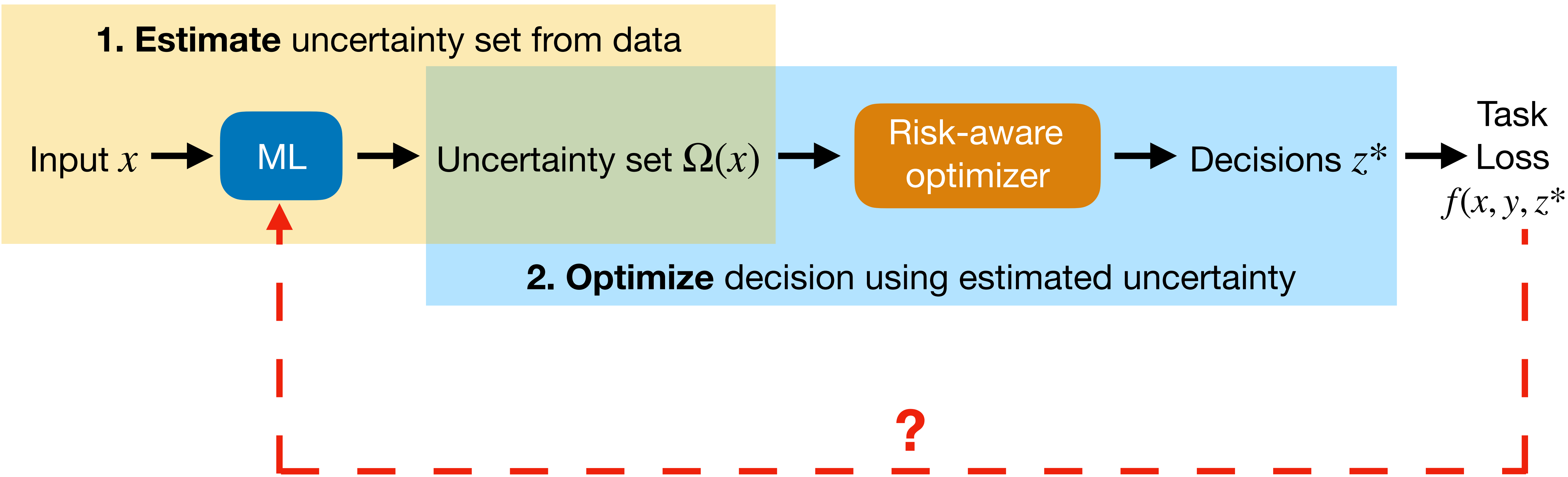
Typical approach: *"Estimate, then Optimize"*



Key challenges:

1. Can we learn more *expressive* uncertainty sets - e.g., general convex sets?

Typical approach: *“Estimate, then Optimize”*



Key challenges:

1. Can we learn more *expressive* uncertainty sets - e.g., general convex sets?
2. Can we train the ML model + uncertainty estimates *using the task loss*?

Solution 1: General convex uncertainties via *Partially Input-Convex Neural Networks (PICNNs)*

Input x → ML → Uncertainty set $\Omega(x)$ → Risk-aware optimizer → Decisions z^* → Task Loss $f(x, y, z^*)$

Can calibrate with conformal prediction

Define $\Omega(x) = \{\hat{y} : s_\theta(x, \hat{y}) \leq q\}$, where s_θ is a PICNN:

When $\bar{W}_l \geq 0$, $s_\theta(x, y)$ is convex in y

$\implies$ $\Omega(x)$ is convex

$$s_\theta(x, \hat{y}) = W_L \sigma_L + V_L \hat{y}_L + b_L$$

$$\begin{aligned}
\sigma_0 &= \mathbf{0}, \qquad u_0 = x & W_l &= \bar{W}_l \operatorname{diag}([\hat{W}_l u_l + w_l]_+) \\
\sigma_{l+1} &= \mathrm{ReLU}\left(W_l \sigma_l + V_l \hat{y} + b_l\right) & V_l &= \bar{V}_l \operatorname{diag}(\hat{V}_l u_l + v_l) \\
u_{l+1} &= \mathrm{ReLU}\left(R_l u_l + r_l\right) & b_l &= \bar{B}_l u_l + \bar{b}_l.
\end{aligned}$$

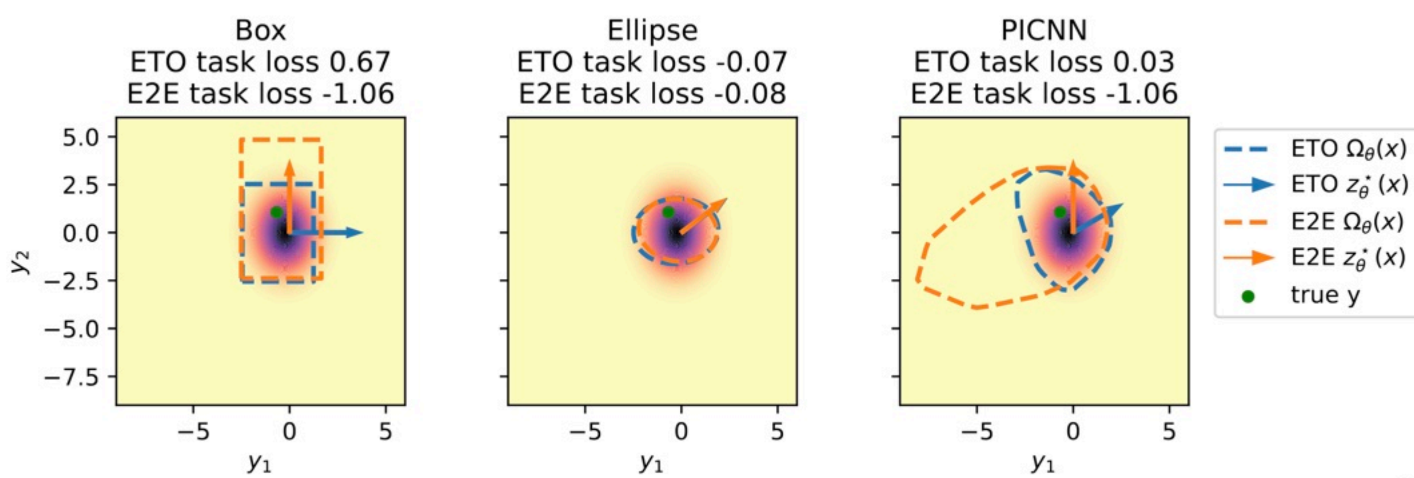
Solution 1: General convex uncertainties via *Partially Input-Convex Neural Networks (PICNNs)*

Conditional robust optimization (CRO):

$$\min_{z} \max_{\hat{y} \in \Omega(x)} f(x, \hat{y}, z) \text{ s.t. } g(x, \hat{y}, z) \leq 0$$

Input x → ML → Uncertainty set $\Omega(x)$ → Risk-aware optimizer → Decisions z^* → Task Loss $f(x, y, z^*)$

Define $\Omega(x) = \{\hat{y} : s_\theta(x, \hat{y}) \leq q\}$, where s_θ is a PICNN:



PICNNs can approximate general convex uncertainty sets…

but are they tractable in CRO?

Solution 1: General convex uncertainties via *Partially Input-Convex Neural Networks (PICNNs)*

Conditional robust optimization (CRO):

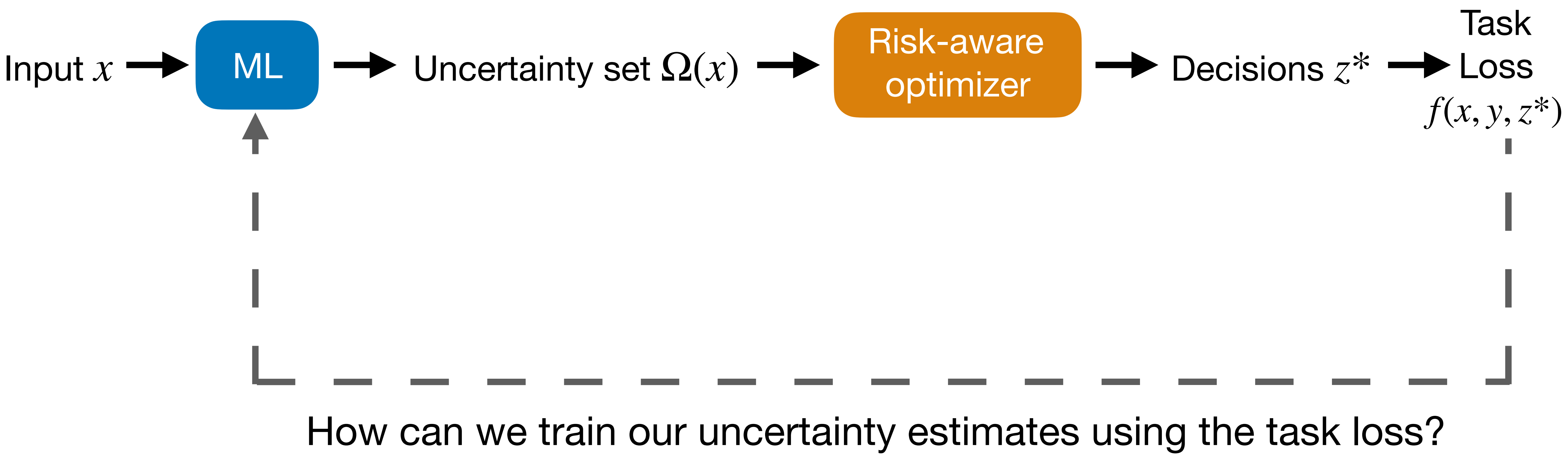
$$\min_{z} \max_{\hat{y} \in \Omega(x)} f(x, \hat{y}, z) \text{ s.t. } g(x, \hat{y}, z) \leq 0$$

Input x → ML → Uncertainty set $\Omega(x)$ → Risk-aware optimizer → Decisions z^* → Task Loss $f(x, y, z^*)$

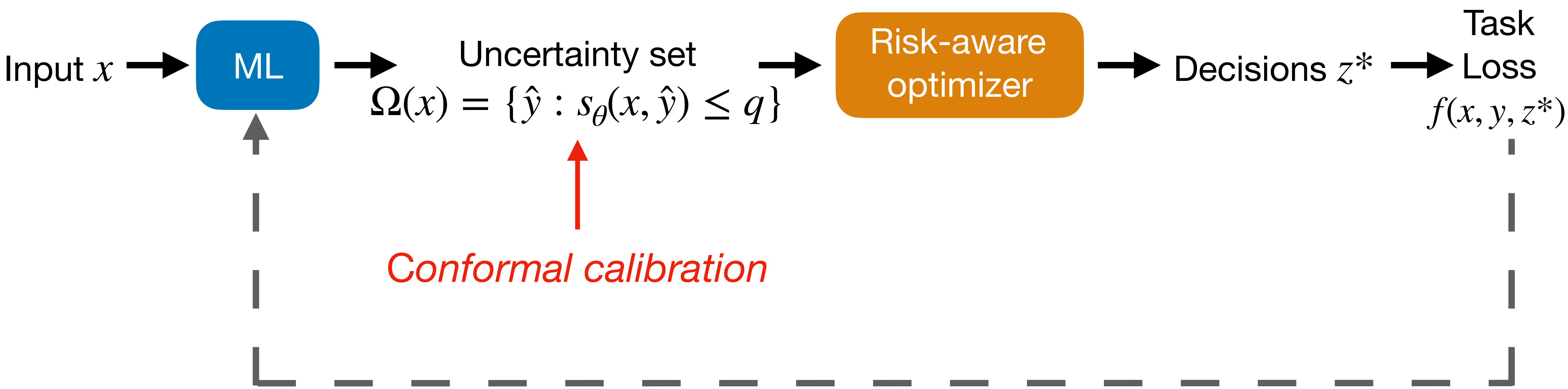
Theorem. Under suitable conditions on f and g, if $\Omega(x)$ is represented by a PICNN, then the CRO problem has an *exact, tractable* convex reformulation.

Yeh*, **Christianson***, Wu, Wierman, Yue (Under review)

Solution 2: *End-to-End* learning over calibrated uncertainties



Solution 2: *End-to-End* learning over calibrated uncertainties



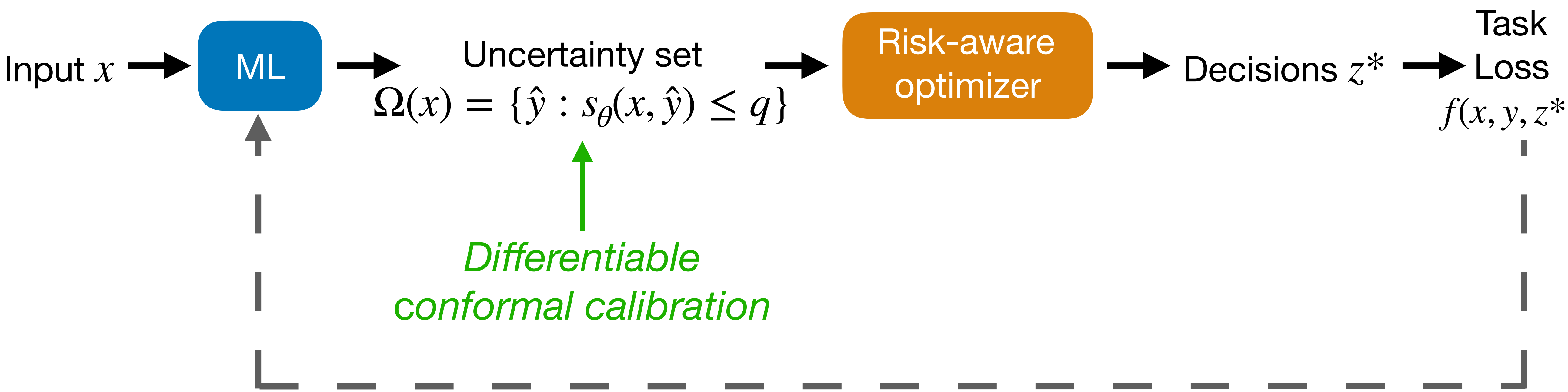
How can we train our uncertainty estimates using the task loss?

Conformal calibration* chooses q so $\mathbb{P}(y \in \Omega(x)) \geq 1 - \alpha$ using held-out calibration data (empirical quantile of s_θ)

Depends on a sorting procedure (to obtain quantile) which is not differentiable

* Angelopoulos and Bates 2023

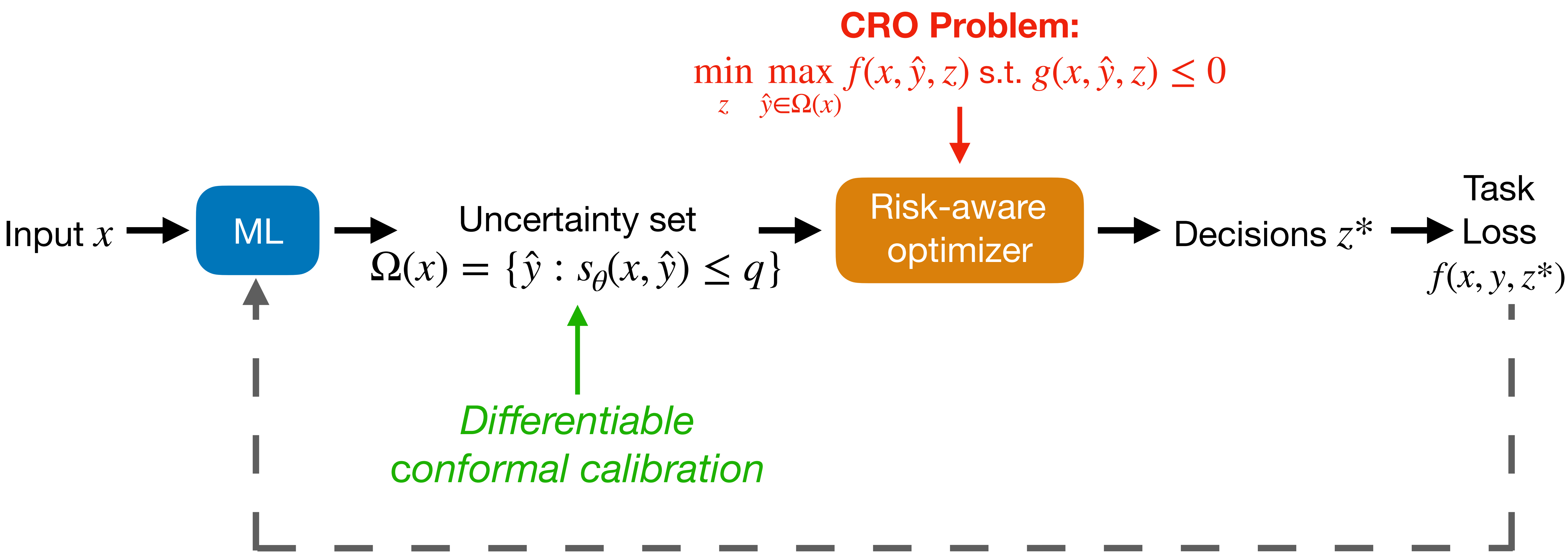
Solution 2: *End-to-End* learning over calibrated uncertainties



How can we train our uncertainty estimates using the task loss?

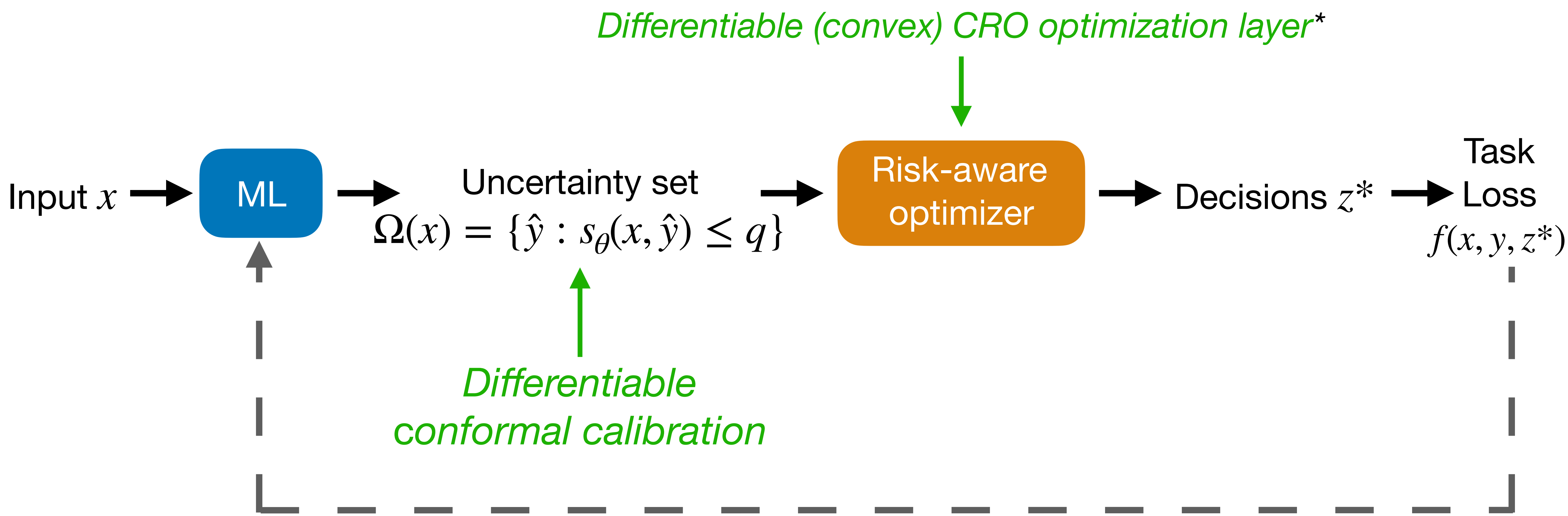
Insight: differentiate through q (empirical quantile of s_θ on calibration batch)

Solution 2: *End-to-End* learning over calibrated uncertainties



How can we train our uncertainty estimates using the task loss?

Solution 2: *End-to-End* learning over calibrated uncertainties

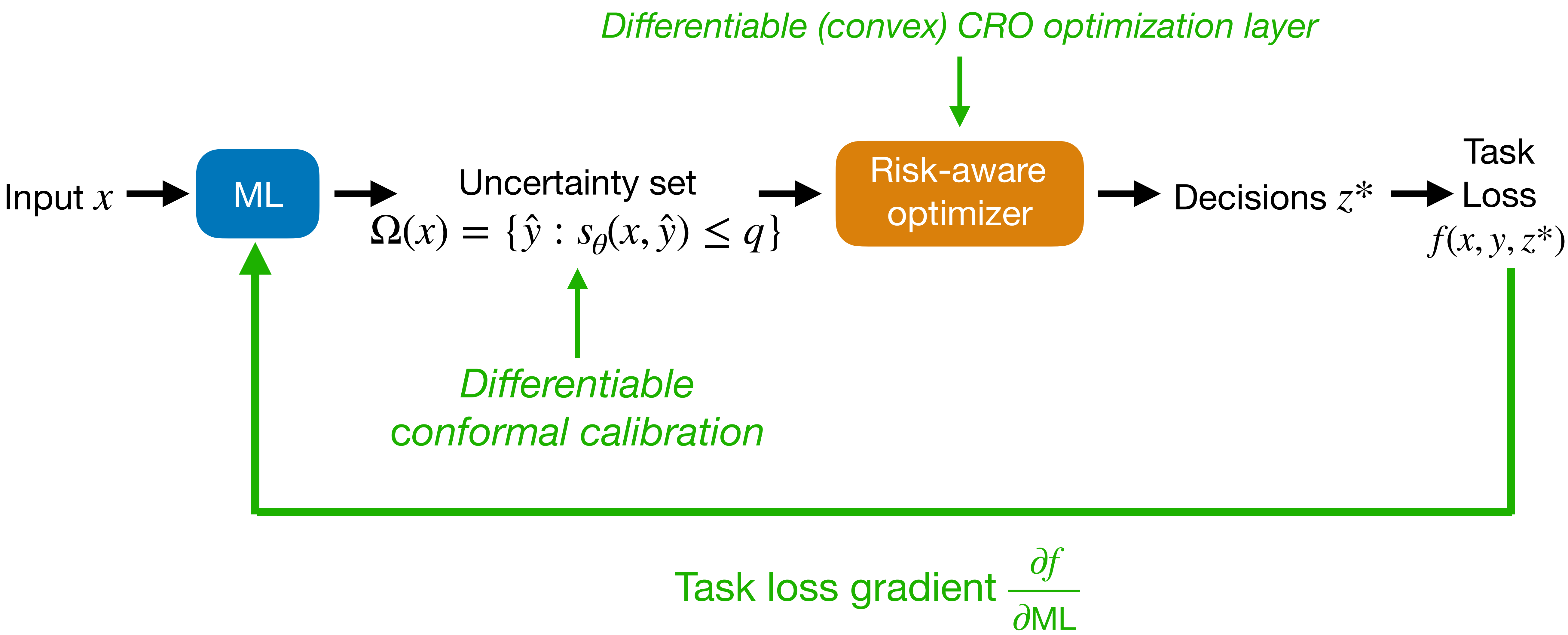


How can we train our uncertainty estimates using the task loss?

Insight: For PICNN (and box, ellipsoidal uncertainty), CRO is convex

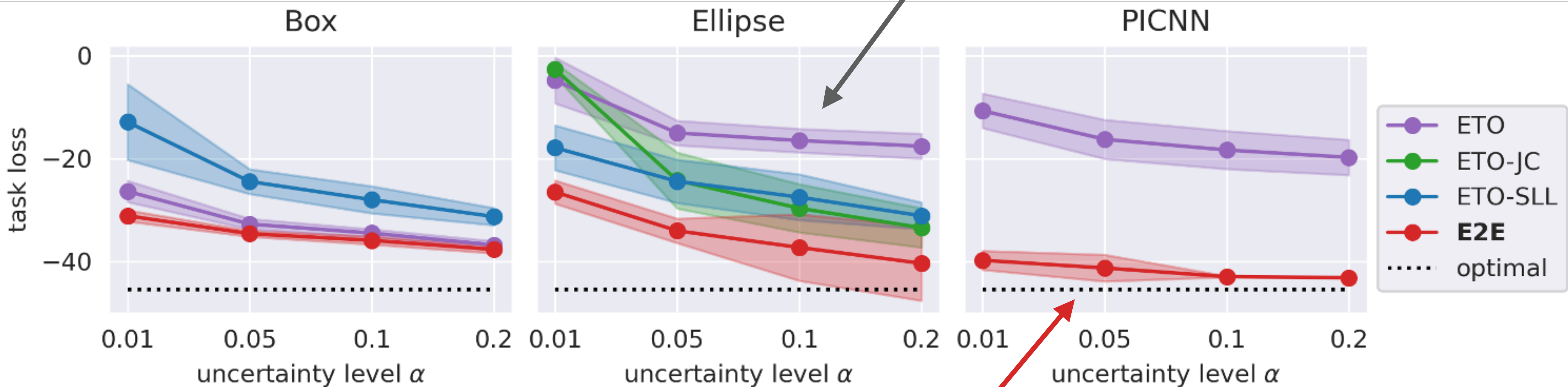
* Agrawal et al., 2019

Solution 2: *End-to-End* learning over calibrated uncertainties



Experiments on energy storage operation problem

Other “Estimate, then Optimize” (ETO) methods



Our End-to-End (E2E) method

End-to-End learning of UQ improves performance
for risk-aware energy storage operation…

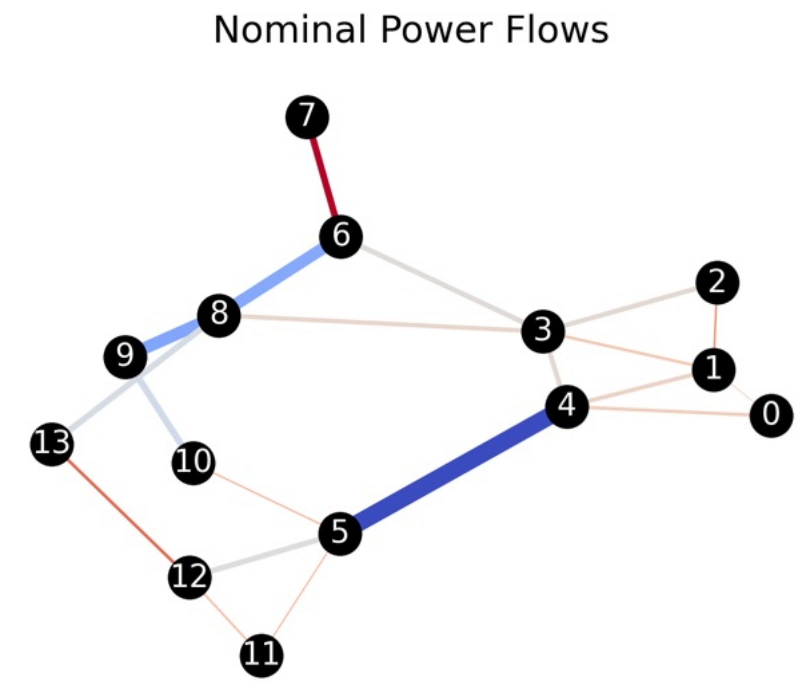
…what other notions of *reliability* can we learn
and enforce end-to-end?

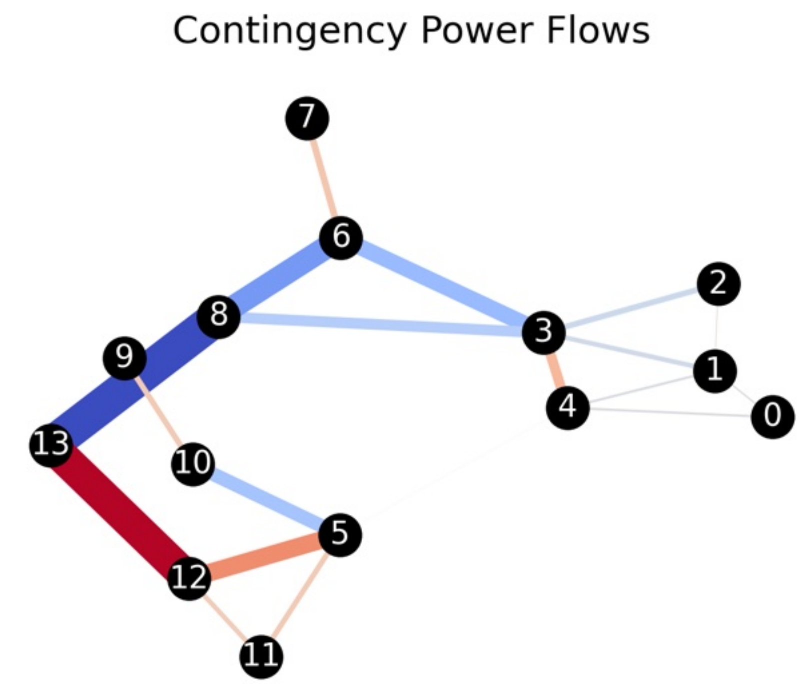
New! Recent work (just accepted at NeurIPS ’25) develops a methodology
for controlling *tail risks* like CVaR end-to-end

Last idea today: controlling *false negative rate* for contingency screening

Motivating example:
Power grid contingency screening

- **Contingency** - failure of (one or more) grid assets
- Line outage(s) redistribute power flows, can cause more outages





Motivating example: Power grid contingency screening

- **Contingency** - failure of (one or more) grid assets
- Line outage(s) redistribute power flows, can cause more outages
- Large-scale blackouts cause billions in economic losses

THE BLACKOUT: THE INVESTIGATION; OHIO LINES FAILED BEFORE BLACKOUT

Share full article

RICHARD PÉREZ-PEÑA

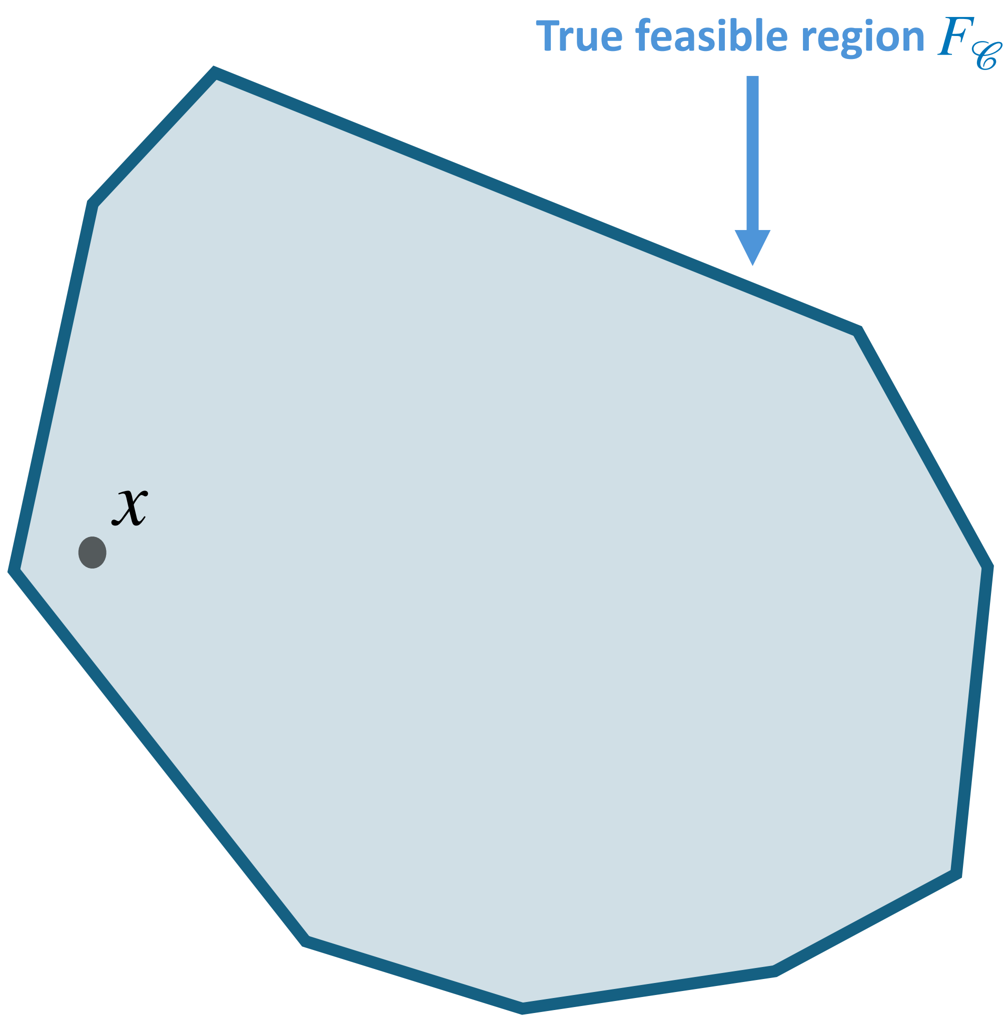
Aug. 17, 2003

The events that led to Thursday's blackout began when several high-voltage transmission lines near Cleveland failed, investigators said yesterday. The utility that owns the lines said an alarm that should have alerted controllers to the shutdowns also failed.

Contingency screening: formal problem description

- Fix a set $\mathscr{C}$ of contingencies
- Checking if dispatch x is feasible for all contingencies $\mathscr{C}$ is a polyhedral containment problem:*

$$x \in F_{\mathscr{C}} := \{y \in \mathbb{R}^n : A_c y \leq b \quad \forall c \in \mathscr{C}\}$$



Challenge: $\mathscr{C}$ is typically very large - ($\Omega(N^k)$ for all "N - k" contingencies)

Cannot check these all at deployment time!

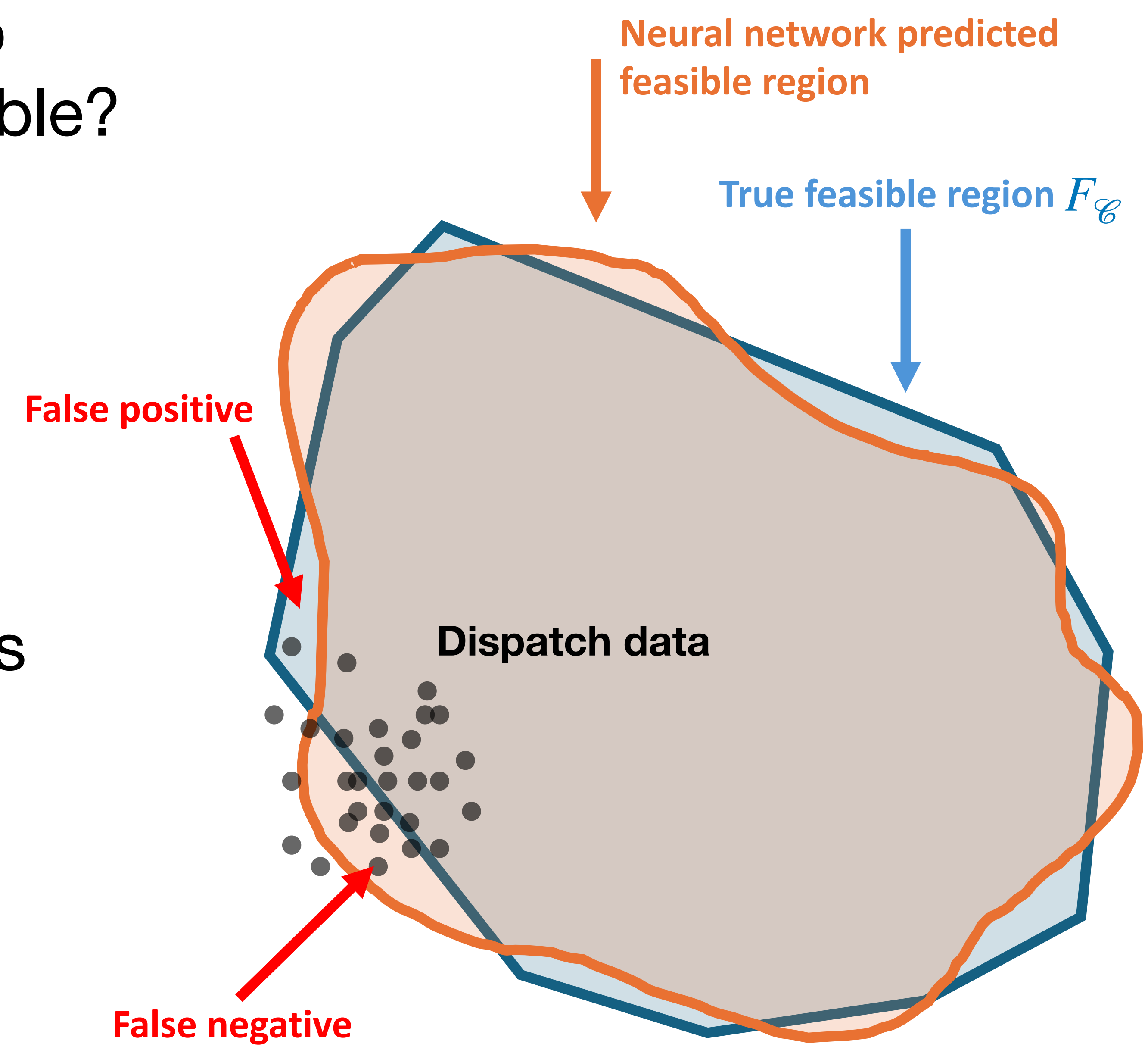
* in the DC-OPF model

Can machine learning help contingency screening?

What if we train a neural network to classify dispatches into feasible/infeasible?

Problems:

- NN predicted feasible set is generally nonconvex
- Can't find false positives/negatives
- **Need to avoid false negatives**

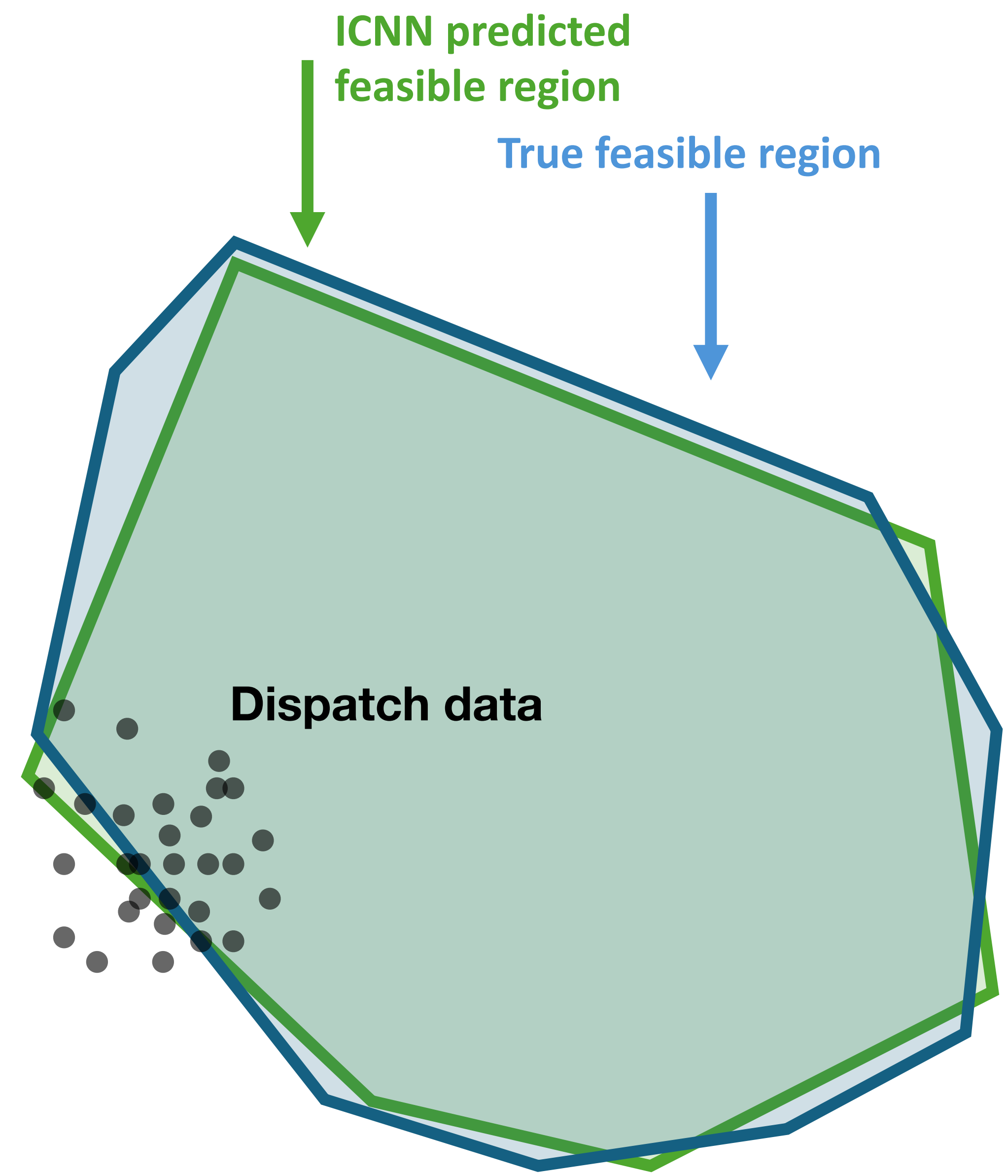


Provably *reliable* contingency screening with Input-Convex Neural Networks (ICNNs)

- ICNNs can approximate *general* convex functions
- It is tractable to transform a given ICNN into a **reliable** one with **0 False Negative Rate (FNR)**:

Theorem. Given an ICNN classifier f^{ICNN}, by solving a collection of linear programs, we can scale its parameters to achieve 0 FNR.

Christianson et al., L4DC '25

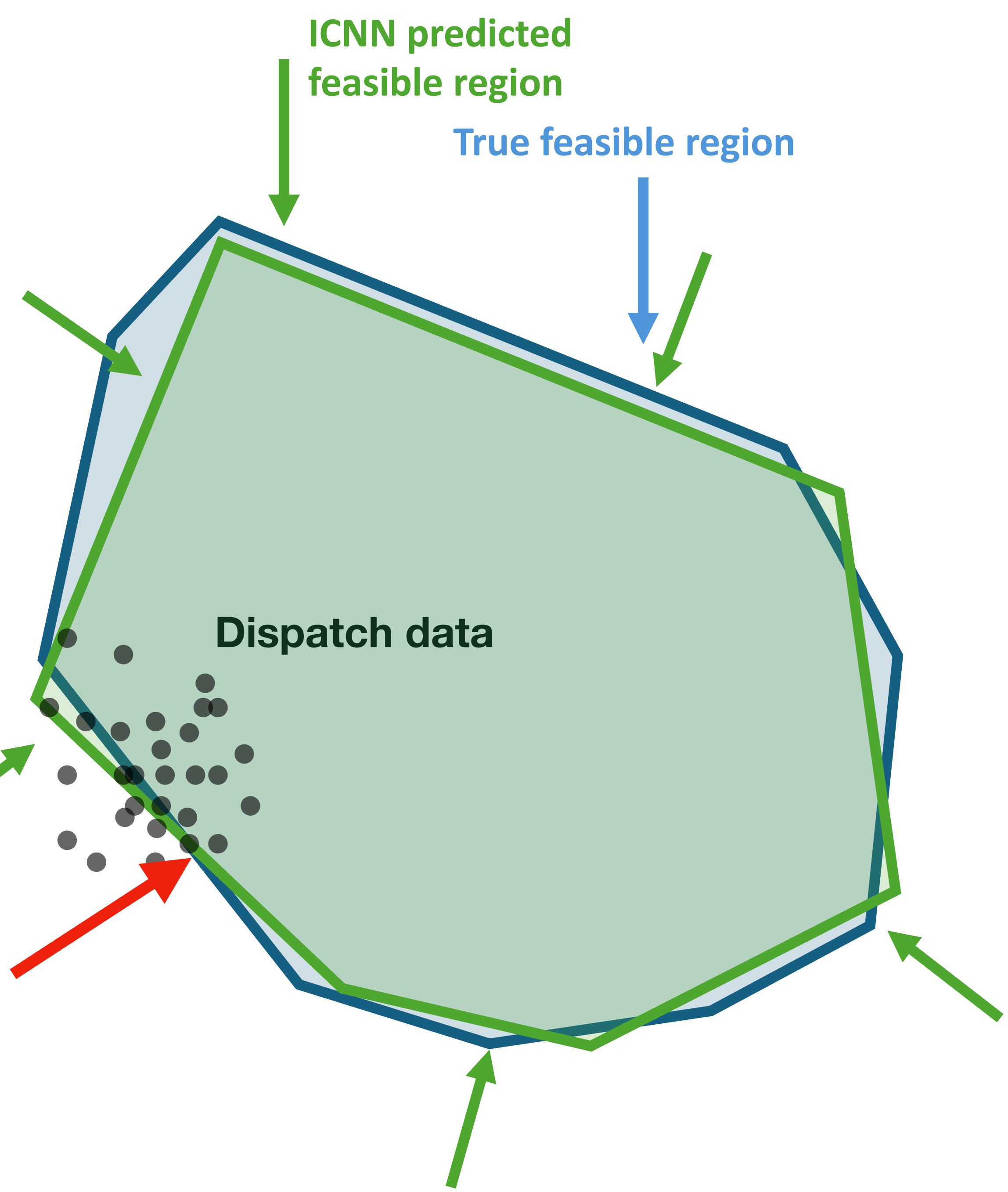


Provably *reliable* contingency screening with Input-Convex Neural Networks (ICNNs)

- ICNNs can approximate *general* convex functions
- It is tractable to transform a given ICNN into a **reliable** one with **0 False Negative Rate (FNR)**:

Theorem. Given an ICNN classifier f^{ICNN}, by solving a collection of linear programs, we can scale its parameters to achieve 0 FNR.

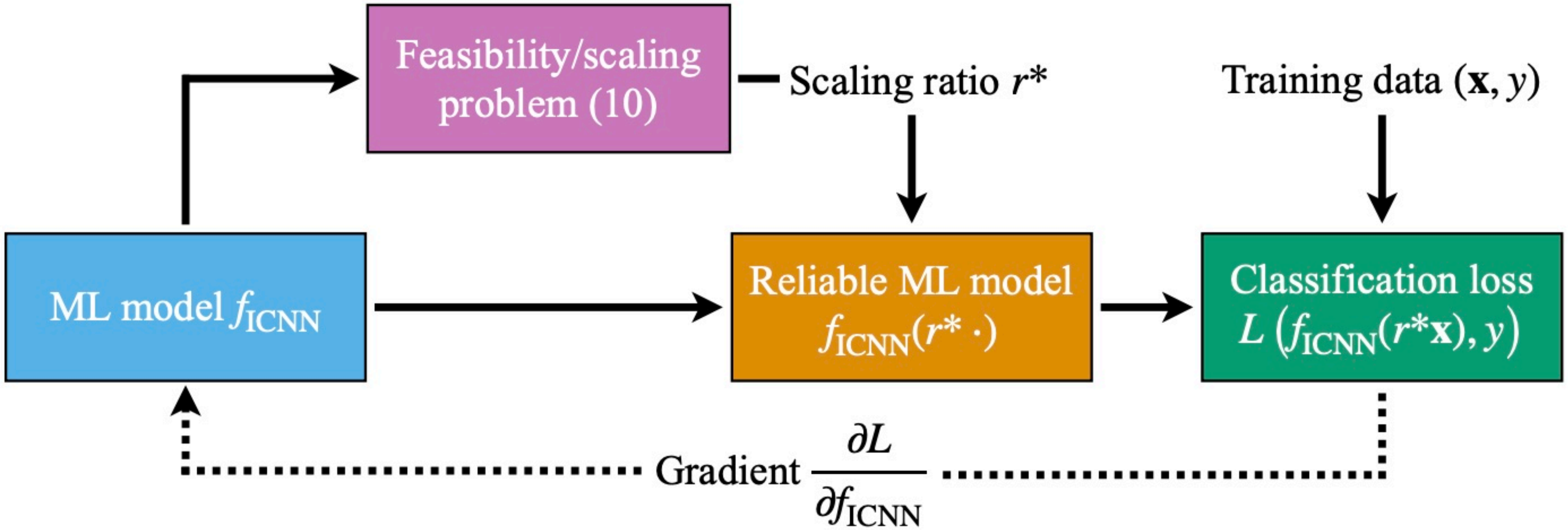
Christianson et al., L4DC '25



More false positives - can we decrease conservativeness while preserving 0 FNR?

An “End-to-End” approach to training reliable ICNN classifiers

- ICNN scaling results from a collection of linear programs
- Compute in a fully differentiable way via, e.g., CVXPYLayers*
- Enforce 0 FNR **differentiably, at each step of training**



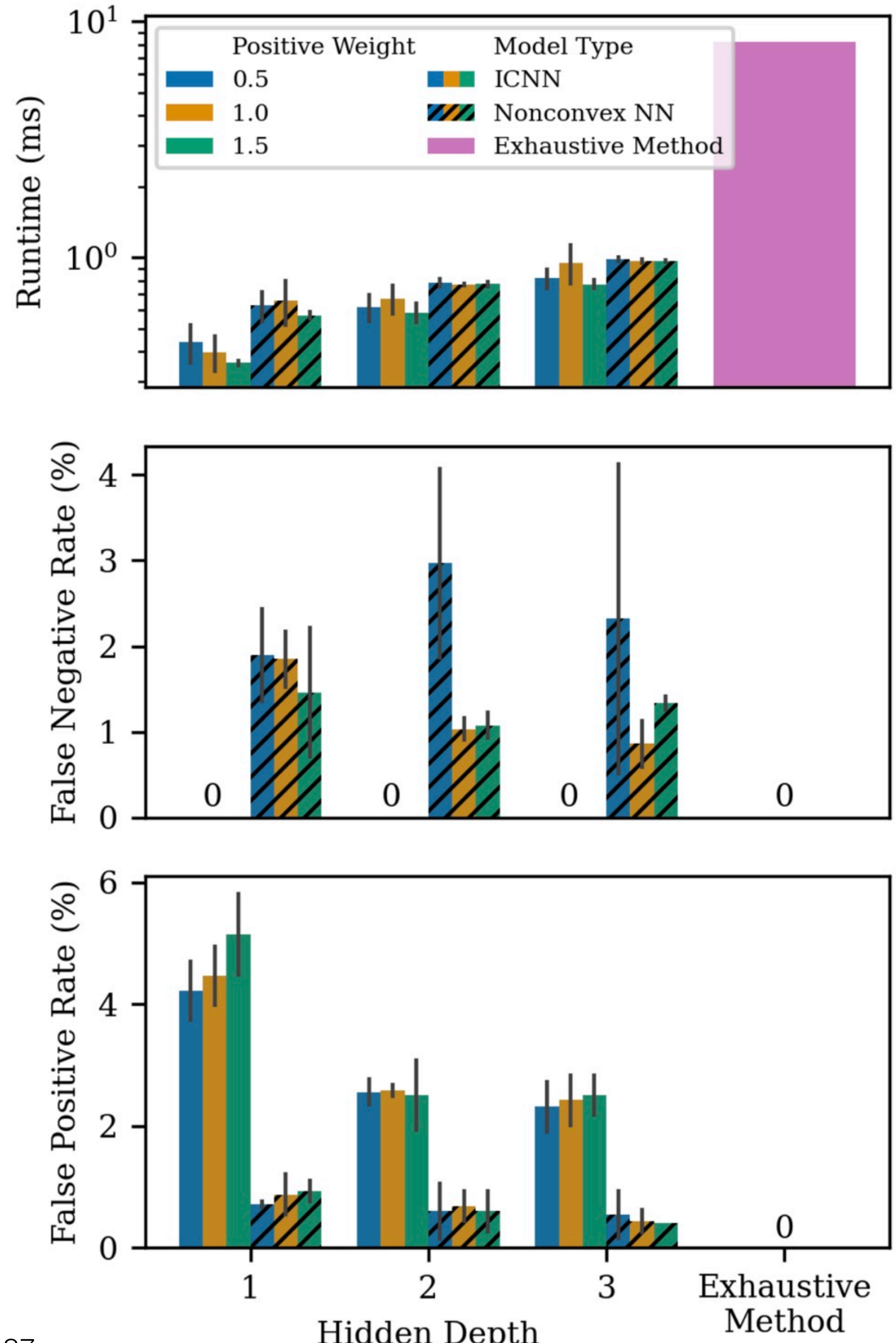
* Agrawal et al., 2019

Experiments: Contingency Screening

Case study on “N - 2” contingency screening, IEEE 39-bus system

ICNN achieves:

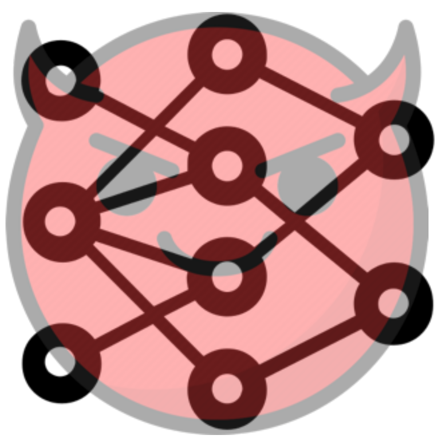
- 10-20x speedup over exhaustive approach
- Provably 0 FNR
- Small FPR



Concluding thoughts

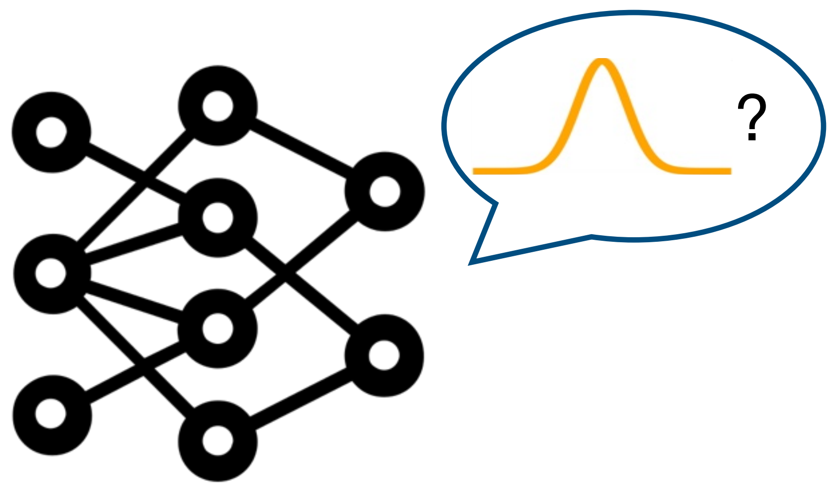
More control over guarantees

"Black-box"



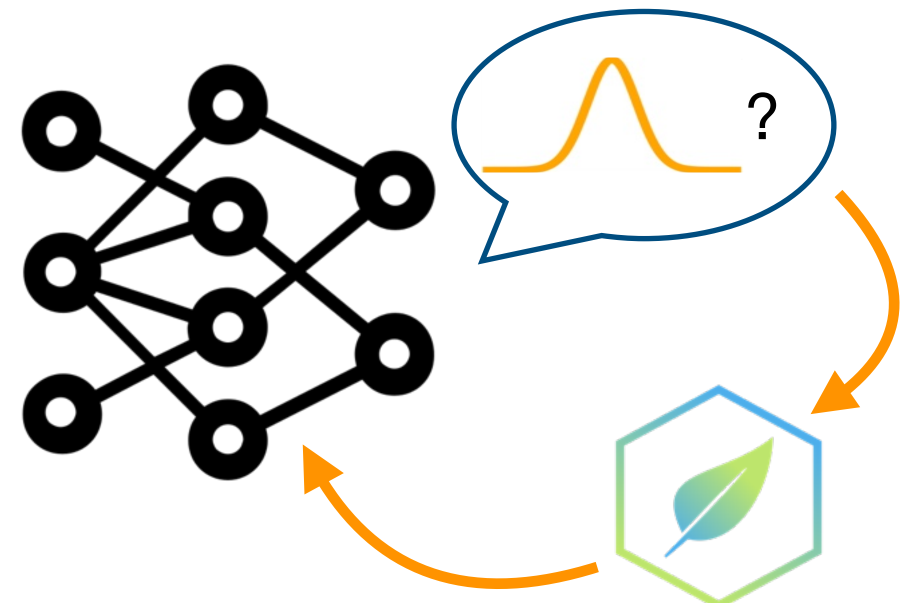
No guarantees on model performance

"Grey-box"



Model provides *some* uncertainty estimates

"End-to-End"



Trained for specific task with reliability (e.g., UQ) in-the-loop

Concluding thoughts

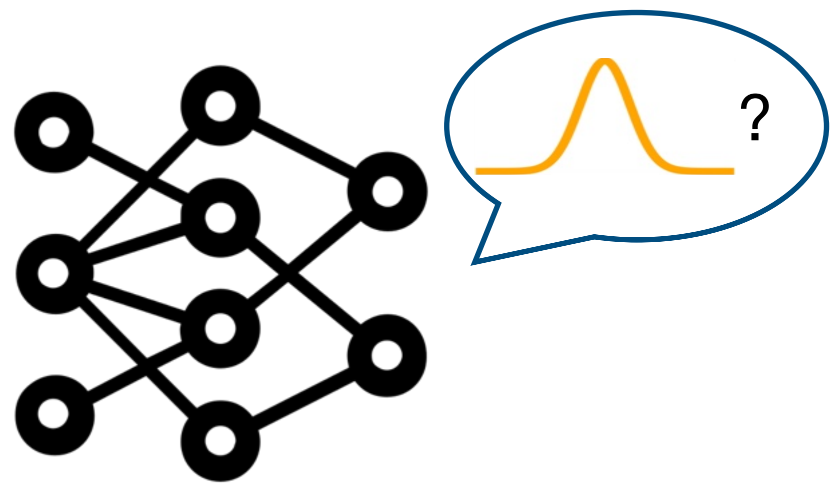
Worst-case guarantees on “black-box” AI/ML

More control over guarantees

“Black-box”

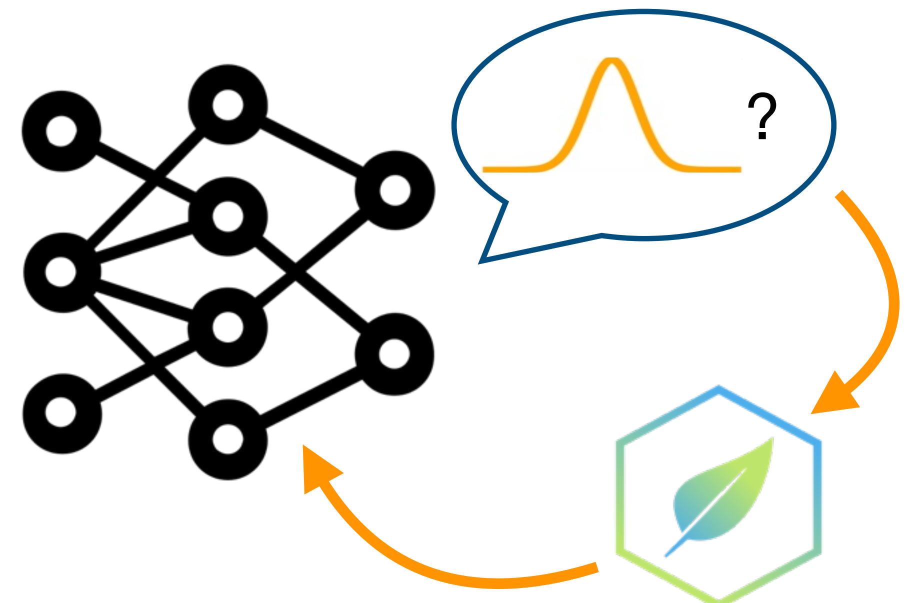
No guarantees on model performance

“Grey-box”



Model provides *some* uncertainty estimates

“End-to-End”



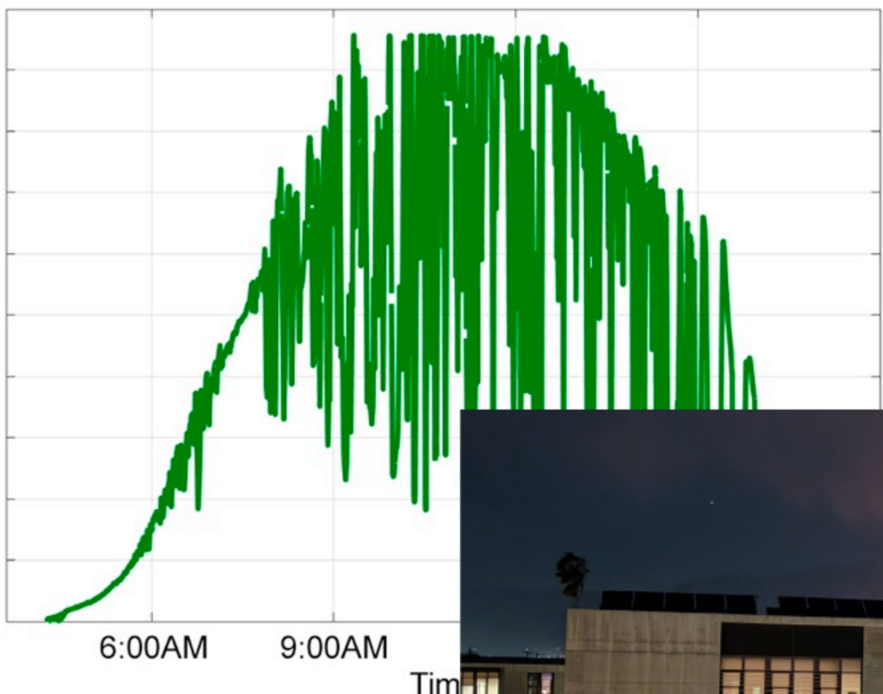
Trained for specific task with reliability (e.g., UQ) in-the-loop

End-to-End learning with UQ + reliability

Right approach depends on problem + operational needs

Growing challenges across energy and sustainability:

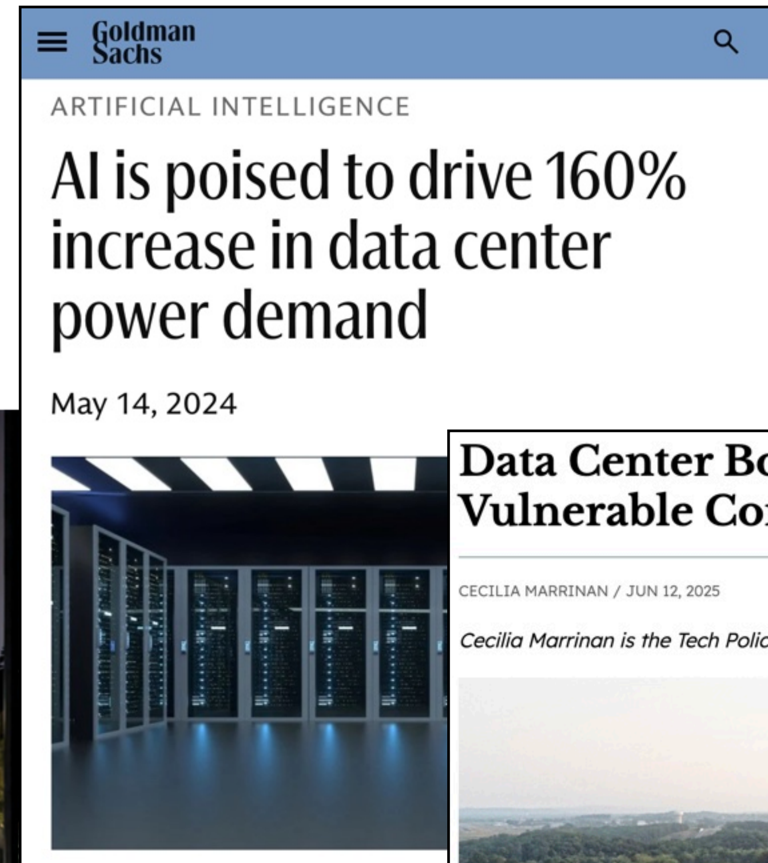
Solar Generation



https://eprijournal.com/can-variable-s

The New York Times

Out-of-Control Fires Devastate Los Angeles Area



ARTIFICIAL INTELLIGENCE

AI is poised to drive 160% increase in data center power demand

May 14, 2024

Data Center Boom Risks Health of Already Vulnerable Communities

CECILIA MARRINAN / JUN 12, 2025

Cecilia Marrinan is the Tech Policy Associate at the Kapor Foundation.

Aerial view of data centers being built in Leesburg, VA. Credit: Gerville/2024

Growing challenges across energy and sustainability motivate new frontiers in algorithms for reliable AI/ML:

- New AI/ML architectures/training methods for ***provable* constraint satisfaction** (hard safety guarantees)
- **ML for optimization** - better solutions for large-scale, nonconvex problems in energy system planning + operation
- Better **forecast calibration + UQ** for complex, multifaceted uncertainties (demand growth, electrification, asset failures, …)
- Learning + optimization for **multi-objective problems** (economic cost, carbon emissions, public health impacts, …)

⋮

Upcoming move…

Caltech

PhD 2020-25

Stanford | ENERGY

Energy Fellow 2025-26

JOHNS HOPKINS UNIVERSITY

Assistant Professor of CS, 2026-

Data Science and AI Institute

Energy at Hopkins

I am hiring **PhD students/postdocs** to start at JHU in **Fall 2026**!
Reach out if interested: **christianson@jhu.edu**

Thank you! Questions?

References

Christianson, Handina, Wierman. *Chasing convex bodies and functions with black-box advice.* COLT 2022.

Christianson, Shen, Wierman. *Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems.* AISTATS 2023.

Christianson et al. *Robust Machine-Learned Algorithms for Efficient Grid Operation*. Environmental Data Science 2025.

Yeh, Li, Datta, Arroyo, **Christianson**, et al. *SustainGym: Reinforcement learning environments for sustainable energy systems*. NeurIPS 2023.

Lechowicz, **Christianson**, et al. *Learning-Augmented Competitive Algorithms for Spatiotemporal Online Allocation with Deadline Constraints*. ACM SIGMETRICS 2025.

Yeh*, **Christianson***, et al. *End-to-End Conformal Calibration for Optimization Under Uncertainty*. Under review.

Christianson et al. *Fast and Reliable $N-k$ Contingency Screening with Input-Convex Neural Networks*. L4DC 2025.

My site:

Additional references

Christianson*, Werner*, et al. *Dispatch-aware planning for feasible power system operation*. PSCC 2022.

Werner*, **Christianson***, et al. *Pricing uncertainty in stochastic multi-stage electricity markets*. CDC 2023.

Rutten, **Christianson**, et al. *Smoothed online optimization with unreliable predictions*. ACM SIGMETRICS 2023.

Christianson et al. *Risk-Sensitive Online Algorithms*. COLT 2024.

Lechowicz, **Christianson**, et al. *The online pause and resume problem: Optimal algorithms and an application to carbon-aware load shifting*. ACM SIGMETRICS 2024.

Lechowicz, **Christianson**, et al. *Online Conversion with Switching Costs: Robust and Learning-augmented Algorithms*. ACM SIGMETRICS 2024.

Lechowicz, **Christianson**, et al. *Chasing Convex Functions with Long-term Constraints*. ICML 2024.

Sun, Huang, **Christianson**, et al. Online Algorithms with Uncertainty-Quantified Predictions. ICML 2024.

Chen, Lin, **Christianson**, et al. *SODA: An adaptive bitrate controller for consistent high-quality video streaming*. ACM SIGCOMM 2024.