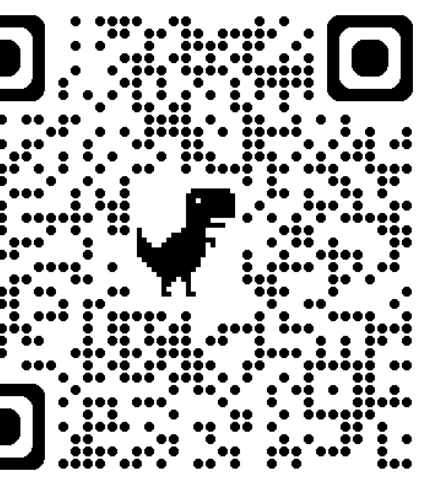




Motivating example: GPT models release timeline

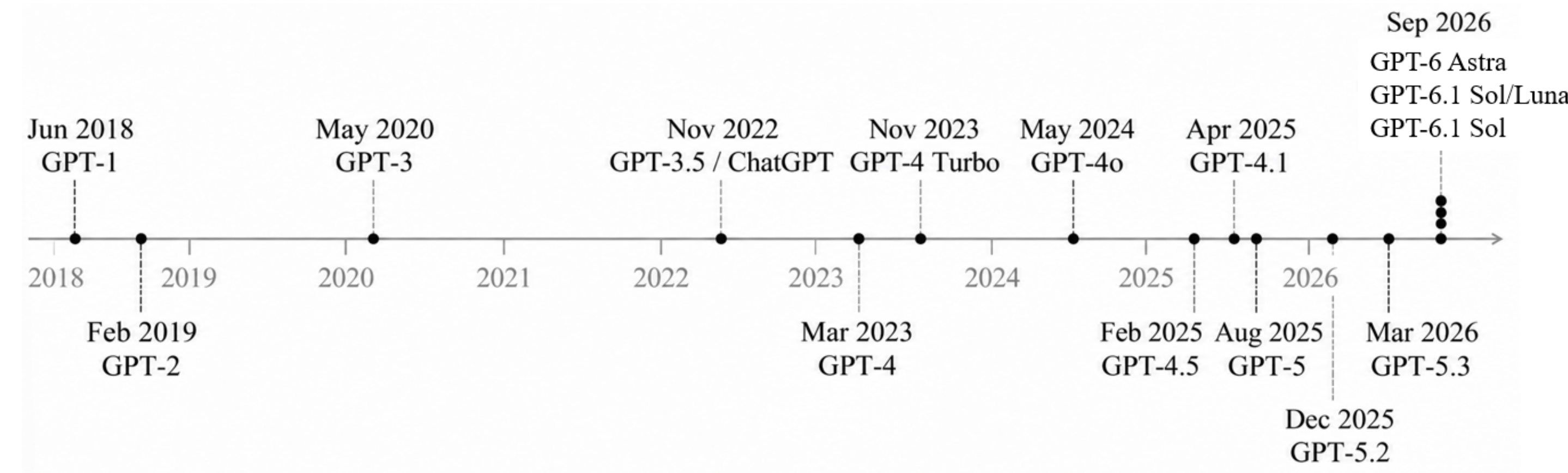


Figure 1: What factors determine the model update frequency?

Let π denote a trained model, and $V(\pi)$ be the expected value of model π .
e.g., GPT X.X e.g., Level of satisfaction with π

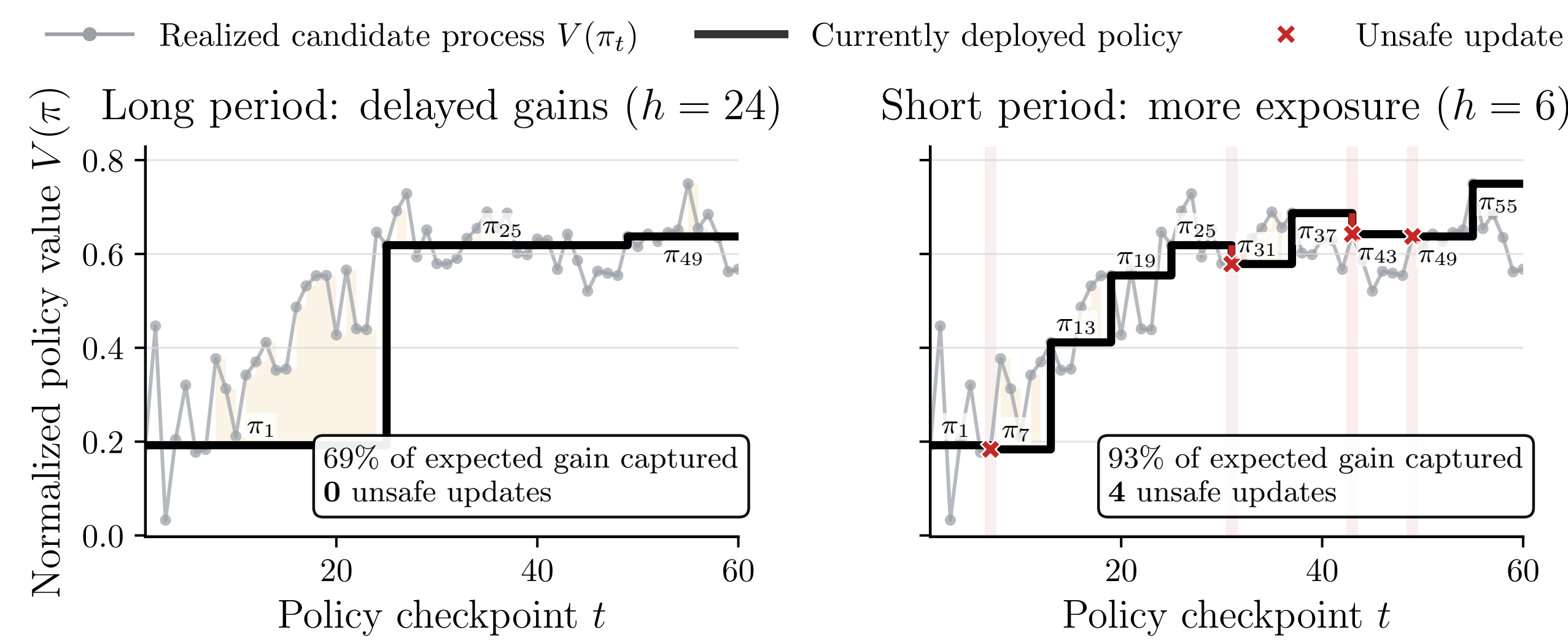


Figure 2: A key tradeoff is cumulative performance gain vs. safety

Research question: what is the optimal schedule of model updates?
a.k.a. meta-policy

Problem formulation: a constrained optimization problem

Setting: Consider an *online* learning setting with finite horizon T , and for each time t , a model π_t is trained from all observed historical data.

Definition: meta-policy

We define the meta-policy as a mapping from **calendar time** to **model version**:

$$m \in \mathcal{M} = \{ \{1, \dots, T\} \mapsto \{1, \dots, T\} : m(t) \leq T \}.$$

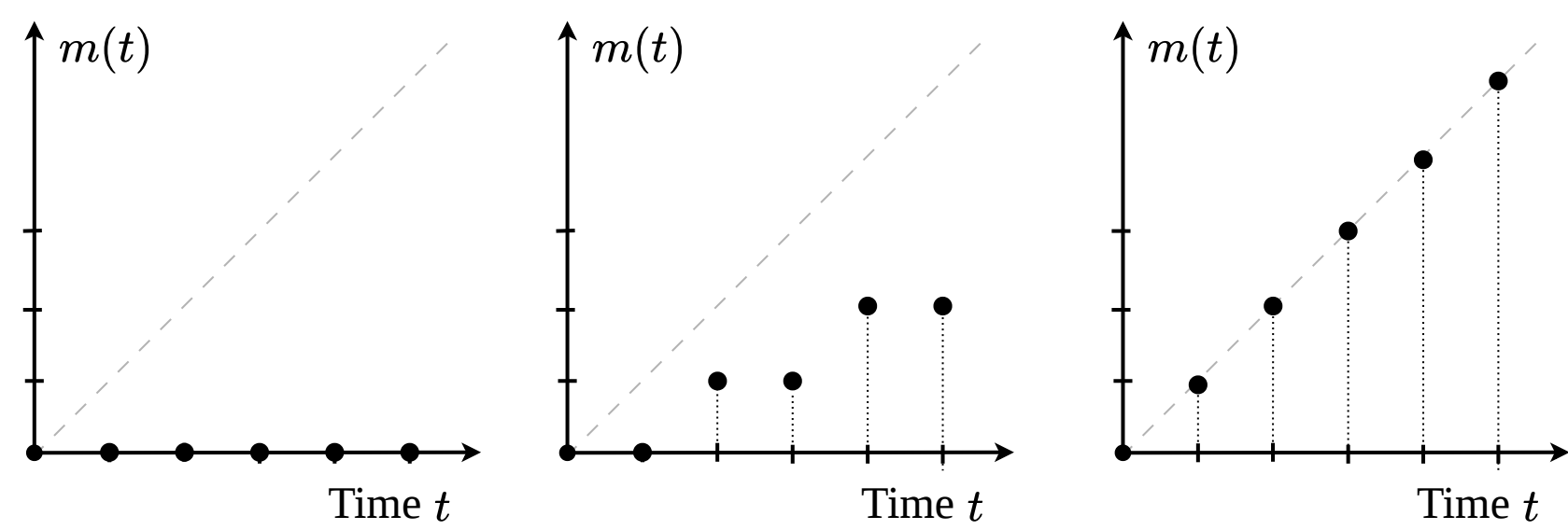


Figure 3: Some examples of meta-policies.

Objective: offline safe meta-policy design

$$\max_{m \in \mathcal{M}} \mathbb{E} \left[\frac{1}{T} \sum_{t=1}^T V(\pi_{m(t)}) \right] \quad \text{s.t.} \quad \mathbb{E} \left[\sum_{t=2}^T \mathbf{1} \{ V(\pi_{m(t)}) < V(\pi_{m(t-1)}) \} \right] \leq \varepsilon.$$

3. Computation: reformulation as a path selection problem

Properties of the meta-policy: There must be $m(t) \leq t$, but $m(t)$ need not be monotone.

Lemma 1: monotone meta-policy reduction

If $\mathbb{E}[V(\pi_i)] \leq \mathbb{E}[V(\pi_j)]$ for $i < j$, then an optimal schedule retains the incumbent or deploys the newest candidate, i.e., $m(t) \in \{m(t-1), t\}$. Therefore,

$$\{m(t) : t \in \{1, \dots, T\}\} \iff \{t_1, \dots, t_K\}.$$

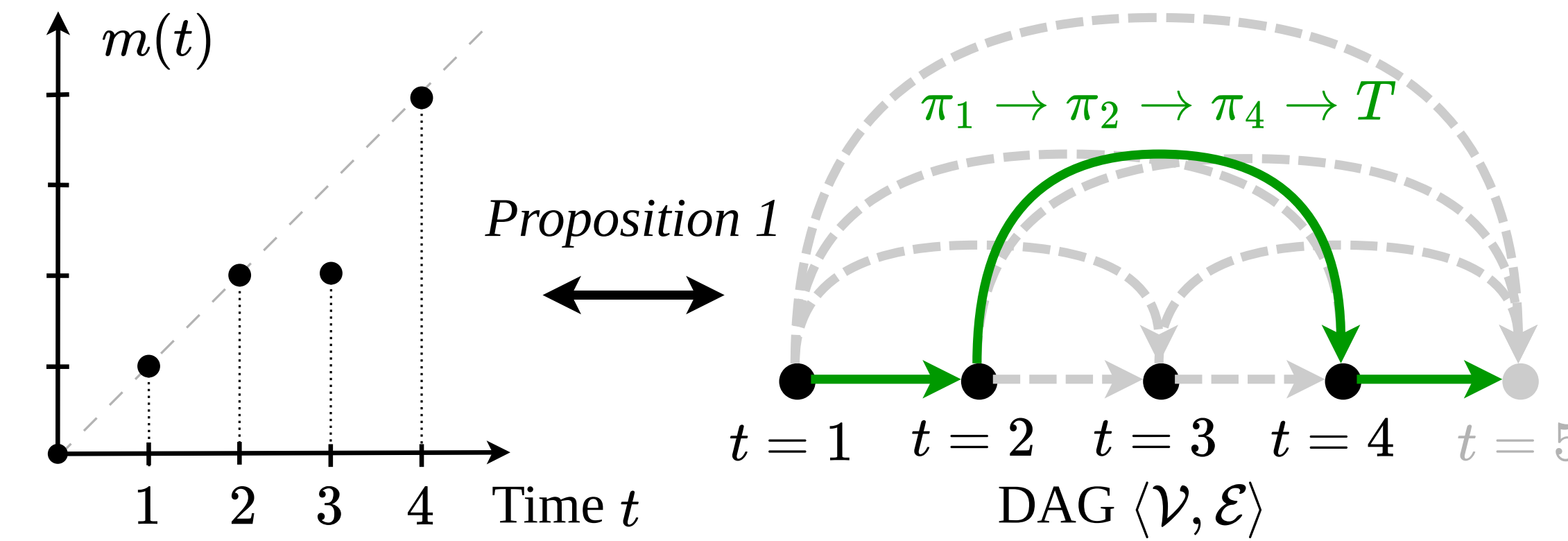


Figure 4: A resource-constrained shortest path problem.

Proposition 1: Resource Constrained Shortest Path Problem

Under the same condition as Lemma 1, the original optimization is equivalent to

$$\max_{p \in \mathcal{P}_{1,T+1}} \sum_{(i,j) \in p} \underbrace{(j-i)\mathbb{E}[V(\pi_i)]/T}_{R_{ij}} \quad \text{s.t.} \quad \sum_{(i,j) \in p} \underbrace{\mathbb{P}\{V(\pi_j) < V(\pi_i)\}}_{S_{ij}} \leq \varepsilon.$$

This can be solved via dynamic programming algorithms.

Theoretical analysis: leading-order optimality system

Assumption 2: Itô Diffusion

$$V(\pi_t) = \underbrace{V_0}_{\text{Const.}} + \underbrace{\frac{1}{T} \int_0^t \mu(\tau/T) d\tau}_{\text{Drift}} + \underbrace{\frac{1}{T} \int_0^t \sigma(\tau/T) dW_\tau}_{\text{Volatility}} \underbrace{BM}_{\text{BM}}.$$

Main Theorem

Under Assumptions 1 and 2, suppose $T \rightarrow +\infty$, denote $u = t/T$ and

$$L_T = \log \left(\frac{T}{\varepsilon} \right), \quad I(u) = \frac{\mu(u)^2}{2\sigma(u)^2},$$

and $A(\cdot), B, C(\cdot), D, E, F$ are unimportant constants or functions. Then, the

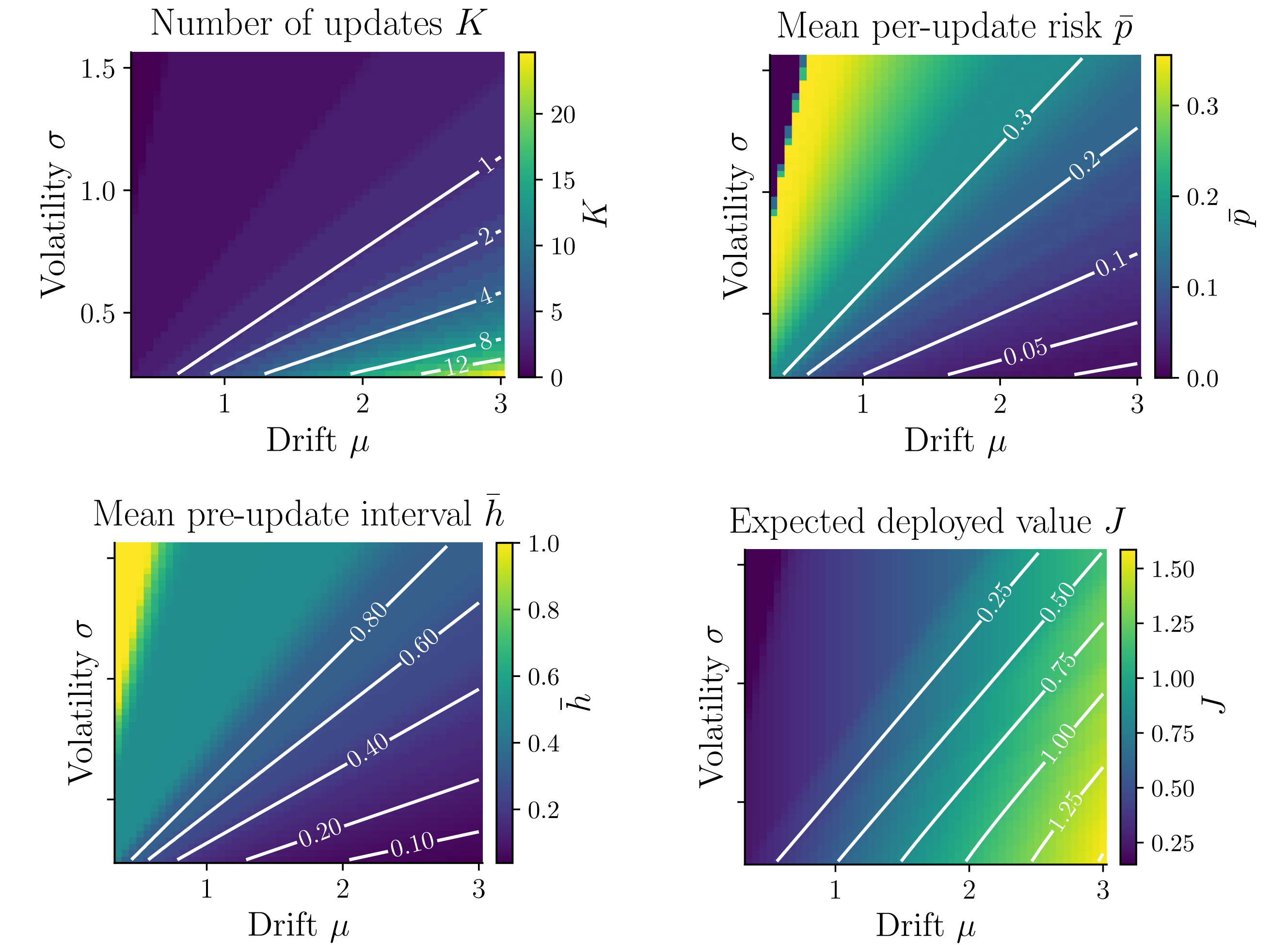
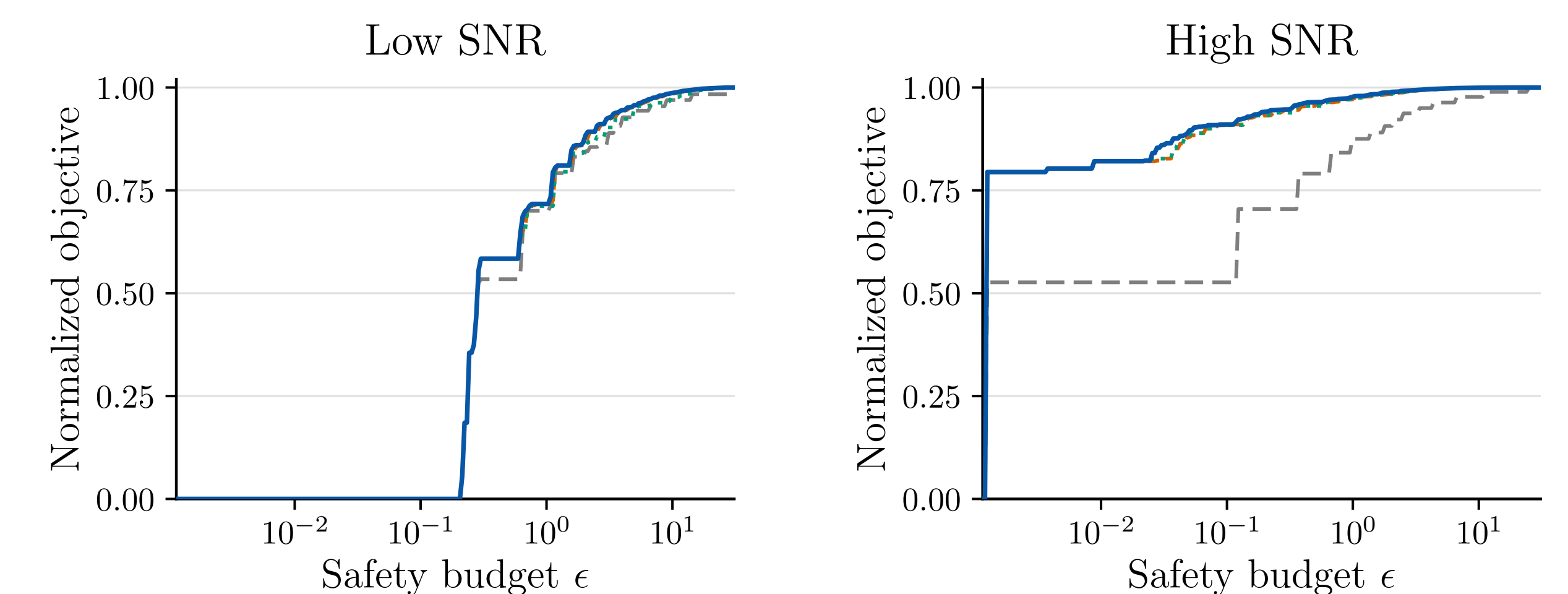
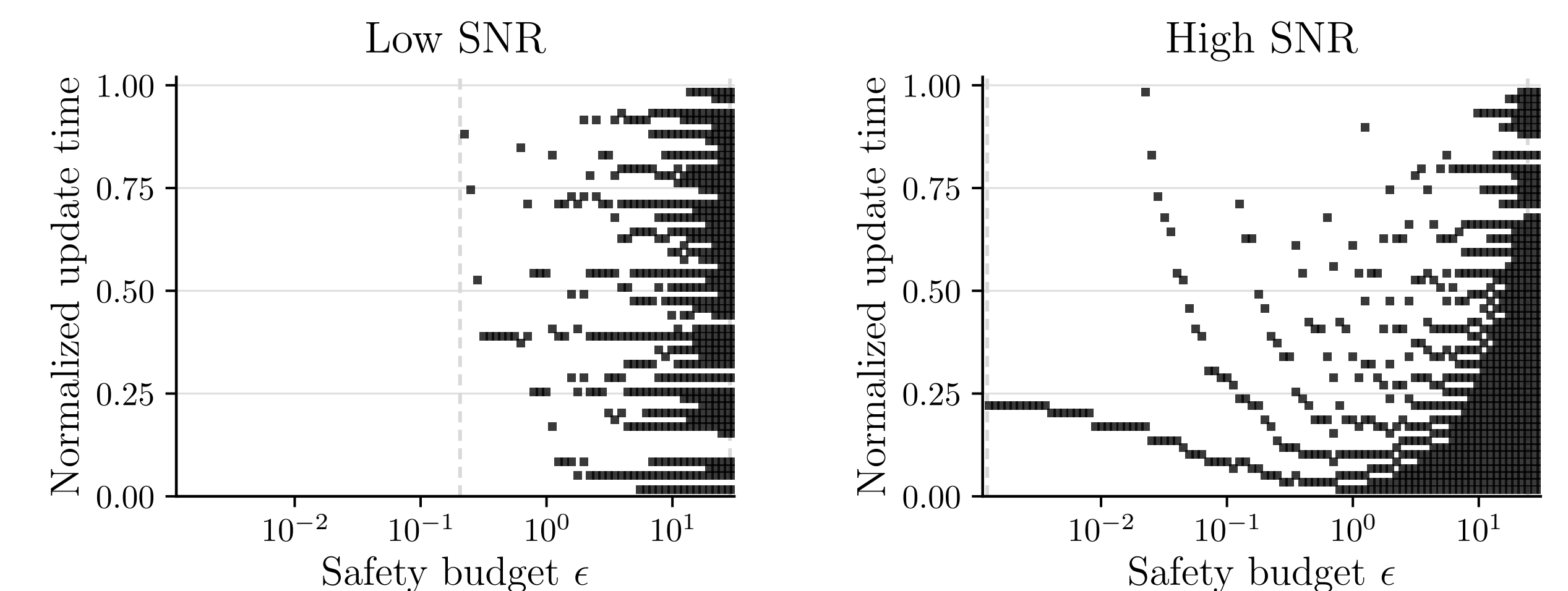
Quantity	Notation	Asymptotic expression
Update interval	$h_T^*(u)$	$\frac{L_T}{I(u)} + \frac{1}{I(u)} \left[-\frac{3}{2} \log L_T + C(u) \right] + o(1)$
Update risk	$p_T^*(u)$	$\frac{\varepsilon L_T \mu(u)}{TB I(u)^2} \{1 + o(1)\}$
Number of updates	K_T^*	$\frac{TD}{L_T} + \frac{T}{L_T^2} \left[\frac{3}{2} D \log L_T - E \right] + o \left(\frac{T}{L_T^2} \right)$
Regret	$\mathcal{R}_T^*(\varepsilon)$	$\frac{L_T B}{2T} + \frac{1}{2T} \left[-\frac{3}{2} B \log L_T + F \right] + o(T^{-1})$

Red: leading-order asymptotic behavior, *blue:* local learning dynamics.

Remark: The derivation relies on proving a continuous approximation when $T \rightarrow +\infty$:

$$\min_{h(\cdot)} \int_0^1 \mu(u) h(u) du \quad \text{s.t.} \quad \int_0^1 \frac{\Phi(-\mu(u)\sqrt{h(u)}/\sigma(u))}{h(u)} du \leq \frac{\varepsilon}{T}.$$

Experiment: theory evaluation, performance, and patterns

Figure 6: Homogeneous value processes. A 50×50 grid with $T = 30$, $\varepsilon = 0.35$, and $\Delta = 0.005$. White contours show theory. Higher SNR supports more updates and shorter waits.Figure 6: Safety-value frontiers. Budget ε is on a logarithmic scale. The normalized objective equals 0 for never updating and 1 for always updating.Figure 7: Selected deployment times. Each dot marks an update at normalized time t/T for a given budget. Dashed lines identify the first budget that permits an update.

Summary: Three Key Findings

- Meta-policy learning $\xrightarrow{\text{Asm. 1}}$ path selection problem.
- Local dynamics scale directly with signal-to-noise ratio.
- Diminishing marginal cost of additional safety.