WHEN IS ONE READING ENOUGH? DECISION-RELEVANT AMBIGUITY IN CONSTRAINED REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Safety rules for AI agents, online services, and industrial equipment are usually written in natural language, yet a learned controller can obey a rule only once it is a precise cost that training can measure. One sentence, such as "drain a server when load stays high," often admits several plausible executable readings, each a different constraint. We cast this as constrained reinforcement learning (RL) with candidate cost functions and ask whether the controller can train under one reading or must obey them all. Obeying all is safe but can give up performance, while picking one can yield a controller another reading forbids. We propose ARROW (**A**mbiguity **R**elevance for **R**einforcement Learning **O**ver **W**ritten Rules), a test that runs before training and searches for a controller nearly optimal under one reading yet violating another. If none exists, training under that reading is safe for every candidate reading. Otherwise ARROW keeps the competing readings or asks the operator targeted questions, re-testing after each answer. The test is exact when the dynamics are known, gives a high-confidence certificate from a fixed log, and works with any constrained learner. On agent-service tasks, training under the reading ARROW certifies raises return by a per-learner median of 15.0 to 37.3 percentage points of the best return that obeys every reading, across six learners. On service-monitoring rules, four questions settle what to enforce for 99% of ambiguous rules, usually before the intended reading is known. On 182 real ventilator, robotics, and cyber-physical requirements, 87.4% keep several distinct readings after redundant ones are removed, so the question is routine.

1 INTRODUCTION

Written safety rules are ambiguous. Safety rules that govern learned controllers are usually written in natural language, and one rule can often be made executable in several reasonable ways. Consider a controller that manages computing servers, for example in a Kubernetes cluster, under the monitoring rule "drain a node when load stays high." "Stays high" could mean five minutes above a threshold, ten minutes, or a high average over a window. Similar rules decide when to service equipment, which configurations to admit, and how often an AI agent may call an external service. Recent work translates such rules into constraints that learning systems can enforce (Yang et al., 2021) or into formal specifications (Mendoza et al., 2026). We call each executable translation a *reading* of the rule.

The choice of reading changes which controllers are safe. We study this in constrained reinforcement learning (RL), where a controller, called a *policy*, maps the current state of the system to an action, the reward it collects over time is its *return*, and a cost limits unsafe behavior. In the server example the state records current and recent load, the action is to keep running or drain the node, and return measures service delivered. Each reading defines its own cost: running after seven minutes of high load violates the five-minute reading but not the ten-minute one. Obeying every reading at once is safe but can forbid good policies and lower return. Training under one reading keeps that return but may produce a policy that another reading forbids. We therefore ask whether one reading is *sufficient*, meaning that every policy whose return is within a chosen tolerance of the best under that reading also satisfies all other readings. Comparing the best returns alone does not

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

answer this, because two readings can share the best return while allowing different near-optimal policies (Section 2).

Existing methods do not ask this question. Current approaches either commit to one reading before learning or protect against every supplied reading. Language-based safe RL infers or generates constraints from language and optimizes under them (Yang et al., 2021; Xue et al., 2026), and offline safe RL, which learns from a fixed log of past data, assumes the safety cost is already specified (Chemingui et al., 2025a;b; Liu et al., 2025b). Robust constrained RL protects against every member of a supplied set of constraints or dynamics models (Varagapriya et al., 2022; Wang et al., 2022; Ma et al., 2025; Ganguly et al., 2025), and requirement-formalization methods seek the one intended specification (Cosler et al., 2023; Mendoza et al., 2026). Related work studies decision-relevant uncertainty in rewards or hypotheses (Hadfield-Menell et al., 2017; Chen et al., 2015; Skalse et al., 2023) and infers unknown constraints from demonstrations (Malik et al., 2021; Papadimitriou et al., 2022; Lindner et al., 2024; Chua et al., 2025). To our knowledge, none tests whether one reading of a written rule is sufficient for training (Appendix A).

Our method tests sufficiency before training. We propose ARROW (**A**mbiguity **R**elevance for **R**einforcement Learning **O**ver **W**ritten Rules), a decision step that runs before any constrained learner. For a candidate reading, ARROW searches for a near-optimal policy under it that violates another reading. If none exists, the reading is sufficient and the learner trains under it alone. Otherwise the search returns such a policy as a counterexample, and ARROW either keeps the competing readings or asks the operator targeted questions, re-testing after each answer and stopping as soon as one remaining reading is sufficient. The test is exact when the dynamics are known and gives a high-confidence certificate from a fixed log when they are not. On the checked route (Section 3.3) the learner is unchanged and its returned policy is checked before deployment.

ARROW preserves return and saves operator effort in practice. On agent-service tasks with six constrained learners, training under the reading that ARROW certifies raises return by a per-learner median of 15.0–37.3 percentage points over training on the combined cost of all readings, and no unsafe policy passes the pre-deployment check. On service-monitoring rules, four targeted questions settle what the learner must enforce for 99% of the ambiguous rules, mostly before the intended reading is identified. On 182 real ventilator, robotics, and cyber-physical requirements translated by two language models, most requirements keep several distinct readings, so the decision is needed in practice. Our contributions are:

- **A decision problem for ambiguous safety rules.** We formulate when one reading is sufficient for constrained learning, separating ambiguity that changes which policies a learner may return from ambiguity that does not (Section 2).
- **An exact test with fixed-data guarantees.** ARROW decides sufficiency exactly with known dynamics and returns a counterexample when it fails. The *sufficiency radius* states how suboptimal a learner may be before another reading matters. From fixed data, ARROW certifies sufficiency with high confidence, and a lower bound shows how much data any sound certificate needs (Sections 3.1, 4.1 and 4.2).
- **Targeted clarification and safe deployment.** ARROW asks only questions that can change what the learner must enforce and stops early. It removes readings already enforced by others without changing the safe policies, and we bound the probability of deploying an unsafe policy (Sections 3.2, 4.3 and 4.4).
- **Evaluation.** We study service monitoring, admission control, equipment maintenance, and agent-service tasks, with six constrained learners on the agent-service benchmark, plus real requirements from three domains, varying budgets, policy classes, and data sizes (Section 5 and Appendix D).

2 PROBLEM FORMULATION

This section defines when training under one reading of a written rule is as safe as enforcing all of its readings. We work in constrained RL, where the learned policy must keep safety violations within an allowed limit (Altman, 1999; García & Fernández, 2015), the form in which offline safe-RL

learners take their constraints (Liu et al., 2024; Xu et al., 2022; Lee et al., 2022). Here the constraint comes from a written rule, and different plausible readings of that rule define violations differently.

2.1 THE LEARNING PROBLEM

States, actions, and policies. We model the system as a finite Markov decision process (MDP) $M = (S, A, P, r, \mu_0)$, the standard model of repeated decisions in which the next situation depends only on the current situation and action. Here S is the finite set of situations, called states, and A is the finite set of actions; $s \in S$ denotes a state and $a \in A$ an action. The dynamics $P(\cdot \mid s, a)$, also called the transition kernel, give the probability distribution of the next state after taking action a in state s. The reward $r(s, a) \geq 0$ is the one-step benefit of taking a in s, and μ_0 is the distribution of the state in which an episode, one run of the system, begins. A policy π specifies a probability distribution $\pi(\cdot \mid s)$ over A in each state s; s_t and a_t denote the state and action at step t. The learner, the algorithm that trains the policy, chooses it from a class Π, part of the problem because different classes express different behaviors.

Return. A policy is evaluated by its normalized discounted *return* $J_r(\pi) = (1 - \gamma)\,\mathbb{E}_\pi\big[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\big]$, a weighted average reward in which later rewards count less. Here $\gamma \in (0, 1)$ is the discount factor and $\mathbb{E}_\pi$ is over the trajectory $s_0 \sim \mu_0$, $a_t \sim \pi(\cdot \mid s_t)$, $s_{t+1} \sim P(\cdot \mid s_t, a_t)$. Larger $J_r(\pi)$ is better. Problems that end after a fixed number of steps use the analogous finite sum.

From a written rule to a safety constraint. Each reading ψ turns the rule into a per-step cost, and a policy satisfies ψ when its discounted cost stays within a budget. For the monitoring rule, the state records the load level and how long it has been high, and the action is to keep running or intervene. A reading determines a nonnegative cost $c_\psi : S \times A \to \mathbb{R}_{\geq 0}$ marking the behavior it prohibits: the five-minute reading charges running once load has been high for five minutes, and the ten-minute reading only after ten. In admission control, the cost counts admitting a disallowed configuration, in maintenance running past the trigger, and in agent-service tasks making a call covered by the stated limit. Following π, the system incurs $c_\psi(s_t, a_t)$ at step t, and its normalized discounted cost $J_{c_\psi}(\pi) = (1 - \gamma)\,\mathbb{E}_\pi\big[\sum_{t=0}^{\infty} \gamma^t c_\psi(s_t, a_t)\big]$ uses the same discounting as return. With a cost of 1 at a violating step and 0 otherwise, $J_{c_\psi}(\pi)$ is the discounted fraction of time spent in behavior that ψ counts as a violation, such as running when ψ says to intervene. The application sets how much of this behavior is tolerable through a budget $d \geq 0$, and π satisfies ψ when $J_{c_\psi}(\pi) \leq d$. We write $\Pi_\psi(d) = \{\pi \in \Pi : J_{c_\psi}(\pi) \leq d\}$ for the policies that satisfy ψ and $V_\psi(d) = \sup_{\pi \in \Pi_\psi(d)} J_r(\pi)$ for their best return, the target of a learner trained under ψ alone.

Candidate readings. When the intended reading is unknown, the safe default is to enforce every plausible reading at once, which we call *full-set protection*. Let $U = \{\psi_1, \ldots, \psi_K\}$ be the K plausible readings of one rule, the *candidate set*. They share the states, actions, dynamics, and reward and differ only in their costs c_ψ, that is, in what counts as a violation. Full-set protection allows only $\Pi_U(d) = \{\pi \in \Pi : J_{c_\phi}(\pi) \leq d \text{ for every } \phi \in U\}$. It is always safe, but it can rule out good policies that only some readings forbid and so lower the return. Training under one reading ψ instead allows every policy in $\Pi_\psi(d) \supseteq \Pi_U(d)$.

2.2 WHEN IS ONE READING SUFFICIENT?

Goal. Our goal is to decide whether one reading can replace the whole candidate set in training, not to identify the correct reading. In monitoring, for example, we ask whether a controller trained under the five-minute reading alone can be trusted under the ten-minute reading as well. A reading is feasible when some policy satisfies it. We call $F^\varepsilon_\psi(d) = \{\pi \in \Pi_\psi(d) : J_r(\pi) \geq V_\psi(d) - \varepsilon\}$ the *acceptable policies* under ψ, those that satisfy ψ and whose return is within a tolerance $\varepsilon \geq 0$ of the best under ψ. We seek a feasible $\psi \in U$ with $F^\varepsilon_\psi(d) \subseteq \Pi_U(d)$, or determine that no such reading exists.

What the learner may return. Sufficiency must cover every policy a learner trained under ψ might return, not only one optimal policy. Several policies may attain the best return $V_\psi(d)$, as

when a monitoring controller can intervene at different moments for the same return, and practical learning may return a slightly suboptimal one. The set $F^{\varepsilon}_{\psi}(d)$ collects exactly these policies. With $\varepsilon = 0$ it contains only the optimal policies, and a larger ε also admits near-optimal ones.

One reading is sufficient. A feasible reading ψ is ε-*sufficient* when $F^{\varepsilon}_{\psi}(d) \subseteq \Pi_U(d)$, and if no candidate is, the ambiguity is *decision-relevant*. If a sufficient reading exists, the ambiguity does not affect which constraint the learner must be trained under. Otherwise every single-reading choice permits some acceptable policy rejected by another candidate, so the ambiguity changes what the learner must enforce; in monitoring, the controller must then be trained against both durations. Sufficiency can depend on the policy class Π, the budget d, and the tolerance ε, since all three determine which policies belong to $F^{\varepsilon}_{\psi}(d)$ and to $\Pi_U(d)$.

3 Proposed Method: Arrow

Arrow decides before training whether one plausible reading of an ambiguous safety rule is sufficient or whether several readings must still be enforced. Without this decision, training under one reading can return a policy that another reading rejects, and enforcing every reading can give up return that no reading requires.

3.1 Deciding whether one reading is sufficient

Arrow tests a reading by its *ambiguity margin*, the largest amount by which an acceptable policy under that reading can exceed the budget of a competing reading. Let $C \subseteq U$ be the readings still under consideration, initially $C = U$, and $\Pi_C(d)$ the policies satisfying every reading in C. A feasible ψ is sufficient for C when $F^{\varepsilon}_{\psi}(d) \subseteq \Pi_C(d)$, which for $C = U$ is the ε-sufficiency of Section 2.2. For a feasible $\psi \in C$, the margin is

$$\Gamma^{\varepsilon}_{\psi}(C, d) = \max_{\phi \in C \setminus \{\psi\}} \left[\sup_{\pi \in F^{\varepsilon}_{\psi}(d)} J_{c_\phi}(\pi) - d \right], \tag{1}$$

where ϕ is a competing reading and c_ϕ its cost. A supremum or maximum over the empty set is $-\infty$, so when $C = \{\psi\}$ the sole remaining reading is trivially sufficient. If $\Gamma^{\varepsilon}_{\psi}(C, d) \leq 0$, every acceptable policy under ψ satisfies every reading in C, so ψ is sufficient for C. If $\Gamma^{\varepsilon}_{\psi}(C, d) > 0$, some acceptable policy under ψ violates a competing reading and serves as a counterexample.

Known dynamics. With known dynamics, each term of the margin is a linear program, and a cheaper *value test* rules out some readings first. For each $\phi \in C \setminus \{\psi\}$, the term solves

$$\sup_{\pi \in \Pi} J_{c_\phi}(\pi) \quad \text{s.t.} \quad J_{c_\psi}(\pi) \leq d, \qquad J_r(\pi) \geq V_\psi(d) - \varepsilon, \tag{2}$$

a linear program in the policy's discounted state–action visit frequencies when Π is the class of all policies. The value test compares $V_\psi(d)$ with $V_C(d) = \sup_{\pi \in \Pi_C(d)} J_r(\pi)$, the best return while satisfying every reading in C. Always $V_C(d) \leq V_\psi(d)$, and a sufficient ψ has $V_\psi(d) = V_C(d)$ for $\varepsilon > 0$, or for $\varepsilon = 0$ when the optimum is attained, because C then admits ψ's near-optimal policies. So $V_\psi(d) > V_C(d)$ rules ψ out with one additional linear program, while equality alone does not rule it in.

Fixed offline data. When the dynamics P are unknown and only a historical log $\mathcal{D}$ is available, Arrow can certify sufficiency with high confidence but cannot refute it. It estimates the dynamics from $\mathcal{D}$ and forms a confidence set that contains the true dynamics with probability at least $1 - \delta_{\text{dec}}$ for a chosen error level δ_{dec} ("dec" for decision). With known P, the dual of the program in Equation (2) yields a certificate of the margin, a set of inequalities checked one transition row at a time (a row being the next-state distribution of one state–action pair). With P unknown, Arrow requires these inequalities to hold for every dynamics in the confidence set, the robust counterpart of the dual program (Ben-Tal et al., 2009); we call this the *robust-dual test*. It yields an upper confidence bound $\overline{\Gamma}^{\varepsilon}_{\mathcal{D},\psi}(C, d)$ on the true margin $\Gamma^{\varepsilon}_{\psi}(C, d)$. If the bound is at most zero, Arrow certifies ψ as sufficient with the stated confidence. Otherwise the data are inconclusive. Appendix C.1 gives the construction.

3.2 When no reading is yet sufficient

If no reading is sufficient, ARROW shrinks the set the learner must protect against, first by pruning readings and then by asking the operator targeted questions, which we call *clarification*. A language model or rule parser generates the readings Ψ_ℓ of the written rule ℓ. The one that actually governs the rule, if generated, is the *governing reading* $\psi^\dagger$, which we do not know. Pruning Ψ_ℓ gives the *retained set* U_0, and ARROW starts with $C = U_0$ rather than U. When implications between readings can be decided, a rule we call SAFE-KEEP removes readings whose constraints are already enforced by others, so $\Pi_{U_0}(d) = \Pi_{\Psi_\ell}(d)$ and fewer competitors need testing. Otherwise we keep every generated reading (*keep-all*) or apply SCORE-KEEP, a plausibility filter calibrated on rules whose governing reading is known. It drops implausible readings, which would cost return under full-set protection, and retains the governing reading, when generated, with probability at least $1 - \delta_{\text{sem}}$ for a chosen δ_{sem} ("sem" for semantic). Appendix C details both.

Clarification asks up to B_q questions and stops as soon as one surviving reading is sufficient. Each question splits a group of readings that jointly block sufficiency, a rule we call *hyperedge cutting*. Let U_ξ be the readings still possible after ξ answers and $q_\xi(\phi)$ the answer reading ϕ predicts for question q_ξ. After the operator answers y_ξ, we keep the consistent readings $U_{\xi+1} = \{\phi \in U_\xi : q_\xi(\phi) = y_\xi\}$. An answer is *truthful* when $y_\xi = q_\xi(\psi^\dagger)$, so truthful answers never remove $\psi^\dagger$. ARROW is rerun after each answer and stops once one surviving reading is sufficient or the budget is exhausted.

3.3 From ARROW's decision to learning

Because the policy a learner returns may fall outside what ARROW's guarantee covers, that policy is verified or certified before deployment. ARROW decides which constraint the learner trains under, one sufficient reading or all retained readings. Under a sufficient reading ψ its guarantee covers the acceptable policies $F^{\varepsilon}_{\psi}(d)$, but the learner returns one policy $\hat{\pi} \in \Pi$ that may be more than ε-suboptimal or not exactly feasible. We verify $\hat{\pi}$ in one of two ways. The first, CHECK, estimates its cost on fresh episodes independent of the log, for each reading in the set U_Ξ still possible when clarification stopped after Ξ answers. It applies an empirical-Bernstein bound (Audibert et al., 2009), a confidence bound that adapts to the observed variance, to each reading's cost, with a Bonferroni correction that splits the error level evenly across U_Ξ. This gives an upper confidence bound $\mathcal{B}(\hat{\pi}, U_\Xi)$ on $\max_{\phi \in U_\Xi} J_{c_\phi}(\hat{\pi})$ that fails with probability at most a chosen δ_{ev} ("ev" for evaluation). We deploy $\hat{\pi}$ if $\mathcal{B}(\hat{\pi}, U_\Xi) \le d$ and abstain otherwise. Keep-all and SCORE-KEEP, which retain the governing reading only with high probability, use this *checked* route. The second route needs no new data. When the learner is a planner that computes its policy on dynamics estimated from the log, it can inflate the costs it plans against by its uncertainty about the dynamics, so a policy meeting the inflated costs meets the budget under the true dynamics with probability at least $1 - \delta_{\text{off}}$ for a chosen δ_{off} ("off" for offline). This planner certifies only the retained set, which covers the governing reading only if retention preserves the feasible policy set. SAFE-KEEP guarantees this, so we pair it with this *certified-offline* route.

Appendix B.3 gives the linear programs behind both tests, the cost inflation, CHECK's settings, the route pairing, and Algorithm 1, the full pipeline from written rule to deployed policy with ARROW as its decision step; Appendix C gives the learner contract under which the guarantee reaches $\hat{\pi}$ and both routes' safety analysis.

4 Theoretical Guarantees

We give four guarantees that follow the steps of Section 3. The first characterizes when one reading can replace the full candidate set. The second shows what fixed offline data can certify and when substantial data are unavoidable. The third justifies stopping clarification before the governing reading is identified. The fourth gives conditions for safe deployment. Proofs are in Appendix C.

4.1 Sufficiency

The *sufficiency radius* measures how suboptimal a learner may be under a reading ψ before a policy that violates another reading becomes acceptable. For $C \subseteq U$ and a feasible ψ, let $R^{\text{unsafe}}_{\psi}(C, d) =$

$\sup\{J_r(\pi) : \pi \in \Pi_\psi(d) \setminus \Pi_C(d)\}$ be the best return among policies that satisfy ψ but violate another reading in C, with $\sup \emptyset = -\infty$ when no such policy exists. The radius

$$\varepsilon^\star_\psi(C, d) = V_\psi(d) - R^{\text{unsafe}}_\psi(C, d) \tag{3}$$

is the return gap between the best policy allowed by ψ and the best such violating policy.

Theorem 4.1 (Sufficiency radius). *Let $\Pi_\psi(d) \neq \emptyset$.*

(a) Threshold. *If $\varepsilon < \varepsilon^\star_\psi(C, d)$, then $F^\varepsilon_\psi(d) \subseteq \Pi_C(d)$. If $\varepsilon > \varepsilon^\star_\psi(C, d)$, the inclusion fails. At equality, when $\varepsilon^\star_\psi(C, d) < \infty$, it holds exactly when the unsafe supremum above is not attained.*

(b) ARROW's test. *$\Gamma^\varepsilon_\psi(C, d) \leq 0$ if and only if $F^\varepsilon_\psi(d) \subseteq \Pi_C(d)$. When $\Gamma^\varepsilon_\psi(C, d) > 0$, some policy in $F^\varepsilon_\psi(d)$ violates a competing reading, and when the programs in Equation (2) attain their suprema* ARROW *returns a maximizing counterexample.*

(c) No loss in the exact problem. *If $\psi \in C$ and $F^\varepsilon_\psi(d) \subseteq \Pi_C(d)$, then, for $\varepsilon > 0$ (or for $\varepsilon = 0$ when the optimum is attained), $V_\psi(d) = V_C(d)$ and $F^\varepsilon_\psi(d) = F^\varepsilon_C(d)$, where $F^\varepsilon_C(d) = \{\pi \in \Pi_C(d) : J_r(\pi) \geq V_C(d) - \varepsilon\}$.*

4.2 OFFLINE CERTIFICATION

The data that offline certification needs are governed by the *decision-information radius*, the smallest statistical change in the dynamics that would reverse ARROW's decision. ARROW does not need to estimate every policy value accurately. Its robust-dual test uses the confidence set of Section 3.1 and certifies only when uncertainty in the dynamics cannot overturn a proof of sufficiency. Let $\mathcal{Z}$ be the finite set of unknown transition rows, P_z the true next-state distribution for row $z \in \mathcal{Z}$, and w_z the proportion of the log drawn from row z, which is not random. For alternative dynamics Q with rows Q_z, let $\Gamma^\varepsilon_{\psi,Q}(C, d)$ be the ambiguity margin under Q and $\mathcal{Q}_\psi = \{Q : \Gamma^\varepsilon_{\psi,Q}(C, d) > 0\}$ the dynamics under which ψ would be insufficient. The radius is

$$I^\star_\psi(P) = \inf_{Q \in \mathcal{Q}_\psi} \sum_{z \in \mathcal{Z}} w_z \, \mathrm{KL}(P_z \,\|\, Q_z), \tag{4}$$

with KL the Kullback–Leibler divergence (relative entropy) between two next-state distributions.

Theorem 4.2 (Offline certification). *Let Π be the class of stationary randomized policies, those that choose $\pi(\cdot \mid s)$ by state alone, in the discounted setting of Section 2, and let ψ be feasible.*

(a) Soundness. *With probability at least $1 - \delta_{\text{dec}}$, every reading certified by the robust-dual test is ε-sufficient for C.*

(b) Information lower bound. *Let $0 < \delta < 1/2$. Call a procedure δ-sound if under every $Q \in \mathcal{Q}_\psi$ it certifies ψ with probability at most δ. Any δ-sound procedure that certifies ψ under P with probability at least $1 - \delta$ must use $n \geq \mathrm{kl}(1 - \delta, \delta)/I^\star_\psi(P)$ transitions, where n is the number of transitions in the log and $\mathrm{kl}(p, q)$ is the relative entropy between $\mathrm{Bernoulli}(p)$ and $\mathrm{Bernoulli}(q)$.*

4.3 STOPPING CLARIFICATION EARLY

Theorem 4.3 (Stopping clarification early). *Assume the governing reading $\psi^\dagger$ is in U_0 and answers are truthful. If, after ξ answers, some feasible reading ψ satisfies $F^\varepsilon_\psi(d) \subseteq \Pi_{U_\xi}(d)$, then every $\pi \in F^\varepsilon_\psi(d)$ satisfies $J_{c_{\psi^\dagger}}(\pi) \leq d$ and $J_r(\pi) \geq V_\psi(d) - \varepsilon$. If moreover $\psi \in U_\xi$, then, for $\varepsilon > 0$ (or $\varepsilon = 0$ when the optimum is attained), $V_\psi(d) = V_{U_\xi}(d)$ and $J_r(\pi) \geq V_{U_\xi}(d) - \varepsilon$.*

4.4 DEPLOYMENT SAFETY

Deployment can fail in four ways, and each has its own error probability. Here δ_{gen} ("gen" for generation) bounds the probability that the generated set Ψ_ℓ misses the governing reading, δ_{sem} the probability that retention removes it when it was generated, δ_{ev} the failure probability of the fresh-data check, and δ_{off} that of the certified planner. When a rule parser rather than a language model generates the readings, we write $\psi^\dagger_{\text{parsed}}$ for the executable governing reading it produces.

Theorem 4.4 (Deployment safety)**.** *Let $\hat{\pi}$ be the learned policy and* DEPLOY *the event that the pipeline deploys it, that is,* CHECK *passes on the checked route and the certified planner returns a policy on the certified-offline route.*

(a) Checked route. *Under truthful clarification,* $\Pr(\text{DEPLOY} \wedge J_{c_{\psi^\dagger}}(\hat{\pi}) > d) \leq \delta_{\text{gen}} + \delta_{\text{sem}} + \delta_{\text{ev}}$. *Conditioned on generation containing the governing reading, the bound is* $\delta_{\text{sem}} + \delta_{\text{ev}}$.

(b) Certified-offline route. *If* $\psi^\dagger_{\text{parsed}}$ *is generated,* SAFE-KEEP *preserves the feasible policy set, and the certified planner fails with probability at most* δ_{off}*, then* $\Pr(\text{DEPLOY} \wedge J_{c_{\psi^\dagger_{\text{parsed}}}}(\hat{\pi}) > d) \leq \delta_{\text{off}}$.

Interpretation. Each result prices one step of the pipeline. The sufficiency radius measures how much suboptimality can be allowed before a policy violating another reading becomes competitive (Theorem 4.1). If some dynamics consistent with the log would make ψ insufficient, certifying ψ soundly requires enough data to rule them out, and the information lower bound says how many transitions that takes (Theorem 4.2). Clarification need not identify the governing reading; it can stop once the remaining ambiguity no longer changes what the learner must enforce (Theorem 4.3). Deployment charges each stage its own error, δ_{gen} for missing the governing reading, δ_{sem} for dropping it, and δ_{ev} for the fresh-data check, while the certified-offline route pays only δ_{off} (Theorem 4.4).

5 EXPERIMENTS

We evaluate ARROW through four practical questions. We test when ambiguity changes which readings the learner must obey (Theorem 4.1, Section 5.1), whether clarification can settle that decision before the governing reading is identified (Theorem 4.3, Section 5.2), what is gained by training with one sufficient reading and checking the returned policy (Theorems 4.1 and 4.4, Section 5.3), and whether real requirements produce multiple plausible readings often enough to matter (Section 5.4). Implementation and experimental setup are in Appendix B. Appendix D adds the admission-control and maintenance domains, a live service agent, the exact return lost by combining readings into one cost, learners that accept several costs, five released safe-RL learners, the SAFE-KEEP audit, and a map from each question to its evidence and theory.

5.1 WHEN DOES AMBIGUITY MATTER?

Why. We test whether ARROW finds rules on which comparing best returns alone misses decision-relevant ambiguity, and whether the answer changes with the safety budget and policy class. Theorem 4.1 allows two readings to share the best return while allowing different near-optimal policies.

Finding. ARROW finds rules where matching the best achievable return does not make one reading sufficient, and the answer can change with the safety budget and the policy class. Table 1 reports, for the 28 compiled kube-prometheus monitoring rules (Prometheus Operator, 2026), how many rules each test leaves reducible to one reading per budget. The value test compares only the best return, so a larger count means more reductions, not a better test. At $d = 0.005$, it leaves 22 of 28 rules reducible and ARROW none. At $d = 0.02$, the counts are 27 and 22. Policy class matters too. A comparison at $d = 0.05$ covers 60 analyzable rules, the 28 of Table 1 plus 32 whose readings nest in the compiled model, one stricter than all others and so sufficient there without any test. Under the duration-only class, which uses load and duration counters, the value test leaves one reading possible for 59 of 60. Under the flexible class, which decides each telemetry scenario separately, it does so for only 5 of them. Appendix D.2 evaluates offline certification and Appendix B.2 defines the two classes.

5.2 CAN CLARIFICATION STOP BEFORE FULL IDENTIFICATION?

Why. We test whether ARROW can stop asking questions before the governing reading is known, because identifying it may require questions that do not change the constraint it trains under.

Finding. ARROW reaches a sufficient reading before the governing reading is identified in 56/61 of the sets that need clarification. Figure 1 compares clarification strategies on 87 candidate

sets from the kube-prometheus monitoring rules (Prometheus Operator, 2026) under the flexible class, each with at least two plausible readings. Hyperedge cutting, our rule of Section 3.2, asks questions that separate readings still preventing sufficiency, while price-guided clarification prioritizes questions by their effect on return. Both stop once one remaining reading is sufficient. Balanced splitting instead halves the remaining readings until one remains. After four questions, hyperedge cutting reaches sufficiency for 99% of the sets, while balanced splitting identifies the governing reading for only 37%. Among the 61 sets that initially require multiple readings, 56 reach sufficiency before exact identification under hyperedge cutting, a per-set count in Appendix D.1.

Table 1: Number of the 28 compiled monitoring rules left reducible to one reading; a rule is compiled when its readings admit an exact finite model.

Setting	Value test	ARROW
$d = 0.005$	22/28	0/28
$d = 0.01$	22/28	22/28
$d = 0.02$	27/28	22/28
$d = 0.05$	27/28	27/28
$d = 0.10$	28/28	28/28

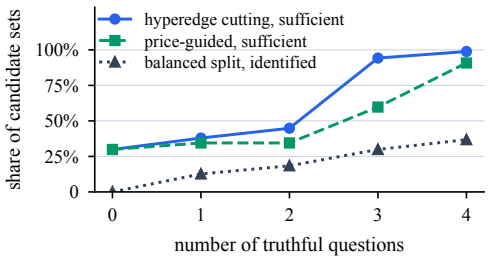

Figure 1: Clarification progress on 87 service-monitoring candidate sets.

5.3 WHAT DOES ARROW BUY DOWNSTREAM?

Why. We test whether training under a sufficient reading helps constrained learners and whether their policies can be checked before deployment. When ARROW finds one reading sufficient, Theorem 4.1 shows replacing the candidate set by that reading preserves the exact constrained target.

Finding. Training with the sufficient reading improves return for learners that accept one safety cost. Some constrained RL learners accept one cost signal, so protecting against several readings requires one combined conservative cost. Table 2 tests this on the 9 of 36 instances of our synthetic agent-service benchmark that ARROW certifies, in which a tool-using agent calls external services and the rule limiting "the external service" can refer to any of them. On the remaining instances the reading is not certified and the learner keeps the combined cost. The learners are six offline constrained RL algorithms: BCQ-FQI-Lagrangian, PID-Lagrangian, O3SRL, CPQ, CAPS, and CQL-Lagrangian (Fujimoto et al., 2019; Stooke et al., 2020; Chemingui et al., 2025a; Xu et al., 2022; Chemingui et al., 2025b; Kumar et al., 2020). We keep the learner, data, and budget fixed and change only the cost. The ARROW reading raises return, as a share of the exact full-set optimum, by a median of 15.0–37.3 percentage points, depending on the learner, and CHECK let no unsafe policy through among 3,960 tested policies. Appendix B.2 describes the benchmark and Appendix D.3 the full results.

Table 2: Return as a share of the exact full-set optimum on the agent-service benchmark.

Learner	return, combined cost	return, ARROW reading	median gain
BCQ-FQI-Lagrangian	74.8%	97.5%	+15.0
PID-Lagrangian	67.8%	94.7%	+22.0
O3SRL	74.7%	97.5%	+15.0
CPQ	75.8%	94.9%	+20.7
CAPS	54.2%	74.5%	+20.2
CQL-Lagrangian (neural)	61.7%	97.4%	+37.3

5.4 DO MULTIPLE PLAUSIBLE READINGS ARISE ON REAL REQUIREMENTS?

Why. We test whether real requirements produce several plausible readings often enough for ARROW to matter, since the previous experiments assume they do.

Finding. Real requirements often leave several readings for ARROW to assess. Table 3 uses 182 expert-annotated ventilator, robotics, and Lockheed Martin cyber-physical (LMCPS) requirements from the ARTEMIS release (Mendoza et al., 2026), a public benchmark pairing natural-language requirements with expert formalizations in temporal logic. A reading is plausible when it matches a formalization accepted by the domain experts. Each method produced ten translations per requirement, fifty for ARTEMIS with reflection. For each method, the main entry is the share whose first translation is plausible. The parenthesized entry is the share with at least one plausible translation, an upper bound under perfect selection. No method's first translation is plausible for more than 35.7%. Across all samples, 80.2% of requirements have a plausible reading, while SCORE-KEEP retains one for 73.1%, preserving 91% of that coverage. The 19.8% with no plausible reading in any sample are generation misses. After SAFE-KEEP's redundancy removal, only 12.6% reduce to one reading, and the retained sets keep a median of 4 readings (Appendix D.4).

Table 3: Plausible readings retained on 182 real requirements. The bottom rows apply one rule to all samples from both large language models (LLMs), giving one entry for the share retaining a plausible reading. Methods: DeepSTL (He et al., 2022), nl2spec (Cosler et al., 2023), NL2TL (Chen et al., 2023), SynthTL (Mendoza et al., 2024) and ARTEMIS (Mendoza et al., 2026) with its released baselines directTL and directTL-t and its NL2TL+ and fine-tuned (FT) adaptations.

		% plausible reading, first translation (any sample)			
Method	LLM or pool	Ventilator	Robotics	LMCPS	All
DeepSTL (FT)	Gemini	15.7 (27.3)	21.7 (30.4)	60.0 (80.0)	20.9 (32.4)
nl2spec	Gemini	18.2 (24.0)	23.9 (32.6)	73.3 (80.0)	24.2 (30.8)
nl2spec	GPT-4.1	13.2 (20.7)	28.3 (32.6)	66.7 (80.0)	21.4 (28.6)
NL2TL+	Gemini	3.3 (9.1)	6.5 (17.4)	0.0 (13.3)	3.8 (11.5)
NL2TL+	GPT-4.1	5.8 (9.9)	8.7 (21.7)	6.7 (6.7)	6.6 (12.6)
NL2TL+ (FT)	Gemini	2.5 (12.4)	8.7 (26.1)	6.7 (6.7)	4.4 (15.4)
SynthTL	Gemini	12.4 (16.5)	15.2 (21.7)	26.7 (73.3)	14.3 (22.5)
SynthTL	GPT-4.1	1.7 (10.7)	0.0 (10.9)	6.7 (20.0)	1.6 (11.5)
directTL	Gemini	17.4 (23.1)	21.7 (32.6)	73.3 (80.0)	23.1 (30.2)
directTL	GPT-4.1	13.2 (23.1)	19.6 (32.6)	46.7 (80.0)	17.6 (30.2)
directTL-t	Gemini	18.2 (31.4)	28.3 (45.7)	80.0 (80.0)	25.8 (39.0)
directTL-t	GPT-4.1	15.7 (24.8)	13.0 (23.9)	46.7 (80.0)	17.6 (29.1)
ARTEMIS	Gemini	31.4 (61.2)	45.7 (71.7)	40.0 (100.0)	35.7 (67.0)
ARTEMIS	GPT-4.1	21.5 (39.7)	32.6 (56.5)	46.7 (86.7)	26.4 (47.8)
ARTEMIS + reflection (50 translations)	Gemini	33.1 (66.9)	26.1 (69.6)	46.7 (100.0)	32.4 (70.3)
Self-consistency vote	union	22.3	26.1	80.0	28.0
All distinct samples	union	74.4	89.1	100.0	80.2
ARROW SCORE-KEEP	union	65.3	84.8	100.0	73.1

6 CONCLUSION

Ambiguous safety rules need not be fully resolved before constrained RL training. What matters is whether their differences change the near-optimal policies a learner may return. ARROW tests this, returns a counterexample when one reading is insufficient, and can stop clarification once the training decision is settled. SAFE-KEEP removes readings already enforced by others. Across our settings, whether ambiguity matters depends on the policy class and the safety budget, and clarification often stops before the governing reading is identified.

Limitations. Fully offline guarantees hold only for rules a parser translates exactly, and require successful parsing, sound implication checks, and certified planning. For other rules, safety also relies on the fresh-data check and on generation including the governing reading, which fails less often with more generated readings. The guarantees concern executable readings, not intent that no reading captures.

AI USE STATEMENT

In this work, we used generative AI tools to help draft and edit the manuscript, to search for related literature and to check references. Also, we used them to help generate synthetic data, which we verified for factual accuracy, and to help in debugging and polishing the code. We did not use them to develop the theory or conceptual framework, to design the methodology or the experiments, to formulate mathematical claims or their proofs, to propose or refine hypotheses, to interpret results, or to clean or reformat data. Beyond that debugging and polishing, we did not use them to execute the research. Translation and qualitative or thematic analysis are not applicable. We have reviewed all the work, verifying every proof, every related-work claim against the cited paper, and regenerating every reported number from the released code. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All data are public, namely the kube-prometheus and Kyverno rule corpora, the further repository rules of Table 5, the ARTEMIS release with its expert annotations, NASA C-MAPSS and the DSRL/OSRL benchmark. No human subjects took part, and the clarification answers in the experiments are simulated by a truthful oracle. The live service-agent check posts to a sandboxed endpoint under a published posting-rate policy, and its only external effect is the billed API cost, which we report. No learned controller was deployed on a physical system. The method makes the constraint a learner trains under explicit and checkable, and it does not certify intent that no candidate reading captures, as the Limitations state.

REPRODUCIBILITY STATEMENT

All code, the recorded experiment outputs and the scripts that produce every table and figure are at https://anonymous.4open.science/r/safe-rl-ambiguous-rules; its documentation maps each table and figure of this paper to the script and output behind it, and one command regenerates every number in the paper from those outputs. All data are public: the kube-prometheus alert rules (Prometheus Operator, 2026), the Kyverno policy library (Kyverno contributors, 2026), the alert rules of the further and public repositories of Table 5, the ARTEMIS release (Mendoza et al., 2026), the NASA C-MAPSS runs (Saxena et al., 2008) and the DSRL/OSRL benchmark with its released learners (Liu et al., 2024); the repository-based sources and the ARTEMIS release were read at the commits given in their bibliography entries. The live-agent study ran against a sandboxed endpoint; its logs are included.

REFERENCES

Eitan Altman. *Constrained Markov Decision Processes*. Stochastic Modeling Series. Chapman & Hall/CRC, 1999.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Exploration-exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009. doi: 10.1016/j.tcs.2009.01.016.

Qinbo Bai, Amrit Singh Bedi, Mridul Agarwal, Alec Koppel, and Vaneet Aggarwal. Achieving zero constraint violation for constrained reinforcement learning via primal-dual approach. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, volume 36, pp. 3682–3689, 2022. doi: 10.1609/aaai.v36i4.20281.

Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

Samuel Berthe. Awesome Prometheus alerts. https://github.com/samber/awesome-prometheus-alerts, 2026. GitHub repository; alert rules read at commit a8533189 (2026-08-01).

Leo Breiman. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199–231, 2001. doi: 10.1214/ss/1009213726.

Ceph. Ceph. https://github.com/ceph/ceph, 2026. GitHub repository; alert rules read at commit `8482cb11` (2026-07-31).

Yassine Chemingui, Aryan Deshwal, Alan Fern, Thanh Nguyen-Tang, and Janardhan Rao Doppa. Online optimization for offline safe reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 156036–156064, 2025a. doi: 10.52202/085713-4700. arXiv:2510.22027.

Yassine Chemingui, Aryan Deshwal, Honghao Wei, Alan Fern, and Janardhan Rao Doppa. Constraint-adaptive policy switching for offline safe reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 39, pp. 15722–15730, 2025b. doi: 10.1609/aaai.v39i15.33726. arXiv:2412.18946.

Yongchao Chen, Rujul Gandhi, Yang Zhang, and Chuchu Fan. NL2TL: Transforming natural languages to temporal logics using large language models. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 15880–15903. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.985.

Yuxin Chen, Shervin Javdani, Amin Karbasi, J. Andrew Bagnell, Siddhartha S. Srinivasa, and Andreas Krause. Submodular surrogates for value of information. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, volume 29, pp. 3511–3518, 2015. doi: 10.1609/aaai.v29i1.9694.

Jaymari Chua, Chen Wang, and Lina Yao. Learning natural language constraints for safe reinforcement learning of language agents. *arXiv preprint arXiv:2504.03185*, 2025.

Matthias Cosler, Christopher Hahn, Daniel Mendoza, Frederik Schmitt, and Caroline Trippel. nl2spec: Interactively translating unstructured natural language to temporal logics with large language models. In *Computer Aided Verification (CAV)*, volume 13965 of *Lecture Notes in Computer Science*, pp. 383–396. Springer, 2023. doi: 10.1007/978-3-031-37703-7_18.

Jean-Michel Couvreur. On-the-fly verification of linear temporal logic. In *World Congress on Formal Methods (FM)*, volume 1708 of *Lecture Notes in Computer Science*, pp. 253–271. Springer, 1999. doi: 10.1007/3-540-48119-2_16.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022.

Alexandre Duret-Lutz, Etienne Renault, Maximilien Colange, Florian Renkin, Alexandre Gbaguidi Aisse, Philipp Schlehuber-Caissier, Thomas Medioni, Antoine Martin, Jérôme Dubois, Clément Gillard, and Henrich Lauko. From Spot 2.0 to Spot 2.10: What's new? In *Computer Aided Verification (CAV)*, volume 13372 of *Lecture Notes in Computer Science*, pp. 174–187. Springer, 2022. doi: 10.1007/978-3-031-13188-2_9.

Marie Farrell, Matt Luckcuck, Rosemary Monahan, Conor Reynolds, and Oisín Sheridan. FRETting and formal modelling: A mechanical lung ventilator. In *Rigorous State-Based Methods (ABZ)*, volume 14759 of *Lecture Notes in Computer Science*, pp. 360–383. Springer, 2024. doi: 10.1007/978-3-031-63790-2_28.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, 20(177):1–81, 2019.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2052–2062. PMLR, 2019.

Tomas Gal. *Postoptimal Analyses, Parametric Programming, and Related Topics*. Walter de Gruyter, Berlin, 2 edition, 1995.

Sourav Ganguly, Kishan Panaganti, Arnob Ghosh, and Adam Wierman. Efficient policy optimization in robust constrained MDPs with iteration complexity guarantees. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 10394–10433, 2025. doi: 10.52202/085713-0318.

Javier García and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(42):1437–1480, 2015.

Dimitra Giannakopoulou, Thomas Pressburger, Anastasia Mavridou, and Johann Schumann. Automated formalization of structured natural language requirements. *Information and Software Technology*, 137:106590, 2021. doi: 10.1016/j.infsof.2021.106590.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

Grafana Labs. Grafana Loki. https://github.com/grafana/loki, 2026a. GitHub repository; alert rules read at commit 6a48053e (2026-08-01).

Grafana Labs. Grafana Mimir. https://github.com/grafana/mimir, 2026b. GitHub repository; alert rules read at commit 9f0e877a (2026-07-31).

Grafana Labs. Grafana Tempo. https://github.com/grafana/tempo, 2026c. GitHub repository; alert rules read at commit df7893fa (2026-07-30).

Zijian Guo, Weichao Zhou, and Wenchao Li. Temporal logic specification-conditioned decision transformer for offline safe reinforcement learning. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17003–17019. PMLR, 2024. arXiv:2402.17217.

Dylan Hadfield-Menell, Smitha Milli, Pieter Abbeel, Stuart Russell, and Anca Dragan. Inverse reward design. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 6765–6774, 2017.

Jie He, Ezio Bartocci, Dejan Ničković, Haris Isakovic, and Radu Grosu. DeepSTL: From English requirements to signal temporal logic. In *International Conference on Software Engineering (ICSE)*, pp. 610–622. ACM, 2022. doi: 10.1145/3510003.3510171.

Ronald A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1):22–26, 1966.

Garud N. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2): 257–280, 2005.

Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, J. Andrew Bagnell, and Siddhartha S. Srinivasa. Near optimal Bayesian active learning for decision making. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 33 of *Proceedings of Machine Learning Research*, pp. 430–438, 2014.

Mark H. Karwan, Vahid Lotfi, Jan Telgen, and Stanley Zionts. *Redundancy in Mathematical Programming: A State-of-the-Art Survey*, volume 206 of *Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, 1983.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, 2016.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. CLAM: Selective clarification for ambiguous questions with generative language models. In *ICML 2023 Workshop on Challenges in Deployable Generative AI*, 2023. arXiv:2212.07769.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 1179–1191, 2020.

Kyverno contributors. Kyverno policy library. https://github.com/kyverno/policies, 2026. GitHub repository; `ClusterPolicy` and `Policy` manifests read at commit `ef9843f0` (2026-07-04).

Jongmin Lee, Cosmin Paduraru, Daniel J. Mankowitz, Nicolas Heess, Doina Precup, Kee-Eung Kim, and Arthur Guez. COptiDICE: Offline constrained reinforcement learning via stationary distribution correction estimation. In *International Conference on Learning Representations (ICLR)*, 2022.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

David Lindner, Xin Chen, Sebastian Tschiatschek, Katja Hofmann, and Andreas Krause. Learning safety constraints from demonstrations with unknown rewards. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 238 of *Proceedings of Machine Learning Research*, pp. 2386–2394, 2024.

Guiliang Liu, Sheng Xu, Shicheng Liu, Ashish Gaurav, Sriram Ganapathi Subramanian, and Pascal Poupart. A comprehensive survey on inverse constrained reinforcement learning: Definitions, progress and challenges. *Transactions on Machine Learning Research (TMLR)*, 2025a. arXiv:2409.07569.

Jason Xinyu Liu, Ziyi Yang, Ifrah Idrees, Sam Liang, Benjamin Schornstein, Stefanie Tellex, and Ankit Shah. Grounding complex natural language commands for temporal tasks in unseen environments. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1084–1110. PMLR, 2023a.

Zifan Liu, Xinran Li, and Jun Zhang. C2IQL: Constraint-conditioned implicit Q-learning for safe offline reinforcement learning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 38827–38841, 2025b.

Zuxin Liu, Zijian Guo, Yihang Yao, Zhepeng Cen, Wenhao Yu, Tingnan Zhang, and Ding Zhao. Constrained decision transformer for offline safe reinforcement learning. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21611–21630. PMLR, 2023b.

Zuxin Liu, Zijian Guo, Haohong Lin, Yihang Yao, Jiacheng Zhu, Zhepeng Cen, Hanjiang Hu, Wenhao Yu, Tingnan Zhang, Jie Tan, and Ding Zhao. Datasets and benchmarks for offline safe reinforcement learning. *Journal of Data-centric Machine Learning Research (DMLR)*, 1(12):1–29, 2024. arXiv:2306.09303.

Shaocong Ma, Ziyi Chen, Yi Zhou, and Heng Huang. Rectified robust policy optimization for model-uncertain constrained reinforcement learning without strong duality. *Transactions on Machine Learning Research*, 2025. arXiv:2508.17448.

Shehryar Malik, Usman Anwar, Alireza Aghasi, and Ali Ahmed. Inverse constrained reinforcement learning. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7390–7399. PMLR, 2021.

Charles Marx, Flavio Calmon, and Berk Ustun. Predictive multiplicity in classification. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6765–6774, 2020.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein bounds and sample variance penalization. In *Proceedings of the 22nd Conference on Learning Theory (COLT)*, 2009.

Anastasia Mavridou, Hamza Bourbouh, Dimitra Giannakopoulou, Thomas Pressburger, Mohammad Hejase, Pierre-Loic Garoche, and Johann Schumann. The ten Lockheed Martin cyber-physical challenges: Formalized, analyzed, and explained. In *IEEE International Requirements Engineering Conference (RE)*, pp. 300–310. IEEE, 2020. doi: 10.1109/RE48521.2020.00040.

Daniel Mendoza, Christopher Hahn, and Caroline Trippel. Translating natural language to temporal logics with large language models and model checkers. In *Formal Methods in Computer-Aided Design (FMCAD)*, pp. 119–129. TU Wien Academic Press, 2024. doi: 10.34727/2024/isbn.978-3-85448-065-5_17.

Daniel Mendoza, Anastasia Mavridou, Andreas Katis, and Caroline Trippel. Automating requirements formalization: Using LLMs and low-complexity distinguishing traces for semantic validation. In *International Conference on Software Engineering (ICSE)*, pp. 1314–1326. ACM, 2026. doi: 10.1145/3744916.3787815. Data and code: https://github.com/dmmendo/ARTEMIS, read at commit 9d809bbc95.

Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

OpenShift. cluster-monitoring-operator. https://github.com/openshift/cluster-monitoring-operator, 2026. GitHub repository; alert rules read at commit 54fe9b4f (2026-07-31).

Dimitris Papadimitriou, Usman Anwar, and Daniel S. Brown. Bayesian methods for constraint inference in reinforcement learning. *Transactions on Machine Learning Research*, 2022.

PingCAP. TiDB. https://github.com/pingcap/tidb, 2026. GitHub repository; alert rules read at commit 6f5bfe19 (2026-08-01).

Nicholas Polosky, Bruno C. Da Silva, Madalina Fiterau, and Jithin Jagannath. Constrained offline policy optimization. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17801–17810. PMLR, 2022.

Prometheus Operator. kube-prometheus. https://github.com/prometheus-operator/kube-prometheus, 2026. GitHub repository; alert rules read at commit ecb17ed8 (2026-07-28).

Kevin Regan and Craig Boutilier. Regret-based reward elicitation for Markov decision processes. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 444–451, 2009.

Kevin Regan and Craig Boutilier. Robust policy computation in reward-uncertain MDPs using nondominated policies. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, pp. 1127–1133, 2010. doi: 10.1609/aaai.v24i1.7740.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *Proceedings of the 7th Conference on Robot Learning (CoRL)*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682. PMLR, 2023.

Rook. Rook. https://github.com/rook/rook, 2026. GitHub repository; alert rules read at commit 5238f377 (2026-07-31).

Reazul Hasan Russel, Mouhacine Benosman, and Jeroen Van Baar. Robust constrained-MDPs: Soft-constrained robust policy optimization under model uncertainty. *arXiv preprint arXiv:2010.04870*, 2020.

Abhinav Saxena, Kai Goebel, Don Simon, and Neil Eklund. Damage propagation modeling for aircraft engine run-to-failure simulation. In *International Conference on Prognostics and Health Management (PHM)*, pp. 1–9. IEEE, 2008. doi: 10.1109/PHM.2008.4711414.

Ankit Shah, Shen Li, and Julie Shah. Planning with uncertain specifications (PUnS). *IEEE Robotics and Automation Letters*, 5(2):3414–3421, 2020. arXiv:1906.03218.

Joar Skalse, Matthew Farrugia-Roberts, Stuart Russell, Alessandro Abate, and Adam Gleave. Invariance in policy optimisation and partial identifiability in reward learning. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32033–32058, 2023.

Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by PID Lagrangian methods. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9133–9143. PMLR, 2020.

David Smith Sundarsingh, Jun Wang, Jyotirmoy V. Deshmukh, and Yiannis Kantaros. ConformalNL2LTL: Translating natural language instructions into temporal logic formulas with conformal correctness guarantees. *arXiv preprint arXiv:2504.21022*, 2025.

Hazel M. Taylor, Anastasia Mavridou, Marie Farrell, and Louise A. Dennis. Explainability pattern specifications for human-robot teamwork. In *IEEE Engineering Reliable Autonomous Systems (ERAS)*, pp. 1–8. IEEE, 2025. doi: 10.1109/ERAS63351.2025.11135747.

Jan Telgen. Identifying redundant constraints and implicit equalities in systems of linear constraints. *Management Science*, 29(10):1209–1222, 1983. doi: 10.1287/mnsc.29.10.1209.

Thanos. Thanos. https://github.com/thanos-io/thanos, 2026. GitHub repository; alert rules read at commit `d7d0f7f6` (2026-07-31).

V. Varagapriya, Vikas Vikram Singh, and Abdel Lisser. Constrained Markov decision processes with uncertain costs. *Operations Research Letters*, 50(2):218–223, 2022. doi: 10.1016/j.orl.2022.02.001.

VictoriaMetrics. VictoriaMetrics. https://github.com/VictoriaMetrics/VictoriaMetrics, 2026. GitHub repository; alert rules read at commit `a8759a53` (2026-07-31).

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005. doi: 10.1007/b106715.

Yue Wang, Fei Miao, and Shaofeng Zou. Robust constrained reinforcement learning. *arXiv preprint arXiv:2209.06866*, 2022.

Tsachy Weissman, Erik Ordentlich, Gadiel Seroussi, Sergio Verdu, and Marcelo J. Weinberger. Inequalities for the L_1 deviation of the empirical distribution. Technical Report HPL-2003-97 (R.1), Hewlett-Packard Laboratories, Palo Alto, CA, 2003.

Wolfram Wiesemann, Angelos Tsoukalas, Polyxeni-Margarita Kleniati, and Berç Rustem. Pessimistic bilevel optimization. *SIAM Journal on Optimization*, 23(1):353–380, 2013. doi: 10.1137/120864015.

Haoran Xu, Xianyuan Zhan, and Xiangyu Zhu. Constraints penalized Q-learning for safe offline reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8753–8760, 2022. doi: 10.1609/aaai.v36i8.20855.

Ruiqi Xue, Lei Yuan, Kainuo Cheng, Jing-Wen Yang, and Yang Yu. Model-based proactive cost generation for learning safe policies offline with limited violation data. *arXiv preprint arXiv:2605.01356*, 2026.

Tsung-Yen Yang, Michael Y. Hu, Yinlam Chow, Peter J. Ramadge, and Karthik Narasimhan. Safe reinforcement learning with natural language constraints. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 13794–13808, 2021.

Yinan Zheng, Jianxiong Li, Dongjie Yu, Yujie Yang, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Safe offline reinforcement learning with feasibility-guided diffusion model. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2401.10700.

CONTENTS OF THE APPENDIX

A Positioning Relative to Prior Work **16**

B Experimental Setup and Implementation **18**

B.1 Data sources and benchmarks 18

B.2 Decision models and policy classes 19

B.3 ARROW implementation and experimental protocols 21

C Theory and Proofs **25**

C.1 Proofs of the main results 25

C.1.1 Theorem 4.1: the sufficiency radius 25

C.1.2 Theorem 4.2: certification from fixed data 28

C.1.3 Theorem 4.3: safe stopping under clarification 34

C.1.4 Theorem 4.4: deployment safety 35

C.2 Supporting theory used by the pipeline 36

C.2.1 The dual certificate in finite CMDPs 36

C.2.2 Composition with an approximate learner 37

C.2.3 Safety-preserving semantic compression (SAFE-KEEP) 37

C.2.4 The single-cost surrogate 38

D Additional Experimental Results **38**

D.1 Decision relevance and clarification 38

D.1.1 Decision geometry, replication and simple alternatives 38

D.1.2 Operational consequences, a live agent and learned maintenance policies . 40

D.1.3 Clarification before identification 41

D.2 Fixed-data certification and retention 41

D.2.1 Certificate comparison and information floor 41

D.2.2 SAFE-KEEP audit and certified-offline deployment 42

D.3 Downstream learning and deployment 43

D.3.1 Exact surrogate price, learned agent-service gains and native learners . . . 44

D.3.2 Portability to released safe-RL learners 46

D.3.3 Deployment checking 46

D.4 Real requirements 47

A POSITIONING RELATIVE TO PRIOR WORK

Table 4 shows that ARROW is the only listed method that takes all three inputs, a written rule, several candidate readings of it and a fixed offline log. A reading is one executable meaning of the rule, given as a cost (Section 2.1).

What ARROW introduces. ARROW asks only whether one reading preserves the policies a constrained learner may return, not the full candidate set or the whole feasible region. Formally, it tests

Table 4: Positioning by problem solved, safety input and guarantee. The narrow columns mark a written rule (rule), multiple candidate readings of it (cand.), and a fixed offline log (log).

Work	Problem	Safety input	rule	cand.	log	Result
Ma et al. (2025)	robust constrained RL	model uncertainty set				approx. optimal, feasible within tolerance for the set
Guo et al. (2024)	conditioned offline RL	STL specification, target satisfaction			✓	empirical alignment with target
Chemingui et al. (2025b)	offline safe RL	cost and limit			✓	conditional finite-horizon cost bound
Chemingui et al. (2025a)	constrained offline RL	cumulative cost			✓	approximate optimality with an offline-RL oracle
Yang et al. (2021)	language-constrained RL	one interpreted textual constraint	✓			commits to one reading
Xue et al. (2026)	offline RL	generated cost	✓		✓	optimizes generated cost
Mendoza et al. (2026)	requirement formalization	candidate specifications	✓	✓		selects one user-validated candidate
ARROW	**constrained RL under candidate readings**	**candidate costs from one rule**	✓	✓	✓	**one reading or the set; on the structured family, certified fixed-data deployment**

whether reading ψ is sufficient, $F^{\varepsilon}_{\psi}(d) \subseteq \Pi_C(d)$, where $F^{\varepsilon}_{\psi}(d)$ holds the acceptable policies under ψ, those satisfying ψ with return within ε of the best (Section 2.2), and $\Pi_C(d)$ holds the policies satisfying every reading in C, the readings still under consideration (Section 3.1). Robust constrained reinforcement learning (RL) (Varagapriya et al., 2022; Wang et al., 2022; Ma et al., 2025; Ganguly et al., 2025) instead protects against every member of a supplied set, which for a set of readings is full-set protection in our terms (Section 2.1). It follows robust dynamic programming over uncertain transition models (Nilim & El Ghaoui, 2005; Iyengar, 2005), and Russel et al. (2020) brought such uncertainty sets to constrained Markov decision processes (Section 2.1). Redundancy analysis (Telgen, 1983; Karwan et al., 1983) asks whether removing a constraint leaves the entire feasible region unchanged. Postoptimal analysis (Gal, 1995) asks how the optimum moves when a constraint is perturbed. For ARROW, by contrast, two readings with different feasible regions can be interchangeable for training. The decision depends on the learner, policy class, budget and tolerance. An insufficient reading comes with a counterexample, an acceptable policy that violates a competing reading.

Adjacent lines of work. Adjacent work studies rewards, inferred constraints or near-optimal model sets, whereas ARROW studies the safety constraint induced by one written rule. Reward uncertainty and reward learning (Hadfield-Menell et al., 2017; Regan & Boutilier, 2009; 2010) and the partial identifiability of rewards (Skalse et al., 2023) concern the objective. Constraint inference (Malik et al., 2021; Papadimitriou et al., 2022; Lindner et al., 2024; Liu et al., 2025a) estimates a constraint or a safe set from demonstrations, and recent work learns natural-language constraints for language agents from demonstrations (Chua et al., 2025). All of it sits upstream of whether supplied candidate readings differ for the learner. Pessimistic bilevel optimization (Wiesemann et al., 2013) and Rashomon-set, predictive-multiplicity and underspecification analyses (Breiman, 2001; Fisher et al., 2019; Marx et al., 2020; D'Amour et al., 2022) range over an optimizer's whole optimal set, or over models that are optimal, near-optimal or indistinguishable under a predictive or validation criterion. ARROW ranges over the near-optimal policies of one executable reading and tests them under the competing readings.

Offline safe reinforcement learning. ARROW composes with offline safe RL learners rather than competing with them, because these learners take the cost function as given without asking where it comes from. They maximize return under a cost limit from a fixed log (Levine et al., 2020; Liu et al., 2024). Their tools are Lagrangian duality, which moves the cost limit into the objective with a

learned price (Stooke et al., 2020; Bai et al., 2022), pessimistic critics (Xu et al., 2022), confidence bounds on the cost (Polosky et al., 2022), occupancy (state–action visit frequency) corrections (Lee et al., 2022), return conditioning (Liu et al., 2023b), feasibility-guided generation (Zheng et al., 2024) and policy switching (Chemingui et al., 2025b). Appendix D.3.2 runs five released implementations unmodified under retained-set protection, which enforces every reading in the retained set U_0 of plausible generated readings (Section 3.2).

From language to executable specifications. Translation methods aim at the right formula, whereas ARROW takes their output as its input. Translating a written requirement into temporal logic, a formal language for how a system evolves over time, is an active line of work (He et al., 2022; Cosler et al., 2023; Chen et al., 2023; Mendoza et al., 2024; Liu et al., 2023a), and recent systems attach a conformal guarantee to the translation (Sundarsingh et al., 2025) or validate it on distinguishing traces (Mendoza et al., 2026). Planning under uncertain specifications (Shah et al., 2020) keeps a distribution over formulas, and language-model planners ask for help when their prediction sets are wide (Ren et al., 2023). ARROW asks the downstream question, whether the remaining disagreement changes the policies a constrained learner may return.

Relation to decision-focused clarification. ARROW induces a decision-focused clarification problem whose decision regions are defined by constrained-policy sufficiency. Clarification removes candidate readings through targeted operator questions (Section 3.2). Decision Region Determination (Javdani et al., 2014; Chen et al., 2015) stops information gathering once the surviving hypotheses fix a downstream decision. It follows value-of-information analysis (Howard, 1966) and adaptive submodular active learning (Golovin & Krause, 2011). Selective clarification (Kuhn et al., 2023) asks only when the user's question is ambiguous. Against full-set protection, the standard robust response (Varagapriya et al., 2022), ARROW certifies whether that protection is needed.

B EXPERIMENTAL SETUP AND IMPLEMENTATION

B.1 DATA SOURCES AND BENCHMARKS

Table 5 lists the sources and their populations, three of them our own, the synthetic maintenance task, the agent-service benchmark and the live service-agent check. The repositories and the ARTEMIS release were read at the commits given in their bibliography entries. A *reading* is one executable meaning of a rule, with its own safety cost, and a *pool* is the set of readings generated for one rule. kube-prometheus holds Prometheus alerting rules for Kubernetes clusters, Kyverno policies decide which Kubernetes configurations a cluster admits, NASA C-MAPSS is simulated turbofan-engine degradation data, and DSRL/OSRL are offline safe-RL datasets and learners. The portability study of Table 19 changes only the released learners' standard cost-limit and training controls, and that benchmark's paired readings are nested.

Candidate pools. A fixed rule-based generator turns each parsed monitoring rule into a pool, and the rules are then grouped by how exactly their pools can be analyzed. The pool holds the parsed rule plus structurally valid variants of its threshold, comparator, duration, window, aggregation, scope and guard logic. Variants that behave identically on the rule's fixtures, short synthetic telemetry scenarios, are merged. Pools with at least two readings form the candidate sets of the flexible class. A rule compiles when its readings admit the exact load-chain model of the duration-only controller (both classes and the model are defined in Appendix B.2). The compiled model analyses a rule's threshold-duration subfamily, its two to four maximal readings, while the flexible class decides over every reading in the candidate pool. A rule is nested when one of its readings is stricter than all the others. That reading is then sufficient without a solve in that model, since every policy it allows satisfies the others. The compiled and the nested rules together are the 60 analyzable rules of the policy-class comparison of Section 5.1. In admission control each Kyverno policy's candidate readings determine which configurations it admits. The structured family holds the rules that the threshold-duration parser admits. These rules carry both the audit of every SAFE-KEEP removal and the certified-offline pipeline (Appendix B.3).

Expert-annotated requirements. For each requirement the ARTEMIS release provides expert readings and archived language-model translations produced without access to them. The require-

Table 5: **Data sources and populations.** Our own constructions are marked (ours).

Source	Population	Used in
kube-prometheus (Prometheus Operator, 2026)	88 rule pools, 87 with two or more readings; 28 compile, 32 nested, 28 outside the grammar	Tables 1, 10, 11, 13, 14, 16
Eight further repositories (Grafana Labs, 2026a;b; OpenShift, 2026; VictoriaMetrics, 2026; Rook, 2026; Ceph, 2026; Thanos, 2026; PingCAP, 2026)	465 retained pools from Grafana Loki, Grafana Mimir, OpenShift cluster-monitoring-operator, VictoriaMetrics, Rook, Ceph, Thanos and TiDB	Appendix D.1.1
Six public repositories (Berthe, 2026; VictoriaMetrics, 2026; Rook, 2026; Ceph, 2026; Grafana Labs, 2026a;c)	1,271 rules from awesome-prometheus-alerts, VictoriaMetrics, Rook, Ceph, Loki and Tempo, 581 admitted (45.7%), 579 with two or more readings	Table 15
Kyverno policy library (Kyverno contributors, 2026)	44 pools with two or more readings	Tables 11, 13
Synthetic maintenance (ours)	horizon $H{=}30$	Table 12
NASA C-MAPSS (Saxena et al., 2008)	replay model, horizon $H{=}120$	Table 12
Live service agent (ours)	50 paired sessions per condition, GPT-4o-mini, retained $k \in \{3, 4, 5, 6, 8\}$	Appendix D.1.2
Agent-service benchmark (ours)	36 instances, 9 with a certified reading	Tables 2, 17, 18, 20
ARTEMIS release (Mendoza et al., 2026)	182 requirements, 15 archives, 190 samples per requirement	Tables 3, 21
DSRL/OSRL (Liu et al., 2024)	AntRun, BallCircle, CarCircle, CarRun and DroneCircle with five released learners, 1,084 runs	Table 19

ments come from three sets, named as in Table 3. Ventilator is a mechanical lung ventilator (Farrell et al., 2024), Robotics is human-robot teamwork (Taylor et al., 2025), and LMCPS is the Lockheed Martin cyber-physical challenges (Mavridou et al., 2020), with 121, 46 and 15 requirements. The expert readings are curated formalizations in FRETish (Giannakopoulou et al., 2021), the structured English of the FRET requirements tool, augmented by three domain experts with additional plausible readings. The translations are in linear temporal logic (LTL), a formal language for properties of executions over time. They come from 9 generators, 6 of them under both Gemini-2.5-Flash and GPT-4.1 and 3 under Gemini-2.5-Flash only, ten samples each and fifty for ARTEMIS with reflection.

B.2 Decision models and policy classes

Each domain is a decision model, a finite Markov decision process (MDP) whose readings differ only in their costs, and Table 6 summarizes the four. This subsection gives the monitoring and agent-service constructions in full and defines the policy classes.

Service monitoring. Each monitoring reading adds one duration counter to the state and charges running while it fires. Let $z_t \in \{0, \ldots, L-1\}$ be the observed load level, with level 0 below every candidate threshold. A reading $\psi_k = (\theta_k, h_k)$, "load above θ_k for h_k minutes", needs one monitor variable $m_{k,t} \in \{0, \ldots, h_k\}$, the number of consecutive 60 s steps at or above lvl_k, the load level of θ_k, capped at h_k. The state is $s_t = (z_t, m_{1,t}, \ldots, m_{K,t})$ plus an absorbing state $s_\perp$, and the action is $a_t \in \{\mathsf{run}, \mathsf{drain}\}$. Under run the load follows a Markov chain $z_{t+1} \sim P_L(\cdot \mid z_t)$, the only stochastic component, and each monitor updates deterministically, $m_{k,t+1} = \min\{h_k, m_{k,t} + 1\}$ if

Table 6: **Decision models.** What each reading charges, and the horizon, budget and policy class of each domain.

Domain	A reading charges	Horizon	Budget d	Policy class
Service monitoring	running while the reading fires	$\gamma = 0.97$	0.05 operating, 0.005–0.10 studied	duration-only controller, flexible fixtures
Admission control	admitting a configuration the reading forbids	one shot	operating budget	flexible over configurations
Maintenance	running past the intervention trigger	H=30 synthetic, H=120 C-MAPSS	expected cumulative cost per episode, five budgets	stationary randomized
Agent-service	each call to service k	$\gamma = 0.97$	0.05, 0.075, 0.1, 0.15	stationary randomized

$z_{t+1} \geq \mathrm{lvl}_k$ and 0 otherwise. drain moves to $s_\perp$. Reading ψ_k fires once its condition has held for its duration and charges running while it fires, $c_{\psi_k}(s_t, a_t) = \mathbf{1}\{a_t = \mathsf{run}\}\,\mathbf{1}\{m_{k,t} \geq h_k\}$. Different readings of "stays high" are different pairs (θ_k, h_k), hence different costs on the same state space. Running serves the workload, $r(s_t, \mathsf{run}) = 1$, and draining throttles it, $r(s_t, \mathsf{drain}) = r(s_\perp, \cdot) = 0.3$. Return and cost are normalized discounted occupancies, so the operating budget $d = 0.05$ permits five percent of discounted time running while ψ_k fires. The initial-state distribution μ_0 splits mass between a fresh spike and a fresh ramp.

Agent-service domain. In our agent-service benchmark the readings disagree on which external service a call limit covers. A tool-using agent handles user requests that decompose into sub-tasks, each resolved by one call to one of K external services (search, database, messaging, translation, of which $K \in \{2, 3, 4\}$ uses the first K). The rule "keep calls to the external service within the plan's limit" does not say which service it means, so reading ψ_k charges one unit per call to service k. The readings fire on disjoint action types and none is stricter than another, a crossing antichain of width K. The session ends with probability $1 - \gamma$ per step. The state is the number of open sub-tasks per service and the actions are a call to service k or a step of local work. A call resolving a primary sub-task is worth $0.8 + 0.02\,j$ with j sub-tasks remaining, so the constrained optimum is a deterministic cutoff. A resolved auxiliary sub-task is worth 1, a call with nothing to resolve is worth 0 and still charged, and local work is worth 0.2. At session start the primary service has 6–10 open sub-tasks (uniform) and each auxiliary 1 to 4, in three variants of auxiliary demand (light, medium, heavy).

Policy classes. Monitoring is studied under two policy classes, the sets of policies the learner may return, while each other domain uses one. The duration-only controller is the stationary policy class of the compiled monitoring model, observing the load level and the monitor counters and choosing run or drain. Its constrained optimum is therefore an occupancy linear program, over discounted state–action visit frequencies. The flexible class decides separately for every fixture of a rule. A fixture is a short synthetic telemetry scenario, a sustained level at a candidate threshold, a pulse of a given duration or a scope variant, on which each candidate reading either fires or not. A set of accepted fixtures is feasible under a reading when the reading's mean firing rate over them is at most d. Its value is the number of accepted fixtures, and the optimum is an integer program solved by exact enumeration. Admission control uses the same flexible class over configurations, an exact integer program, so its readings need not nest. The maintenance and agent-service domains use the stationary randomized class of Section 2, and their readings differ in the intervention trigger and in the service charged.

In the episodic maintenance domains d bounds expected cumulative per-episode cost, and in the discounted monitoring and agent-service models it bounds a normalized discounted occupancy rate in $[0, 1]$. The theory holds under both cost conventions, since it needs only $\Pi_{U_0}(d) \subseteq \Pi_\psi(d)$ and attainment of the optima. Cumulative units multiply everything by $1/(1-\gamma)$ and change no verdict or relative price.

B.3 ARROW IMPLEMENTATION AND EXPERIMENTAL PROTOCOLS

Exact implementation. When P is known and Π is the stationary randomized class, ARROW's decision is two linear programs over the normalized occupancy measure $x_\pi(s,a) = (1-\gamma)\,\mathbb{E}_\pi[\sum_{t\ge 0}\gamma^t \mathbf{1}\{s_t = s, a_t = a\}]$. The occupancies are the nonnegative solutions of the flow equations $\sum_a x(s,a) = (1-\gamma)\mu_0(s) + \gamma \sum_{s',a'} P(s \mid s',a')\, x(s',a')$, a polytope $\mathcal{X}(P)$ on which return and costs are linear. The value program $V_\psi(d) = \max\{J_r(x) : x \in \mathcal{X}(P),\ J_{c_\psi}(x) \le d\}$ gives the value test, which rules ψ out when its best return exceeds the full set's. The face program of Equation (2) searches the acceptable policies under ψ, its feasible policies within ε of $V_\psi(d)$, for the largest cost a competitor can reach. When that cost exceeds d, its maximizing occupancy, read as the policy $\pi_x(a \mid s) = x(s,a)/\sum_{a'} x(s,a')$, is an executable counterexample. Restricted policy classes use a representation that respects them, and for $|C| = K$ readings the decision costs one value program and $K-1$ face programs per tested reading. The ambiguity margin $\Gamma^\varepsilon_\psi(C,d)$ of Equation (1), the most an acceptable policy under ψ can exceed the budget on a competitor, is nondecreasing in ε. The sufficient tolerances therefore form an interval ending at the sufficiency radius $\varepsilon^\star_\psi(C,d)$ of Equation (3). Every certified reading is safe, so the order in which readings are tried affects only the return we keep. With a known model they are tried in decreasing $V_\psi(d)$, ties broken by the SCORE-KEEP plausibility score and then by index, and offline in decreasing order of the confidence-bounded value the certificate computes.

Question selection. REFINE, the question-selection step of Algorithm 1, asks questions by hyperedge cutting over decision regions, and certified regions preserve the stopping guarantee. The region $R^\varepsilon_\psi(d) = \{\phi \in U_0 : \sup_{\pi \in F^\varepsilon_\psi(d)} J_{c_\phi}(\pi) \le d\}$ collects the retained readings under which every acceptable policy under ψ is safe, one face program per pair. A reading ψ is an *anchor* for the survivors U_ξ, the readings consistent with the first ξ answers, when $U_\xi \subseteq R^\varepsilon_\psi(d)$. Any anchor gives the SAFE stop of Theorem 4.3, where every acceptable policy satisfies the governing reading under truthful answers. A surviving anchor gives the FREE stop, at which also $V_\psi(d) = V_{U_\xi}(d)$. When no anchor exists, the sets of readings that block sufficiency, the hyperedges, are the minimal subsets of U_ξ contained in no region. Hyperedge cutting weights each by the total prior probability of its members, scores a question by the expected weight it splits, and asks the maximizer. Regions estimated from data make the selector a heuristic.

The end-to-end pipeline. Algorithm 1 embeds ARROW in the full path from a written rule to a deployed policy. A generator G turns the rule ℓ and its context X_ℓ into the pool Ψ_ℓ. In structured families an admission predicate $\mathrm{Adm}(\ell, X_\ell) \in \{0,1\}$ marks rules in the supported family, whose parser returns $\psi^\dagger_{\mathrm{parsed}}$, its executable version of the governing reading $\psi^\dagger$ (the reading that actually applies). Retention returns U_0 by SAFE-KEEP, which drops readings implied by others when a sound implication relation $\succeq$ is available (Appendix C.2.3). It may instead use SCORE-KEEP, which drops readings scoring below a conformal plausibility threshold (Appendix C.1.4), or keep everything (keep-all). PROTECT is the learner. It receives $\{c_\phi\}_{\phi \in C}$ if it accepts one cost per reading, and otherwise the single-cost surrogate, the pointwise maximum $c_C(s,a) = \max_{\phi \in C} c_\phi(s,a)$ (Appendix C.2.4).

Two deployment routes follow the protection step, one testing the returned policy $\hat\pi$ on fresh episodes and one certifying it from the log alone. On the checked route CHECK evaluates $\hat\pi$ on episodes independent of the log $\mathcal{D}$ and returns an upper confidence bound $\mathcal{B}(\hat\pi, U_\Xi)$ on $\max_{\phi \in U_\Xi} J_{c_\phi}(\hat\pi)$ that fails with probability at most δ_{ev}, where U_Ξ are the survivors after the last of Ξ answers. The pipeline deploys when $\mathcal{B}(\hat\pi, U_\Xi) \le d$, for any downstream learner. The certified-offline route is available in a structured family. There the certified planner deploys, on an offline confidence event $\mathcal{E}_{\mathrm{off}}$ of probability at least $1-\delta_{\mathrm{off}}$, only policies with $\hat\pi \in \Pi_C(d)$ for the set C handed to CERTIFIED-PROTECT, the SAFE-KEEP retained set U_0 in every reported run. The planner is an occupancy LP on the estimated model over the *covered pairs*, the state–action pairs observed at least five times. Its costs are inflated by the simulation-lemma penalty $\gamma(1-\gamma)^{-1} b(s,a)$, where $b(s,a) = \alpha_{z(s,a)}$ is the L_1 confidence radius of Weissman et al. (2003) for the next-state distribution of row $z(s,a)$. This charge is a factor $2/\mathrm{span}(c)$ above the bound of Lemma S1 and so conservative, which makes any solution with penalized cost at most d feasible in the true model.

Algorithm 1 End-to-end pipeline from a written rule ℓ to a deployed policy $\hat{\pi}$ or an abstention. Lines 7–23 are ARROW itself (Section 3.1).

Input: rule ℓ, context X_ℓ, generator G, questions Q, question budget B_q, log $\mathcal{D}$ and optionally the model P, class Π, budget d, tolerance ε, levels $\delta_{\mathrm{sem}}, \delta_{\mathrm{dec}}, \delta_{\mathrm{ev}}, \delta_{\mathrm{off}}$, retention $\in$ {safe, score, all} and deployment $\in$ {checked, certified-offline}, with checked paired with score or all and certified-offline paired with safe

1: $\Psi_\ell \leftarrow G(\ell, X_\ell)$ ▷ executable candidate readings
2: **if** $\Psi_\ell = \emptyset$, or the rule lies outside the structured family in certified-offline mode, **then return** abstain
3: **if** retention = safe **then** $U_0 \leftarrow$ SAFE-KEEP(Ψ_ℓ) ▷ same allowed policies as Ψ_ℓ
4: **else if** retention = score **then** $U_0 \leftarrow$ SCORE-KEEP$(\Psi_\ell, \delta_{\mathrm{sem}})$ **else** $U_0 \leftarrow \Psi_\ell$
5: **if** $U_0 = \emptyset$ **then return** abstain
6: **for** $\xi = 0, 1, \ldots, B_q$ **do**
7: verdict $\leftarrow$ NOT CERTIFIED ▷ **ARROW begins**: the decision of Section 3.1 on the candidates U_ξ
8: **for all** $\psi \in U_\xi$ **do**
9: **if** $\Pi_\psi(d) = \emptyset$ **then** skip ψ ▷ ψ infeasible; offline, over policies supported by $\mathcal{D}$
10: **if** P is known **then**
11: $V_\psi(d) \leftarrow \max\{J_r(\pi) : \pi \in \Pi_\psi(d)\}$ ▷ best return under ψ, one linear program
12: **for all** $\phi \in U_\xi \setminus \{\psi\}$ **do**
13: $\pi_\phi \leftarrow \arg\max\{J_{c_\phi}(\pi) : \pi \in \Pi_\psi(d),\ J_r(\pi) \geq V_\psi(d) - \varepsilon\}$ ▷ face program, Equation (2)
14: **end for**
15: **if** $U_\xi = \{\psi\}$ **then** $\Gamma_\psi \leftarrow -\infty$ **else** $\phi^\star_\psi \leftarrow \arg\max_{\phi \in U_\xi \setminus \{\psi\}} J_{c_\phi}(\pi_\phi)$, $\pi^\star_\psi \leftarrow \pi_{\phi^\star_\psi}$, $\Gamma_\psi \leftarrow J_{c_{\phi^\star_\psi}}(\pi^\star_\psi) - d$ ▷ margin, Equation (1); no competitor when $U_\xi = \{\psi\}$
16: **else**
17: $\Gamma_\psi \leftarrow \overline{\Gamma}^{\varepsilon}_{\mathcal{D},\psi}(U_\xi, d)$, an upper confidence bound on $\Gamma^{\varepsilon}_{\psi}(U_\xi, d)$ from $\mathcal{D}$
18: **end if**
19: **if** $\Gamma_\psi \leq 0$ **then**
20: verdict $\leftarrow$ SUFFICIENT, $\psi^\star \leftarrow \psi$, **break**
21: **end if**
22: **end for**
23: **if** verdict $\neq$ SUFFICIENT and P is known **then** verdict $\leftarrow$ NO SINGLE READING, with the counterexamples $\{(\phi^\star_\psi, \pi^\star_\psi)\}$ for every tested ψ
▷ **ARROW ends**
24: **if** verdict = SUFFICIENT **then** $C \leftarrow \{\psi^\star\}$, $\Xi \leftarrow \xi$, **break** ▷ train under one reading
25: **if** $\xi = B_q$ or no $q \in \mathsf{Q}$ separates U_ξ **then** $C \leftarrow U_\xi$, $\Xi \leftarrow \xi$, **break** ▷ train under the remaining set
26: $q_\xi \leftarrow$ REFINE(U_ξ, Q), a question that splits readings still blocking sufficiency; observe y_ξ, $U_{\xi+1} \leftarrow \{\phi \in U_\xi : q_\xi(\phi) = y_\xi\}$, **abstain if** $U_{\xi+1} = \emptyset$
27: **end for**
28: **if** deployment = certified-offline **then**
29: **if** verdict $\neq$ SUFFICIENT or training under C is not certified on $\mathcal{E}_{\mathrm{off}}$ to satisfy all of U_0 **then** $C \leftarrow U_0$ ▷ Theorem 4.4(b) fallback
30: $\hat{\pi} \leftarrow$ CERTIFIED-PROTECT$(\mathcal{D}, C, d, \delta_{\mathrm{off}})$
31: **return** $\hat{\pi}$ if its fixed-data certificate holds, **else** abstain
32: **end if**
33: $\hat{\pi} \leftarrow$ PROTECT(C), restricted to the covered pairs when the offline certificate was used
34: **return** $\hat{\pi}$ if CHECK$(\hat{\pi}, U_\Xi, \delta_{\mathrm{ev}}) \leq d$, **else** abstain

Each retention rule pairs with the route whose guarantee it supports. Keep-all and SCORE-KEEP pair with the checked route (Theorem 4.4(a), safe for $\psi^\dagger$). SAFE-KEEP pairs only with the certified-offline route (Theorem 4.4(b), safe for $\psi^\dagger_{\mathrm{parsed}}$). The reason is that it may retain a covering reading, one whose constraint implies that of $\psi^\dagger$, in place of $\psi^\dagger$. A later truthful answer may remove that reading, after which CHECK over the survivors says nothing about $\psi^\dagger$.

Study protocols. The exact and fixed-data studies share two conventions, every exact result is an occupancy linear program on the declared model and every learned policy is scored exactly on the true model. Table 7 gives the population and settings of each exact and fixed-data study. In the comparison of alternatives a reading counts as unsafe when its most favorable optimal policy (monitoring) or its returned policy (admission) violates a retained reading. Exact analyses run on one core of an Apple M4 with 16 GB (HiGHS via SciPy 1.13.1) in seconds, the compiled rules of Table 1 taking 420 value, 126 face and 2,700 tie-break programs in 3.5 s. The learners of Appendix D.3 run on CPU and the released DSRL/OSRL learners on one NVIDIA V100 per run. Compilation is what

SAFE-KEEP protects by removing readings, since each reading adds a monitor counter and the joint monitor state grows from 32,806 states at $K{=}4$ to 387,420,490 at $K{=}8$.

Table 7: **Protocols of the exact and fixed-data studies.** Learner settings are in Table 8.

Study	Population	Settings	Repeats
Exact geometry and alternatives	28 compiled rules, 60 analyzable rules, 44 admission pools	known model, d from 0.005 to 0.10, alternatives at the operating budget	none, exact
Clarification	44 admission pools, 87 flexible monitoring pools, 28 compiled rules at $d{=}0.005$	up to four questions, recovery measured at $d{=}0.05$	20 draws for the random rule
Fixed-data certification	28 compiled rules, shared load chain in stationary proportions	$\varepsilon{=}0.01$, $\delta_{\mathrm{dec}}{=}0.05$, $d \in \{0.02, 0.05, 0.10\}$, n from 10^3 to 10^8 in half-decades	10 draws per size, 50 more for the 5 hardest readings at 10^5–10^6
SAFE-KEEP audit and certified-offline deployment	579 multi-reading families, 581 admitted rules	every removal checked by implication and LP redundancy, 4 budgets, penalized planner	400 seeded pools

Clarification protocol. The clarification study compares five question-selection rules against a truthful oracle, one that answers as the governing reading would. Questions are chosen by hyperedge cutting, by the price-guided rule (ranking questions by their effect on return), by the witness-restricted rule, by the most balanced split or at random. The witness-restricted rule is price-guided among the fixtures that separate the anchor whose value ties the set's optimum from its worst violator. After each answer we record whether the reading is identified ($|U_\xi| = 1$), whether a survivor ties the set's optimal value, and whether the FREE or the SAFE stop is reached, the first when a surviving reading is sufficient for U_ξ and the second when any reading of U_0 is. Return recovered after ξ questions is $(V_{U_\xi}(d) - V_{U_0}(d))/(V_{\psi^\dagger}(d) - V_{U_0}(d))$, where $\psi^\dagger$ is the governing reading. No human answered questions.

Fixed-data protocol. The fixed-data study runs three certificates on identical draws of the shared load chain and compares each with an information floor on the data any sound procedure needs. All three use per-row confidence radii, a row being the next-state distribution of one state–action pair. A reading counts as reliably certified at a log size when at least 90% of the draws certify it. The certificates are the generic uniform certificate (Corollary S3), the row-specific occupancy-weighted certificate and the robust-dual certificate (Theorem 4.2), which requires the dual proof of sufficiency to hold for every transition model in the confidence set. The information floor needs a chain under which the decision reverses, which directional bisection and local refinement find as a Q with $\Gamma^{\varepsilon}_{\psi,Q} > 0$. Its weighted relative entropy $\bar{I}_\psi$ upper-bounds the decision-information radius of Equation (4), the smallest weighted Kullback–Leibler divergence that reverses the decision. Hence $\mathrm{kl}(1-\delta_{\mathrm{dec}}, \delta_{\mathrm{dec}})/\bar{I}_\psi$ lower-bounds the transitions any δ_{dec}-sound procedure (one certifying with probability at most δ_{dec} under every reversing chain) needs to certify ψ with probability at least $1 - \delta_{\mathrm{dec}}$.

Learning experiments. On each sufficient instance, one with a certified reading, we train six learners that accept one cost signal under either the single-cost surrogate c_C or ARROW's selected reading, holding log, seed, learner and cost limit d fixed. A Lagrangian learner subtracts the cost, scaled by a multiplier (the dual variable), from the reward and raises the multiplier until the budget holds. BCQ-FQI-Lagrangian does so in fitted Q-iteration restricted to logged actions, PID-Lagrangian tunes the multiplier with a proportional-integral-derivative controller, and O3SRL picks it with the EXP3 bandit. CPQ is constraints-penalized Q-learning, CAPS switches among policies trained at different cost limits, and CQL-Lagrangian is conservative Q-learning. Logs contain $n = 20{,}000$ transitions (2,000 in a second sweep over the 5 tabular learners, CQL-Lagrangian excepted), 10 seeds per cell, from a $0.7/0.3$ mixture of the primary-reading optimum and a uniform policy, so the selected-reading target is the log mode. Actions are restricted to log support. Search uses the empirical model with the true initial distribution, and all returned policies are evaluated

exactly on the true model. Table 8 gives learner-specific settings. At $n = 2{,}000$, 1 PID-Lagrangian run found no offline-feasible iterate and returned its best-safety iterate, counted as unsafe. Native multi-constraint experiments, with learners that accept one cost per reading, use the same instances and logs at $n \in \{2{,}000, 5{,}000, 20{,}000, 50{,}000\}$, 10 seeds, with the learner-aware tolerance $\varepsilon'_{\mathcal{L}}(n)$ of Proposition S5 estimated from 5 disjoint held-out seeds. Comparisons use the instances certified at that tolerance, and otherwise the 9 instances sufficient at ε=0.01. CHECK uses n_{ev}=20,000, δ_{ev}=0.05 and empirical-Bernstein (variance-adaptive) bounds (Audibert et al., 2009; Maurer & Pontil, 2009), Bonferroni-corrected by splitting the error level evenly across readings. It evaluates all 3,960 policies, wrong-reading policies included. Its multinomial simulation is identical in law to fresh evaluation because the costs are disjoint-action indicators.

Table 8: **Learner settings of the agent-service study**, shared by the six single-cost learners and both training arms. The native learners of Table 18 keep the maintenance-domain gains and limit grid.

Component	Setting
Common	our own implementations, $K + 1$ actions, charged-service indicator cost, same log, seed and d across arms
Multiplier grid	$0,\ 0.01,\ 0.03,\ \ldots,\ 1.19$, then $2, 5, 10, 20, 40, 80, 160, 320$
O3SRL	EXP3 over the grid points ≤ 5, four rounds per arm, returns an iterate, matches BCQ-FQI-Lagrangian on 178/180 runs at $n = 20{,}000$
CPQ, CAPS	ratio-1.1 limit grid from $1/(1-\gamma)$ to 1, then $0.8, 0.5, 0.3, 0.15, 0.05, 0$
PID-Lagrangian	gains $(0.8, 0.6, 0.4)$, our implementation's defaults divided by 100, 96 dual steps, best offline-feasible iterate
CQL-Lagrangian	$\alpha = 1$, the selected-reading $\lambda = 0$ policy is the log mode and is selected on 82/90 runs
CAPS	published critic, filter and select construction with a cost-to-go threshold
Native	vector Lagrangian FQI, vector PID-Lagrangian FQI, vector CPQ, supported-pair occupancy LP

Real requirements. On the ARTEMIS requirements a candidate counts as plausible when it is LTL-equivalent to an expert-plausible reading. Each requirement's expert-plausible readings are formed from the product of ARTEMIS's option lists, expanded as in ARTEMIS's accuracy computation (unspecified durations set to one tick) and translated to LTL with its template dictionary. Expert sets contain up to 56 readings. Each candidate pool is the union of all available archives, totaling 190 samples per requirement. We use DeepSTL (FT) (He et al., 2022), nl2spec (Cosler et al., 2023), NL2TL+ and NL2TL+ (FT) (Chen et al., 2023), SynthTL (Mendoza et al., 2024), directTL, directTL-t, ARTEMIS and ARTEMIS + reflection (Mendoza et al., 2026). The fine-tuned and NL2TL+ adaptations are those released with ARTEMIS. A method is evaluated by its first sample, with coverage defined by LTL equivalence to any expert-plausible reading. Table 3's parenthesized entries use ARTEMIS's any-of-ten metric (50 samples for ARTEMIS with reflection). The table's bottom rows, a self-consistency vote that commits to the most frequent reading, the pool of all distinct samples and SCORE-KEEP, use the same 190 archived samples, with no new model calls. Candidates are merged into equivalence classes and matched to expert readings by exact LTL equivalence using a formula-progression Büchi construction (Couvreur, 1999), preceded by a bounded-lasso inequivalence check. On 34,400 translations, our "equivalent to some expert reading" decision agrees with the verdicts of the Spot LTL library (Duret-Lutz et al., 2022) released with ARTEMIS on 99.87% of cases. The 44 disagreements are all cases where our checker declines an equivalence. Pairs exceeding the automaton budget are treated as inequivalent (227 pairs).

SCORE-KEEP retains the equivalence classes whose pooled frequency clears a conformal threshold, so that a plausible reading survives with probability at least 0.9 whenever one is proposed. It uses leave-one-unit-out split conformal prediction, a unit being one requirement, at $\delta_{\text{sem}} = 0.10$, conditioned on the governing reading being proposed, as in Appendix C.1.4. The threshold thus comes from the conformity scores of the other units. Units with no plausible candidate are generation misses and contribute to the generation-miss rate $\hat{\delta}_{\text{gen}}$, not the conformal threshold. For unit ℓ, with P_ℓ its expert-plausible readings and U_ℓ its retained set, the conformity score is the largest frequency among its expert-plausible classes, giving $\Pr[P_\ell \cap U_\ell \neq \emptyset \mid \text{proposed}] \geq 0.9$. We report

any-plausible coverage, expert-reading recall $|P_\ell \cap U_\ell|/|P_\ell|$, and the median number of retained equivalence classes per requirement.

C THEORY AND PROOFS

Appendix C.1 proves the four theorems of Section 4 in order. Appendix C.2 collects the supporting results, the linear-program form of the fixed-tolerance test with its dual certificate, composition with an approximate learner, the dominance relation behind SAFE-KEEP (the rule that drops readings enforced by others), and the single-cost surrogate. Results stated only in the appendix are numbered S1 to S5, and the four main-text theorems keep their numbers.

C.1 PROOFS OF THE MAIN RESULTS

This subsection proves the four main-text theorems in order, after fixing their shared notation. They are Theorem 4.1 (the sufficiency radius), Theorem 4.2 (certification from fixed data, together with its completeness theorem and the worst-case margin law), Theorem 4.3 (safe stopping) and Theorem 4.4 (deployment safety). Throughout, Π is the policy class of Section 2 (the policies the learner may return), U_0 the retained set of Section 3 (the candidate readings kept after generation and filtering) and U_ξ the readings still consistent with ξ clarification answers. For a budget d, $\Pi_\psi(d) = \{\pi \in \Pi : J_{c_\psi}(\pi) \le d\}$ are the policies satisfying ψ, $\Pi_{U_0}(d) = \bigcap_{\psi \in U_0} \Pi_\psi(d)$ is full-set protection, and $V_\psi(d) = \sup_{\Pi_\psi(d)} J_r$ and $V_{U_0}(d) = \sup_{\Pi_{U_0}(d)} J_r$ are the best returns. The containment $\Pi_{U_0}(d) \subseteq \Pi_\psi(d)$ gives $V_\psi(d) \ge V_{U_0}(d)$, so the value test $\min_{\psi \in U_0}[V_\psi(d) - V_{U_0}(d)]$ is a minimum of nonnegative terms. In the compiled regime, where the readings admit an exact finite model and which the proofs call the *LP case*, both programs are linear programs over a compact occupancy polytope (the discounted state–action visit frequencies of Appendix B.3), so the suprema are attained whenever the feasible set is nonempty. If $\Pi_{U_0}(d) = \emptyset$, ARROW abstains.

C.1.1 THEOREM 4.1: THE SUFFICIENCY RADIUS

The sufficient tolerances of a reading form an interval ending at the sufficiency radius, and this part fixes the notation for proving it. We call the acceptable policies $F^\varepsilon_\psi(d)$ of Section 2.2 the *ε-face* of ψ, and every other reading of the current set a *competitor*. Dropping ε from a symbol means $\varepsilon = 0$ throughout the appendix. Thus $F_\psi(d) = F^0_\psi(d)$ is the exact optimal face and $\Gamma_\psi(d) = \Gamma^0_\psi(U_0, d)$ the exact margin, the ambiguity margin of Equation (1) at zero tolerance. The maximum in Equation (1) runs over $U_0 \setminus \{\psi\}$. Including ψ changes no verdict but would raise any negative margin to zero whenever the budget binds. We write $W^\varepsilon_\psi(U_0; d) = \Gamma^\varepsilon_\psi(U_0, d) + d$ for the worst competing cost over the face and $F^\varepsilon_{U_0}(d) = \{\pi \in \Pi_{U_0}(d) : J_r(\pi) \ge V_{U_0}(d) - \varepsilon\}$. Since the faces grow with ε, $\Gamma^\varepsilon_\psi(C, d)$ is nondecreasing in ε, and part (a) identifies the right end of the sufficient interval as the radius $\varepsilon^\star_\psi(C, d)$ of Equation (3).

The exact case and the value test. At $\varepsilon = 0$, when the optima are attained, three forms of sufficiency coincide and the value test is strictly weaker than all of them. The first is that every ψ-optimal policy is safe under the full retained set, $F_\psi(d) \subseteq \Pi_{U_0}(d)$. The second is that optimizing under ψ and under U_0 gives the same optimal set, $F_\psi(d) = F_{U_0}(d)$. The third is $\Gamma_\psi(d) \le 0$. The first and third are one statement, read pointwise or through the supremum. The first implies the second because a ψ-optimal policy is then U_0-feasible with return $V_\psi(d) \ge V_{U_0}(d)$, forcing $V_\psi(d) = V_{U_0}(d)$ and equality of the two faces. The second implies the first since $F_{U_0}(d) \subseteq \Pi_{U_0}(d)$. Each implies $V_\psi(d) = V_{U_0}(d)$, but value equality alone implies none of them. Take three actions a, b, c of rewards $1, 1, 0$, budget $d = 0$, ψ admitting all three and ϕ excluding b. Then $V_\psi = V_{\{\psi,\phi\}} = 1$ while $F_\psi = \{a, b\} \ne \{a\} = F_{\{\psi,\phi\}}$. The *value test* of Section 3.1, $\min_{\psi \in U_0}[V_\psi(d) - V_{U_0}(d)] = 0$, is exactly the statement that some ψ-optimal policy is feasible for every retained reading. The containment $\Pi_{U_0}(d) \subseteq \Pi_\psi(d)$ gives $V_\psi(d) \ge V_{U_0}(d)$, with equality if and only if a full-set optimum, which is ψ-feasible, attains $V_\psi(d)$. The value test reads only optimal values, which are unique even when optimal policies are not, so no solver's choice among tied optima changes its verdict. A solver's returned point may be U_0-feasible or not. This is why a passed value test licenses only equality of optimal values, and why Theorem 4.4 checks whatever policy the optimizer returns.

Sufficiency is not monotone in the policy class, so enlarging the class can remove a sufficient reading and then restore one. Take actions a, b, c, z, rewards $2, 2, 1, 3$, $d = 0$, ψ_1 admitting $\{a, c, z\}$ and ψ_2 admitting $\{b, c, z\}$. The classes $\{c\} \subset \{a, b, c\} \subset \{a, b, c, z\}$ make both readings sufficient, then neither ($F_{\psi_1} = \{a\}$, $F_{\psi_2} = \{b\}$, $F_{\{\psi_1,\psi_2\}} = \{c\}$), then both again ($F = \{z\}$). This is the class dependence of Section 5.1 in miniature.

Proof of Theorem 4.1. Let $C \subseteq U$ and $\psi \in U$ with $\Pi_\psi(d) \neq \emptyset$, and write $F^\varepsilon_\psi = F^\varepsilon_\psi(d)$. Let $B = \Pi_\psi(d) \setminus \Pi_C(d)$ be the ψ-feasible policies that are unsafe under some reading of C, $R = R^{\text{unsafe}}_\psi(C, d) = \sup_{\pi \in B} J_r(\pi)$ and $\varepsilon^\star = \varepsilon^\star_\psi(C, d) = V_\psi(d) - R$. Every argument of this proof is over the declared policy class Π, and no representation of policies is used.

$$
\begin{aligned}
B &\subseteq \Pi_\psi(d) && \text{[definition of } B\text{]} \\
R &\leq \sup_{\pi \in \Pi_\psi(d)} J_r(\pi) = V_\psi(d) && \text{[supremum over a subset]} \\
\varepsilon^\star &\in [0, \infty] && \text{[previous line; } R = -\infty \text{ when } B = \emptyset\text{]}
\end{aligned}
$$

Returns are bounded, so R is finite whenever $B \neq \emptyset$, and $\varepsilon^\star = +\infty$ exactly when $B = \emptyset$.

(a) Below the radius: $\varepsilon < \varepsilon^\star$. If $B = \emptyset$,

$$
\begin{aligned}
F^\varepsilon_\psi &\subseteq \Pi_\psi(d) && \text{[definition of the face]} \\
\Pi_\psi(d) &= \Pi_\psi(d) \cap \Pi_C(d) \subseteq \Pi_C(d) && \text{[} B = \emptyset\text{]}
\end{aligned}
$$

for every $\varepsilon \geq 0$, which is the claim with $\varepsilon^\star = +\infty$. Otherwise $\varepsilon^\star < \infty$. Take $\pi \in F^\varepsilon_\psi$ and suppose, for contradiction, that $\pi \notin \Pi_C(d)$.

$$
\begin{aligned}
\pi &\in B && \text{[} \pi \in \Pi_\psi(d) \text{ by face membership; } \pi \notin \Pi_C(d)\text{]} \\
J_r(\pi) &\leq R = V_\psi(d) - \varepsilon^\star && \text{[supremum; definition of } \varepsilon^\star\text{]} \\
J_r(\pi) &\geq V_\psi(d) - \varepsilon && \text{[} \pi \in F^\varepsilon_\psi\text{]} \\
V_\psi(d) - \varepsilon &> V_\psi(d) - \varepsilon^\star && \text{[} \varepsilon < \varepsilon^\star\text{]}
\end{aligned}
$$

The last three lines are contradictory, so $\pi \in \Pi_C(d)$ and $F^\varepsilon_\psi \subseteq \Pi_C(d)$.

(a) Above the radius: $\varepsilon > \varepsilon^\star$. Then $\varepsilon^\star < \infty$, so $B \neq \emptyset$ and R is finite.

$$
\begin{aligned}
0 &< \delta < \varepsilon - \varepsilon^\star && \text{[choice of } \delta\text{]} \\
\exists\, \pi_\delta \in B :\ J_r(\pi_\delta) &> R - \delta && \text{[definition of the supremum]} \\
&= V_\psi(d) - \varepsilon^\star - \delta && \text{[definition of } \varepsilon^\star\text{]} \\
&> V_\psi(d) - \varepsilon && \text{[choice of } \delta\text{]} \\
\pi_\delta &\in F^\varepsilon_\psi && \text{[} \pi_\delta \in \Pi_\psi(d) \text{ and the previous line]} \\
\pi_\delta &\notin \Pi_C(d) && \text{[} \pi_\delta \in B\text{]}
\end{aligned}
$$

so $F^\varepsilon_\psi \not\subseteq \Pi_C(d)$.

(a) At the radius: $\varepsilon = \varepsilon^\star < \infty$. If some $\bar{\pi} \in B$ attains R,

$$
\begin{aligned}
J_r(\bar{\pi}) &= R = V_\psi(d) - \varepsilon^\star && \text{[attainment]} \\
\bar{\pi} &\in F^{\varepsilon^\star}_\psi \setminus \Pi_C(d) && \text{[} \bar{\pi} \in \Pi_\psi(d); \bar{\pi} \in B\text{]}
\end{aligned}
$$

and the inclusion fails. If no unsafe policy attains R, then for every $\pi \in B$

$$
\begin{aligned}
J_r(\pi) &< R = V_\psi(d) - \varepsilon^\star && \text{[supremum not attained]} \\
\pi &\notin F^{\varepsilon^\star}_\psi && \text{[face requires } J_r(\pi) \geq V_\psi(d) - \varepsilon^\star\text{]} \\
F^{\varepsilon^\star}_\psi &\subseteq \Pi_\psi(d) \setminus B = \Pi_\psi(d) \cap \Pi_C(d) \subseteq \Pi_C(d) && \text{[previous line; definition of } B\text{]}
\end{aligned}
$$

and the inclusion holds. Hence at $\varepsilon = \varepsilon^\star$ the inclusion holds if and only if no unsafe policy attains R. Together with part (b) and the monotonicity of $\Gamma^\varepsilon_\psi(C, d)$ in ε, the tolerances with $\Gamma^\varepsilon_\psi(C, d) \leq 0$ form the interval $[0, \varepsilon^\star)$ or $[0, \varepsilon^\star]$, so $\varepsilon^\star$ is their supremum whenever that interval is nonempty. When $\varepsilon^\star = 0$ and an unsafe policy attains R, no tolerance is sufficient, the exact-optimum case included.

(b) The sign of the margin decides containment.

$$\begin{aligned}
\Gamma^{\varepsilon}_{\psi}(C,d) \le 0 &\iff \sup_{F^{\varepsilon}_{\psi}} J_{c_\phi} \le d \ \ \forall \phi \in C \setminus \{\psi\} && \text{[Equation (1); } \max \emptyset = -\infty\text{]}\\
&\iff J_{c_\phi}(\pi) \le d \ \ \forall \phi \in C \setminus \{\psi\},\ \forall \pi \in F^{\varepsilon}_{\psi} && \text{[a sup is } \le d \text{ iff every member is]}\\
&\iff F^{\varepsilon}_{\psi} \subseteq \textstyle\bigcap_{\phi \in C} \Pi_\phi(d) && \text{[} \pi \in \Pi_\psi(d) \text{ already, so add } \phi = \psi\text{]}\\
&\iff F^{\varepsilon}_{\psi} \subseteq \Pi_C(d) && \text{[definition of } \Pi_C\text{]}
\end{aligned}$$

No supremum needs to be attained.

A positive margin yields a counterexample.

$$\begin{aligned}
\Gamma^{\varepsilon}_{\psi}(C,d) > 0 \ \Rightarrow\ &\sup_{\pi \in F^{\varepsilon}_{\psi}} J_{c_{\phi^\star}}(\pi) > d \ \text{ for the maximizing } \phi^\star \in C \setminus \{\psi\} && \text{[Equation (1)]}\\
\Rightarrow\ &\exists\, \pi^\star \in F^{\varepsilon}_{\psi} :\ J_{c_{\phi^\star}}(\pi^\star) > d && \text{[definition of the supremum]}
\end{aligned}$$

In the LP case the supremum is attained and $\pi^\star$ is the maximizer of Equation (2). Part (b) uses no attainment and no representation of Π. It is the fixed-tolerance decision ARROW computes (lines 7–23 of Algorithm 1).

(c) Containment gives value equality when $\psi \in C$. The value statement uses $\varepsilon > 0$, or $\varepsilon = 0$ with $V_\psi(d)$ attained. Attainment holds in particular for the stationary randomized classes of the linear-program results, whose occupancy sets are compact, and for the finite fixture classes of Appendix B.2. The idea is to approximate $V_\psi(d)$ within ρ, because any policy that comes within $\rho \le \varepsilon$ of the optimum already lies in the face, so containment applies to it.

$$\begin{aligned}
&\rho \in (0, \varepsilon], \text{ or } \rho = 0 \text{ when } \varepsilon = 0 && \text{[} V_\psi(d) \text{ attained in the second case]}\\
\exists\, \pi_\rho \in \Pi_\psi(d) :\ &J_r(\pi_\rho) \ge V_\psi(d) - \rho && \text{[definition of the supremum; the maximizer when } \rho = 0\text{]}\\
&\pi_\rho \in F^{\varepsilon}_{\psi} && \text{[} J_r(\pi_\rho) \ge V_\psi(d) - \rho \ge V_\psi(d) - \varepsilon\text{]}\\
&\pi_\rho \in \Pi_C(d) && \text{[containment]}\\
&V_C(d) \ge J_r(\pi_\rho) \ge V_\psi(d) - \rho && \text{[} \pi_\rho \text{ competes in the } C \text{ program]}\\
&V_C(d) \ge V_\psi(d) && \text{[} \rho \downarrow 0\text{; immediate when } \rho = 0\text{]}\\
&V_C(d) \le V_\psi(d) && \text{[} \Pi_C(d) \subseteq \Pi_\psi(d) \text{ because } \psi \in C\text{]}
\end{aligned}$$

so $V_\psi(d) = V_C(d)$. The weaker bound $V_C(d) \ge V_\psi(d) - \varepsilon$, which follows from face membership alone, would not give equality.

(c) Face equality when $\psi \in C$. Write $F^{\varepsilon}_{C}(d) = \{\pi \in \Pi_C(d) : J_r(\pi) \ge V_C(d) - \varepsilon\}$.

$$\begin{aligned}
F^{\varepsilon}_{C}(d) &\subseteq F^{\varepsilon}_{\psi} && \text{[} \Pi_C(d) \subseteq \Pi_\psi(d) \text{ and value equality]}\\
F^{\varepsilon}_{\psi} &\subseteq \Pi_C(d) && \text{[containment]}\\
J_r(\pi) &\ge V_\psi(d) - \varepsilon = V_C(d) - \varepsilon \ \ \forall \pi \in F^{\varepsilon}_{\psi} && \text{[face membership and value equality]}\\
F^{\varepsilon}_{\psi} &\subseteq F^{\varepsilon}_{C}(d) && \text{[previous two lines]}
\end{aligned}$$

For $\psi \notin C$ parts (a) and (b) hold verbatim, the last line of the value argument fails, and only $V_C(d) \ge V_\psi(d)$ survives, because an attained ψ-optimum is C-feasible. ARROW (Algorithm 1, line 9) skips readings with $\Pi_\psi(d) = \emptyset$, for which the radius is undefined. □

The two extremes. The two extreme values of the radius have plain meanings. $\varepsilon^\star_\psi(C,d) = +\infty$ means that, at this budget and over this policy class, every policy feasible under ψ also satisfies C. For $\psi \in C$ this is $\Pi_\psi(d) = \Pi_C(d)$, a statement specific to this budget and policy class. $\varepsilon^\star_\psi(C,d) = 0$ is a zero-robustness boundary, at which no positive optimization-error tolerance is uniformly protected. Whether the exact optimum itself is safe is then decided by the boundary case of part (a).

Representation preservation. One certified reading can replace the whole retained set without changing the acceptable policies or their best return. When $\psi \in U_0$ is ε-sufficient for U_0, Theorem 4.1(c) gives $V_\psi(d) = V_{U_0}(d)$ and $F^{\varepsilon}_{\psi}(d) = F^{\varepsilon}_{U_0}(d)$ under its attainment condition, met in the LP case (the exact case is the equivalence shown above). This removes representation-induced conservatism, the return lost when a learner that accepts one cost must receive all readings merged into one conservative cost (Appendix C.2.4). It also settles which constraints the learner receives, the representation, before clarification stops. In the LP case W^{ε}_{ψ} is one face program, Equation (2), per competing reading and $F^{\varepsilon}_{\psi}(d)$ is a polytope, so the supremum is attained.

C.1.2 THEOREM 4.2: CERTIFICATION FROM FIXED DATA

The robust-dual certificate, ARROW's offline test (Algorithm 1, line 17), is sound, and every sound procedure needs a minimum amount of data. It robustifies only the pointwise inequalities of the dual witness of Proposition S4, the multipliers that prove a margin nonpositive. This subsubsection first fixes the confidence event and the simulation bound, then defines the certificate and proves Theorem 4.2(a), its soundness, and Theorem 4.2(b), a change-of-measure bound for every procedure. It then proves the completeness theorem of the certificate (Theorem S2), and closes with the generic uniform certificate (the comparison certificate of Appendix D.2) and the worst-case margin law that relates the two (Corollary S3). The certificates are run with a tolerance $\varepsilon > 0$, because at exact optimality the decision can be arbitrarily sensitive to the dynamics. This is the case of Theorem 4.2(b) in which the nearest decision-reversing model approaches the true one and the decision-information radius of Equation (4) vanishes, $I^\star_\psi \to 0$.

Rows, radii and the confidence event. With probability at least $1-\delta_{\mathrm{dec}}$ every empirical transition row lies within a known L_1 radius of the true row, and the offline proofs work on this event E. The unknown dynamics are the transition rows P_z, $z \in \mathcal{Z}$, each with m_z possible successors. Every state–action pair (s,a) draws its next state from row $z(s,a)$ through a known successor map. In the plain tabular case $z(s,a) = (s,a)$ with $m_z = |S|$. In the compiled monitoring models the rows are the three load levels and the monitor memory updates deterministically. The log follows the fixed-row design of Section 4.2, with $N_z = nw_z$ transitions recorded from row z, each an independent draw from P_z, and empirical row $\widehat{P}_z$ (any distribution on the m_z successors when $N_z = 0$). Set

$$\alpha_z = \min\Big\{2,\ \sqrt{2\big(m_z \ln 2 + \ln(|\mathcal{Z}|/\delta_{\mathrm{dec}})\big)/N_z}\Big\}, \qquad E = \{\|P_z - \widehat{P}_z\|_1 \le \alpha_z \ \ \forall z \in \mathcal{Z}\},$$

with $\alpha_z = 2$ when $N_z = 0$. The L_1 concentration inequality of Weissman et al. (2003) for empirical distributions and a union bound over the $|\mathcal{Z}|$ rows give $\Pr(E) \ge 1 - \delta_{\mathrm{dec}}$. For a log of length n sampled along a path, where N_z is random, the pre-drawn-sequence coupling and the union over counts used for the generic uniform certificate of Corollary S3 give the same event with $\alpha_z = \min\big\{2, \sqrt{2\big(m_z \ln 2 + \ln(|\mathcal{Z}|n/\delta_{\mathrm{dec}})\big)/N_z}\big\}$ for $N_z \ge 1$ and $\alpha_z = 2$ for $N_z = 0$. Every statement of the soundness proof is deterministic on E, so no further union over candidates, competitors or witnesses is needed. For a per-step signal f and an occupancy x write $B_f(x) = \frac{\gamma\,\mathrm{span}(f)}{2(1-\gamma)} \sum_{s,a} x(s,a)\,\alpha_{z(s,a)}$ for the simulation penalty, with $\mathrm{span}(f) = \max f - \min f$.

Lemma S1 (Occupancy-weighted simulation bound)**.** *On E, for every stationary policy π with empirical occupancy $\widehat{x}_\pi \in \mathcal{X}(\widehat{P})$ and every per-step f, $|J_f^P(\pi) - J_f^{\widehat{P}}(\pi)| \le B_f(\widehat{x}_\pi)$.*

Proof. Write $\tilde{V}^P$ for the unnormalized value of π under P, so $J_f^P(\pi) = (1-\gamma)\mu_0 \cdot \tilde{V}^P$ and $\tilde{V}^P \in [f_{\min}, f_{\max}]/(1-\gamma)$.

$$\begin{aligned}
\tilde{V}^{\widehat{P}} - \tilde{V}^P &= \gamma\,(I - \gamma\widehat{P}_\pi)^{-1}(\widehat{P}_\pi - P_\pi)\tilde{V}^P && \text{[Bellman equations of the two models]}\\
J_f^{\widehat{P}}(\pi) - J_f^P(\pi) &= \gamma \sum_{s,a} \widehat{x}_\pi(s,a)\,(\widehat{P} - P)(\cdot \mid s,a)\cdot \tilde{V}^P && \text{[}(1-\gamma)\mu_0^\top (I-\gamma\widehat{P}_\pi)^{-1} = \widehat{x}_\pi\text{]}\\
\big|(\widehat{P} - P)(\cdot \mid s,a)\cdot \tilde{V}^P\big| &\le \tfrac{\alpha_{z(s,a)}}{2}\,\mathrm{span}(\tilde{V}^P) && \text{[rows sum to one; on } E\text{]}\\
&\le \tfrac{\alpha_{z(s,a)}}{2}\,\frac{\mathrm{span}(f)}{1-\gamma} && \text{[range of } \tilde{V}^P\text{]}
\end{aligned}$$

and the last line weighted by $\widehat{x}_\pi$ and summed is $B_f(\widehat{x}_\pi)$. □

The robust-dual certificate. The certificate lower-bounds the best return under ψ, upper-bounds every competitor's worst cost over the face of ψ, and certifies ψ when every such bound is at most d. For $\psi \in C$ with $\Pi_\psi(d) \ne \emptyset$, step (1) sets $\underline{V}_\psi \leftarrow \max\{r \cdot x - B_r(x) : \ x \in \mathcal{X}(\widehat{P}),\ c_\psi \cdot x + B_{c_\psi}(x) \le d\}$ (one LP), or $\underline{V}_\psi \leftarrow 0$ if that program is infeasible. Step (2) solves, for every competitor $\phi \in C \setminus \{\psi\}$, the robust dual

$$\overline{W}_\phi \ = \ \min\ (1-\gamma)\mu_0 \cdot h + \mu d - \nu\big(\underline{V}_\psi - \varepsilon\big) \tag{5}$$

over a potential $h \in \mathbb{R}^S$ on states, $\mu, \nu \geq 0$ and $t \geq 0$, with μ the price of the budget constraint, ν that of the return floor and μ_0 the initial distribution of Section 2, subject to, for every (s, a),

$$c_\phi(s,a) \leq \mu c_\psi(s,a) - \nu r(s,a) + h(s) - \gamma \Big[\widehat{P}(\cdot \mid s,a) \cdot h + \tfrac{\alpha_{z(s,a)}}{2}\, t_{s,a} \Big],$$

and $t_{s,a} \geq h(s') - h(s'')$ for all s', s'' in the successor set $\mathrm{supp}(s,a)$. Step (3) certifies ψ iff $\overline{\Gamma} := \max_{\phi \neq \psi} \overline{W}_\phi - d \leq 0$. This $\overline{\Gamma}$ is the upper confidence bound on the ambiguity margin that ARROW tests (Algorithm 1, line 17). Rows with $\alpha_z = 2$ are charged at least the largest witness value among their successors, so they help a certificate only through the slack of the other terms. In the compiled monitoring models the only stochastic rows are the load levels, so B_f charges only the run action at the monitored states. A budget tightening η replaces d by $d - \eta$ in step (1) and in the objective of Equation (5) while the test $\overline{W}_\phi \leq d$ is unchanged, which certifies $F^\varepsilon_\psi(d - \eta) \subseteq \Pi_C(d)$.

Proof of Theorem 4.2(a), soundness. On E the certificate's lower value and dual witness stay valid for the true model, so every certified reading is ε-sufficient. Work on E. *The lower bound is valid.* Let $\widehat{x}$ attain $\underline{V}_\psi$ and $\pi_{\widehat{x}}$ be its policy.

$$\begin{aligned} J_{c_\psi}(\pi_{\widehat{x}}) \;&\leq\; c_\psi \cdot \widehat{x} + B_{c_\psi}(\widehat{x}) \;\leq\; d && \text{[Lemma S1; feasibility in step (1)]} \\ V_\psi(d) \;&\geq\; J_r(\pi_{\widehat{x}}) \;\geq\; r \cdot \widehat{x} - B_r(\widehat{x}) \;=\; \underline{V}_\psi && \text{[}\pi_{\widehat{x}} \in \Pi_\psi(d)\text{; Lemma S1]} \end{aligned}$$

and $\underline{V}_\psi = 0 \leq V_\psi(d)$ in the infeasible case because $r \geq 0$. *The witness holds for the true model.* Let (h, μ, ν, t) be optimal in Equation (5) for competitor ϕ. For every (s, a),

$$\begin{aligned} P(\cdot \mid s,a) \cdot h \;&\leq\; \widehat{P}(\cdot \mid s,a) \cdot h + \tfrac{\alpha_{z(s,a)}}{2}\, \mathrm{span}_{\mathrm{supp}(s,a)}(h) && \text{[}\|P - \widehat{P}\|_1 \leq \alpha\text{, both rows sum to one]} \\ &\leq\; \widehat{P}(\cdot \mid s,a) \cdot h + \tfrac{\alpha_{z(s,a)}}{2}\, t_{s,a} && \text{[}t_{s,a}\text{ dominates the span]} \\ c_\phi(s,a) \;&\leq\; \mu c_\psi(s,a) - \nu r(s,a) + h(s) - \gamma\, P(\cdot \mid s,a) \cdot h && \text{[robust constraint and the previous line]} \end{aligned}$$

Every acceptable policy is safe for ϕ. Let $\pi \in F^\varepsilon_\psi(d)$ with true occupancy $x \in \mathcal{X}(P)$. The flow equations of $\mathcal{X}(P)$ give $\sum_{s,a} x(s,a)[h(s) - \gamma P(\cdot \mid s,a) \cdot h] = (1-\gamma)\mu_0 \cdot h$.

$$\begin{aligned} c_\phi \cdot x \;&\leq\; \mu\, c_\psi \cdot x - \nu\, r \cdot x + (1-\gamma)\mu_0 \cdot h && \text{[integrate the pointwise inequality against } x \geq 0\text{]} \\ &\leq\; \mu d - \nu\big(V_\psi(d) - \varepsilon\big) + (1-\gamma)\mu_0 \cdot h && \text{[}c_\psi \cdot x \leq d,\, r \cdot x \geq V_\psi(d) - \varepsilon\text{]} \\ &\leq\; \mu d - \nu\big(\underline{V}_\psi - \varepsilon\big) + (1-\gamma)\mu_0 \cdot h && \text{[}\underline{V}_\psi \leq V_\psi(d)\text{]} \\ &=\; \overline{W}_\phi \;\leq\; d && \text{[certification]} \end{aligned}$$

Hence $\Gamma^\varepsilon_\psi(C, d) \leq 0$ and ψ is ε-sufficient by Theorem 4.1(b). Since $\Pr(E) \geq 1 - \delta_{\mathrm{dec}}$, a certified reading is insufficient with probability at most δ_{dec}. E is one event and each offline ARROW call is a deterministic function of $\widehat{P}$, so soundness holds simultaneously for every clarification round without spending a fresh δ_{dec}. Failure to certify leaves the decision open. □

Proof of Theorem 4.2(b), the information lower bound. A sound procedure must see enough transitions to tell the true dynamics from every decision-reversing model. Let $\mathcal{A}$ be the event that the procedure certifies ψ, let $Q \in \mathcal{Q}_\psi$, and let $P^{(n)}, Q^{(n)}$ be the laws of the log under P and Q. Here kl denotes the binary relative entropy, and the first line is the data-processing inequality in the form of Kaufmann et al. (2016). The second line uses $P^{(n)}(\mathcal{A}) \geq 1 - \delta > \delta \geq Q^{(n)}(\mathcal{A})$, which needs $\delta < 1/2$, and that $\mathrm{kl}(p, q)$ increases in p and decreases in q when $p \geq q$.

$$\begin{aligned} \mathrm{kl}\big(P^{(n)}(\mathcal{A}), Q^{(n)}(\mathcal{A})\big) \;&\leq\; \mathrm{KL}\big(P^{(n)} \,\|\, Q^{(n)}\big) && \text{[data processing]} \\ \mathrm{kl}(1-\delta, \delta) \;&\leq\; \mathrm{kl}\big(P^{(n)}(\mathcal{A}), Q^{(n)}(\mathcal{A})\big) && \text{[}1 - \delta > \delta\text{; kl monotone]} \\ \mathrm{KL}\big(P^{(n)} \,\|\, Q^{(n)}\big) \;&=\; \sum_z \mathbb{E}_P[N_z]\, \mathrm{KL}(P_z \,\|\, Q_z) && \text{[chain rule over transitions]} \\ &=\; n \sum_z w_z\, \mathrm{KL}(P_z \,\|\, Q_z) && \text{[}\mathbb{E}_P[N_z] = n w_z\text{]} \end{aligned}$$

Combining the three lines and taking the infimum over $Q \in \mathcal{Q}_\psi$, with $\inf \emptyset = +\infty$, gives $n\, I^\star_\psi(P) \geq \mathrm{kl}(1-\delta, \delta)$. The chain rule holds for the fixed-row model, where N_z is deterministic, and for a

sampled path, where $\mathbb{E}_P[N_z] = nw_z$ defines w_z. Appendix D.2 reports, for each class, the weighted relative entropy $\bar{I}_\psi = \sum_z w_z \,\mathrm{KL}(P_z \,\|\, \bar{Q}_z)$ of a chain $\bar{Q}$ with $\Gamma^{\varepsilon}_{\psi,\bar{Q}}(C,d) > 0$ found by directional bisection and local refinement. Since $\bar{Q} \in \mathcal{Q}_\psi$, $\bar{I}_\psi \geq I^\star_\psi(P)$, so $\mathrm{kl}(1-\delta,\delta)/\bar{I}_\psi \leq \mathrm{kl}(1-\delta,\delta)/I^\star_\psi(P)$ and part (b) holds with $\bar{I}_\psi$ in place of $I^\star_\psi(P)$, a weaker but valid bound. □

Margins for completeness. The certificate succeeds once the confidence radii are small relative to the slacks of the true-model certificates, defined here. Fix the class Π of stationary randomized policies and a feasible ψ with $C \setminus \{\psi\} \neq \emptyset$ (when $C = \{\psi\}$ sufficiency is immediate). Let $m_{\max} = \max_z m_z$, $\alpha_{\max} = \max_z \alpha_z$, and, with $N_z = nw_z$ transitions observed from row z, $w_{\min} = \min_z w_z > 0$. Suppose every competitor $\phi \neq \psi$ has a certificate $(h_\phi, \mu_\phi, \nu_\phi)$ of Proposition S4 on the true model whose pointwise inequalities have positive slack. Let χ_ψ be the smallest such slack over all (s,a) and all competitors, and H_ψ the largest variation of any h_ϕ over the possible next states of one row. Let $A_\nu = \{\phi \neq \psi : \nu_\phi > 0\}$. When $A_\nu \neq \emptyset$, suppose the objective inequalities of its certificates have positive slack, with smallest value χ_{obj}, let $\nu_{\max} = \max_{\phi \in A_\nu} \nu_\phi$, and suppose some policy π_S has budget slack $s_\psi = d - J_{c_\psi}(\pi_S) > 0$. Let λ_ψ bound the optimal budget multipliers of $V_\psi(b)$, the rates at which the best return grows with the budget, for $b \in [d - s_\psi/2, d]$ ($\lambda_\psi = 2r_{\max}/s_\psi$ always works, with $r_{\max} = \max r$). Write $\bar{B}_f = \gamma\,\mathrm{span}(f)\,\alpha_{\max}/(2(1-\gamma))$ for $f \in \{r, c_\psi\}$.

Theorem S2 (Finite-data completeness of the robust-dual certificate)**.** *With the margins just defined, on E the robust-dual certificate certifies ψ whenever (i) $\gamma\,\alpha_{\max} H_\psi \leq \chi_\psi$ and (ii) when $A_\nu \neq \emptyset$, $2\bar{B}_{c_\psi} \leq s_\psi/2$ and $\nu_{\max}\big(2\bar{B}_r + 2\lambda_\psi \bar{B}_{c_\psi}\big) \leq \chi_{\mathrm{obj}}$. With $N_z = nw_z$ both hold once*

$$n \;\geq\; \frac{2\gamma^2\big(m_{\max}\ln 2 + \ln(|\mathcal{Z}|/\delta_{\mathrm{dec}})\big)}{w_{\min}} \max\left\{\frac{H_\psi}{\chi_\psi}, \frac{\nu_{\max}(\mathrm{span}(r) + \lambda_\psi \mathrm{span}(c_\psi))}{(1-\gamma)\chi_{\mathrm{obj}}}, \frac{2\mathrm{span}(c_\psi)}{(1-\gamma)s_\psi}\right\}^2,$$

the last two entries omitted when $A_\nu = \emptyset$. Hence, with $M_\delta = m_{\max} + \log(|\mathcal{Z}|/\delta_{\mathrm{dec}})$, $n = O(M_\delta \max\{\cdot,\cdot,\cdot\}^2/w_{\min})$, and $n = O(M_\delta H_\psi^2/(w_{\min}\chi_\psi^2))$ when $A_\nu = \emptyset$.

Proof of Theorem S2. Under (i) and (ii), each competitor's true-model certificate stays feasible for the robust program on E. In this proof $(h^\star, \mu^\star, \nu^\star)$ is the certificate $(h_\phi, \mu_\phi, \nu_\phi)$ of one competitor ϕ, $\chi \geq \chi_\psi$ its smallest pointwise slack and $\omega_{s,a} = \mathrm{span}_{\mathrm{supp}(s,a)}(h^\star) \leq H_\psi$ the variation of $h^\star$ over the successors of (s,a). Here $\phi \in A_\nu$ means $\nu^\star > 0$. Every competitor's certificate satisfies $c_\phi(s,a) \leq \mu^\star c_\psi(s,a) - \nu^\star r(s,a) + h^\star(s) - \gamma P(\cdot \mid s,a)\cdot h^\star - \chi$ for every (s,a) and objective $(1-\gamma)\mu_0 \cdot h^\star + \mu^\star d - \nu^\star(V_\psi(d) - \varepsilon)$ at most d, and for $\phi \in A_\nu$ at most $d - \chi_{\mathrm{obj}}$. Take $t_{s,a} = \omega_{s,a}$ and write $\alpha = \alpha_{z(s,a)}$. On E,

$$\begin{aligned}
\widehat{P}(\cdot \mid s,a)\cdot h^\star + \tfrac{\alpha}{2} t_{s,a} \;&\leq\; P(\cdot \mid s,a)\cdot h^\star + \alpha\,\omega_{s,a} && [L_1 \text{ bound of the soundness proof, other direction}]\\
\text{robust slack at } (s,a) \;&\geq\; \chi - \gamma\,\alpha\,\omega_{s,a} \;\geq\; 0 && [\text{condition (i)}]
\end{aligned}$$

so the pointwise inequalities hold for every competitor. If $\phi \notin A_\nu$ then $\nu^\star = 0$, the robust and true objectives coincide, and the witness is feasible. If $\phi \in A_\nu$,

$$\begin{aligned}
\text{robust objective} \;&=\; \text{true objective} + \nu^\star\big(V_\psi(d) - \underline{V}_\psi\big) && [\text{only } \underline{V}_\psi \text{ differs}]\\
&\leq\; d - \chi_{\mathrm{obj}} + \nu^\star\big(V_\psi(d) - \underline{V}_\psi\big) \;\leq\; d && [\text{objective} \leq d - \chi_{\mathrm{obj}};\ \text{(ii), bound in the next display}]
\end{aligned}$$

so the witness is feasible for Equation (5) with $\overline{W}_\phi \leq d$ in both cases, and the same for every competitor certifies ψ. Substituting α_z with $N_z = nw_z$ turns condition (i) into $n \geq 2\gamma^2\big(m_{\max}\ln 2 + \ln(|\mathcal{Z}|/\delta_{\mathrm{dec}})\big)H_\psi^2/(w_{\min}\chi_\psi^2)$, the first entry of the maximum in Theorem S2, and the two parts of condition (ii) into the remaining two entries. Assume now $A_\nu \neq \emptyset$. It remains to bound $V_\psi(d) - \underline{V}_\psi$. Since $B_f(x) \leq \bar{B}_f$ for every occupancy, let π' be optimal for $V_\psi(d - 2\bar{B}_{c_\psi})$ under P, which is feasible because $2\bar{B}_{c_\psi} \leq s_\psi/2 < s_\psi$.

$$\begin{aligned}
c_\psi \cdot \widehat{x}_{\pi'} + B_{c_\psi}(\widehat{x}_{\pi'}) \;&\leq\; J_{c_\psi}(\pi') + 2\bar{B}_{c_\psi} \;\leq\; d && [\text{Lemma S1; choice of } \pi']\\
\underline{V}_\psi \;&\geq\; r\cdot\widehat{x}_{\pi'} - B_r(\widehat{x}_{\pi'}) \;\geq\; V_\psi(d - 2\bar{B}_{c_\psi}) - 2\bar{B}_r && [\widehat{x}_{\pi'} \text{ admissible in (1); Lemma S1}]\\
V_\psi(d) - \underline{V}_\psi \;&\leq\; 2\bar{B}_r + 2\lambda_\psi \bar{B}_{c_\psi} && [\text{concavity of } V_\psi, \text{ see the text after}]\\
&=\; \frac{\gamma\,\alpha_{\max}}{1-\gamma}\big(\mathrm{span}(r) + \lambda_\psi\,\mathrm{span}(c_\psi)\big) && [\text{definition of } \bar{B}_f]
\end{aligned}$$

where the concavity step is $V_\psi(d) - V_\psi(b) \le \lambda_b(d-b)$ at $b = d - 2\bar{B}_{c_\psi}$ for any optimal budget multiplier λ_b at b, a supergradient of the concave V_ψ, with $\lambda_b \le \lambda_\psi$ because $b \in [d - s_\psi/2, d]$. The term is $O(\alpha_{\max}/(1-\gamma))$, and condition (ii) makes the robust objective at most d for every competitor in A_ν. The stated choice $\lambda_\psi = 2r_{\max}/s_\psi$ works because for every $b \in [d - s_\psi/2, d]$ the slack $b - J_{c_\psi}(\pi_S)$ is at least $s_\psi/2$. Strong duality at b, with the Lagrangian (return plus the multiplier times the budget slack) evaluated at π_S, then gives $V_\psi(b) \ge J_r(\pi_S) + \lambda_b s_\psi/2 \ge \lambda_b s_\psi/2$ as $r \ge 0$, while $V_\psi(b) \le r_{\max}$ as returns are normalized, so $\lambda_b \le 2r_{\max}/s_\psi$. □

The generic uniform certificate. The generic uniform certificate is the standard offline baseline, one uniform transition-error radius propagated through every policy value. It is the comparison certificate of Corollary S3(b) and of the uniform column of Table 14, defined here so that the corollary is precise. Consider a finite tabular MDP with S states, A actions, unknown P, known μ_0, $r \in [0, r_{\max}]$ and $c_\phi \in [0, c_{\max}]$, and horizon H or discount γ. The log holds at most $N_{\max}$ transitions from an arbitrary, possibly adaptive behavior policy (the policy that generated the log). Let $\mathcal{K}$ be the covered pairs, those with $N(s,a) \ge n_{\min}$, $\widehat{P}$ the empirical model, $\Pi_\mathcal{K}$ the policies supported on $\mathcal{K}$ (all sets and values of this certificate are relative to $\Pi_\mathcal{K}$), $\widehat{J}_f$ values under $\widehat{P}$, and

$$\beta_n = L_H \sqrt{2\big(S \ln 2 + \ln(SAN_{\max}/\delta_{\mathrm{dec}})\big)/n_{\min}}, \qquad \beta_c = c_{\max}\beta_n, \quad \beta_r = r_{\max}\beta_n,$$

with $L_H = H(H-1)/2$ in the finite-horizon case and $L_H = \gamma/(2(1-\gamma))$, with $\mathrm{span}(f)$ in place of $f_{\max}$, in the discounted case. Given ψ, d, $\varepsilon \ge 0$ and δ_{dec}, step (1) sets $\widehat{V}^- \leftarrow \widehat{V}_\psi(d - \beta_c) - \beta_r$, where $\widehat{V}_\psi(b) = \sup\{\widehat{J}_r(\pi) : \pi \in \Pi_\mathcal{K},\ \widehat{J}_{c_\psi}(\pi) \le b\}$, returning NOT CERTIFIED if infeasible. Step (2) forms $\widehat{G} \leftarrow \{\pi \in \Pi_\mathcal{K} :\ \widehat{J}_{c_\psi}(\pi) \le d + \beta_c,\ \widehat{J}_r(\pi) \ge \widehat{V}^- - \beta_r - \varepsilon\}$. Step (3) computes $\widehat{W} \leftarrow \max_{\phi \ne \psi} \sup_{\pi \in \widehat{G}} \widehat{J}_{c_\phi}(\pi) + \beta_c$, one LP per competitor. Step (4) certifies ψ iff $\widehat{W} \le d$.

Uniform model error. With probability at least $1 - \delta_{\mathrm{dec}}$, simultaneously for every $\pi \in \Pi_\mathcal{K}$ and every per-step f with $0 \le f \le f_{\max}$, $|J_f(\pi) - \widehat{J}_f(\pi)| \le f_{\max}\beta_n$. Generate the data from sequences drawn in advance, for every pair (s,a) an infinite i.i.d. sequence $X_1^{s,a}, X_2^{s,a}, \cdots \sim P(\cdot \mid s,a)$, independent across pairs and of the behavior policy's randomness, the j-th transition recorded from (s,a) receiving $X_j^{s,a}$. Conditionally on the history, the entry consumed at each step is a fresh draw from the true row, so this has exactly the joint law of the data, and $\widehat{P}(\cdot \mid s,a)$ is the empirical distribution of the first $N(s,a) \le N_{\max}$ entries of a fixed sequence, with $N(s,a)$ random. For the first n entries, Weissman et al. (2003) give $\Pr[\|\widehat{P}^{(n)}(\cdot \mid s,a) - P(\cdot \mid s,a)\|_1 \ge t_n] \le (2^S - 2)e^{-nt_n^2/2} \le \delta_{\mathrm{dec}}/(SAN_{\max})$ for $t_n = \sqrt{2(S\ln 2 + \ln(SAN_{\max}/\delta_{\mathrm{dec}}))/n}$, and a union over the SA pairs and the counts $n \le N_{\max}$ gives, with probability at least $1 - \delta_{\mathrm{dec}}$, $\|\widehat{P}(\cdot \mid s,a) - P(\cdot \mid s,a)\|_1 \le t_{N(s,a)} \le t_{n_{\min}}$ on every covered pair, since t_n decreases in n and $N(s,a) \ge n_{\min}$ on $\mathcal{K}$, whichever pairs $\mathcal{K}$ turns out to contain. Adaptive collection costs only the $\ln N_{\max}$ inside the root, and the bound does not depend on $|U_0|$. On that event the value difference telescopes over the horizon, $J_f(\pi) - \widehat{J}_f(\pi) = \sum_{h=0}^{H-2} \mathbb{E}_\pi^{\widehat{P}}[(P - \widehat{P})(\cdot \mid s_h, a_h) \cdot V_{h+1}]$ with V_h the value-to-go under P from step h ($V_H \equiv 0$) and $0 \le V_{h+1} \le (H - h - 1)f_{\max}$, so by Hölder $|J_f(\pi) - \widehat{J}_f(\pi)| \le t_{n_{\min}} f_{\max} \sum_{h=0}^{H-2}(H-h-1) = t_{n_{\min}} f_{\max} H(H-1)/2$. In the discounted case $J_f(\pi) = (1-\gamma)\mu_0 \cdot V_P^\pi$ with $V_P^\pi, V_{\widehat{P}}^\pi \in [f_{\min}, f_{\max}]/(1-\gamma)$ and $V_P^\pi - V_{\widehat{P}}^\pi = \gamma(I - \gamma P_\pi)^{-1}(P_\pi - \widehat{P}_\pi)V_{\widehat{P}}^\pi$. Each row of $P_\pi - \widehat{P}_\pi$ sums to zero, so when every row has L_1 error at most t, $\|(P_\pi - \widehat{P}_\pi)V\|_\infty \le t\,\mathrm{span}(V)/2$, and $\|(I - \gamma P_\pi)^{-1}\|_\infty \le 1/(1-\gamma)$, which gives $|J_f(\pi) - \widehat{J}_f(\pi)| \le \frac{\gamma}{2(1-\gamma)}\, t\,\mathrm{span}(f)$, the constant $L_H = \gamma/(2(1-\gamma))$ with $\mathrm{span}(f)$ in place of $f_{\max}$.

The regularity condition. Regularity asks that the best return and the worst competing cost change at a bounded rate near the operating point. Call (ψ, d, ε) *regular* with constants $(\bar{b}, \lambda, \Lambda)$, $\bar{b} > 0$, if V_ψ is finite and λ-Lipschitz on $[d - \bar{b}, d]$ and the worst competing cost $\Phi(d', t) = \max_{\phi \ne \psi} \sup\{J_{c_\phi}(\pi) : J_{c_\psi}(\pi) \le d',\ J_r(\pi) \ge t\}$ is finite and Λ-Lipschitz on $[d, d + \bar{b}] \times [V_\psi(d) - \varepsilon - \bar{b}, V_\psi(d) - \varepsilon]$. In the LP case this holds whenever some policy has $J_{c_\psi} < d$ and $\varepsilon > 0$, with λ and Λ bounds on the optimal dual multipliers of the budget and of the two side constraints of the face programs. Put $\mathrm{c_{inst}} = c_{\max}(2 + 2\Lambda(1+\lambda)) + 4\Lambda r_{\max}$ and

$\rho(\kappa) = \min\{\kappa/\mathrm{c_{inst}},\ \bar{b}/(2c_{\max}),\ \bar{b}/(4r_{\max} + 2\lambda c_{\max})\}/L_H$, with L_H the simulation constant of the generic uniform certificate.

The regularity condition holds in the LP case. Here a strictly feasible policy bounds the dual multipliers, which makes both value functions Lipschitz. Write $\mathrm{val}(b)$ for the optimal value of a linear program as a function of the right-hand side b of its side constraints and $D(b)$ for its optimal dual set. By strong duality $\mathrm{val}(b) = \min_{y\ge 0}\{b^\top y + \mathrm{const}(y)\}$ is concave and finite in b, its superdifferential is $D(b)$, and $|\mathrm{val}(b_1) - \mathrm{val}(b_2)| \le (\sup_{b\in B}\sup_{y\in D(b)} \|y\|)\,\|b_1 - b_2\|$ on any window B on which the dual sets are bounded. Slater's condition, the existence of a strictly feasible point, bounds them wherever the right-hand sides are strictly feasible. Let π_S satisfy $J_{c_\psi}(\pi_S) < d$ and, with $\varepsilon > 0$, let $\pi^\star$ be $\frac{\varepsilon}{2}$-optimal in $\Pi_\psi(d)$. For $\bar{b} < d - J_{c_\psi}(\pi_S)$, π_S is strictly feasible for every $b \in [d - \bar{b}, d]$, which bounds the budget multiplier of V_ψ. For the face programs, mix in occupancy space, $\pi_\alpha = (1-\alpha)\pi^\star + \alpha\pi_S$. Costs are affine, so $J_{c_\psi}(\pi_\alpha) < d \le d'$ for $\alpha > 0$, and $J_r(\pi_\alpha) \ge (1-\alpha)(V_\psi(d) - \varepsilon/2) > V_\psi(d) - \varepsilon \ge t$ for $\alpha < \varepsilon/(2V_\psi(d))$ (any α if $V_\psi(d) = 0$), since $r \ge 0$. So π_α is strictly feasible for every (d', t) in the box of the regularity condition once $\bar{b}$ is small enough, and Slater's condition bounds the dual optimal sets there, with Λ a bound on the ℓ_1 norm of the optimal multipliers of the two side constraints and λ the bound on the budget multiplier.

Corollary S3 (Worst-case margin law)**.** *Suppose $\Gamma^{\varepsilon}_{\psi,P}(C,d) \le -\kappa$ with $\kappa > 0$ and (ψ, d, ε) is regular. Then (a) every $Q \in \mathcal{Q}_\psi$ satisfies $\max_z \|P_z - Q_z\|_1 \ge \rho(\kappa)$, so for $w_{\min} > 0$, $I^\star_\psi(P) \ge w_{\min}\rho(\kappa)^2/2 = \Omega(\kappa^2)$; (b) the generic uniform certificate, run on the policies supported on the covered pairs, those observed at least $n_{\min}$ times, is sound on its confidence event and, when the margin and regularity hypotheses hold for that class, certifies once $n_{\min} = O\big((|S| + \log(|S||A|N_{\max}/\delta_{\mathrm{dec}}))/\kappa^2\big)$ transitions are observed from every covered pair; (c) on some instances every procedure needs $\Omega(\log(1/\delta_{\mathrm{dec}})/\kappa^2)$ transitions, so the $1/\kappa^2$ dependence is worst-case tight.*

Proof of Corollary S3, soundness. Every step of the certificate is pessimistic on the uniform-model-error event, on which we work, $|J_f(\pi) - \widehat{J}_f(\pi)| \le \beta_f$, with β_r for r and β_c for every cost, for every $\pi \in \Pi_{\mathcal{K}}$ and every $f \in \{r\} \cup \{c_\phi\}_{\phi\in U_0}$.

The reward reference is not optimistic.

$$
\begin{aligned}
\widehat{J}_{c_\psi}(\pi') \le d - \beta_c \ &\Rightarrow\ J_{c_\psi}(\pi') \le d && \text{[uniform error for } c_\psi\text{]}\\
&\Rightarrow\ J_r(\pi') \le V_\psi(d) && \text{[}\pi'\text{ is feasible for } V_\psi(d)\text{]}\\
&\Rightarrow\ \widehat{J}_r(\pi') \le V_\psi(d) + \beta_r && \text{[uniform error for } r\text{]}\\
\widehat{V}^- = \widehat{V}_\psi(d - \beta_c) - \beta_r \ &\le\ V_\psi(d) && \text{[supremum over such } \pi'\text{]}
\end{aligned}
$$

The true face lies inside the surrogate face. For $\pi \in F^{\varepsilon}_\psi(d)$,

$$
\begin{aligned}
\widehat{J}_{c_\psi}(\pi) \ &\le\ J_{c_\psi}(\pi) + \beta_c \ \le\ d + \beta_c && \text{[}\pi\text{ is feasible under } \psi\text{]}\\
\widehat{J}_r(\pi) \ &\ge\ J_r(\pi) - \beta_r \ \ge\ V_\psi(d) - \varepsilon - \beta_r && \text{[}\pi\text{ is } \varepsilon\text{-optimal under } \psi\text{]}\\
&\ge\ \widehat{V}^- - \varepsilon - \beta_r && \text{[previous display]}\\
F^{\varepsilon}_\psi(d) \ &\subseteq\ \widehat{G} && \text{[the two defining inequalities of } \widehat{G}\text{]}
\end{aligned}
$$

The reported worst competing cost is not optimistic. For every $\phi \ne \psi$,

$$
\begin{aligned}
\sup_{\widehat{G}} \widehat{J}_{c_\phi} \ &\ge\ \sup_{F^{\varepsilon}_\psi(d)} \widehat{J}_{c_\phi} && \text{[supremum over a larger set]}\\
&\ge\ \sup_{F^{\varepsilon}_\psi(d)} J_{c_\phi} - \beta_c && \text{[uniform error for } c_\phi\text{]}\\
\widehat{W} \ &\ge\ W^{\varepsilon}_\psi(U_0; d) && \text{[maximize over } \phi\text{, restore the } +\beta_c\text{]}\\
\widehat{W} \le d \ &\Rightarrow\ \Gamma^{\varepsilon}_\psi(U_0, d) \le 0 && \text{[definition of the margin]}\\
&\Rightarrow\ F^{\varepsilon}_\psi(d) \subseteq \Pi_{U_0}(d) && \text{[Theorem 4.1(b)]}
\end{aligned}
$$

The bad event is contained in the complement of the uniform-error event, of probability at most δ_{dec}. The event is uniform over $\Pi_{\mathcal{K}}$ and over every per-step function, so, as for the robust-dual certificate, soundness holds simultaneously for every clarification round. □

Proof of Corollary S3, completeness. A small enough radius β_n cannot overturn a margin of $-\kappa$. Suppose $W^{\varepsilon}_{\psi}(U_0; d) \le d - \kappa$ and $\beta_n \le \min\{\kappa/\mathrm{c_{inst}},\ \bar{b}/(2c_{\max}),\ \bar{b}/(4r_{\max} + 2\lambda c_{\max})\}$. Since $\beta_n \propto \sqrt{(S\ln 2 + \ln(SAN_{\max}/\delta_{\mathrm{dec}}))/n_{\min}}$, solving $\beta_n \le \kappa/\mathrm{c_{inst}}$ for $n_{\min}$ is the stated rate. Work on the uniform-error event.

The first program is feasible and its value is not too pessimistic.

$$
\begin{aligned}
2\beta_c \le \bar{b} \;&\Rightarrow\; V_\psi(d - 2\beta_c) \text{ finite} && \text{[regularity: } V_\psi \text{ finite on } [d-\bar{b}, d]\text{]}\\
&\Rightarrow\; \exists\, \pi' \in \Pi_{\mathcal{K}} :\; J_{c_\psi}(\pi') \le d - 2\beta_c && \text{[a witness of that finite value]}\\
\widehat{J}_{c_\psi}(\pi') \;&\le\; J_{c_\psi}(\pi') + \beta_c \;\le\; d - \beta_c && \text{[uniform error: the first program is feasible]}\\
\widehat{V}_\psi(d - \beta_c) \;&\ge\; \widehat{J}_r(\pi') \;\ge\; J_r(\pi') - \beta_r && \text{[}\pi' \text{ admissible in step (1); uniform error]}\\
\widehat{V}^{-} \;&\ge\; V_\psi(d - 2\beta_c) - 2\beta_r && \text{[supremum over such } \pi', \text{ minus } \beta_r\text{]}
\end{aligned}
$$

Every surrogate-face policy is nearly on the true face. For $\pi \in \widehat{G}$,

$$
\begin{aligned}
J_{c_\psi}(\pi) \;&\le\; \widehat{J}_{c_\psi}(\pi) + \beta_c \;\le\; d + 2\beta_c && \text{[definition of } \widehat{G}\text{]}\\
J_r(\pi) \;&\ge\; \widehat{J}_r(\pi) - \beta_r \;\ge\; \widehat{V}^{-} - \varepsilon - 2\beta_r && \text{[definition of } \widehat{G}\text{]}\\
&\ge\; V_\psi(d - 2\beta_c) - \varepsilon - 4\beta_r && \text{[previous display]}\\
&\ge\; V_\psi(d) - \varepsilon - 4\beta_r - 2\lambda\beta_c && \text{[}V_\psi \text{ is } \lambda\text{-Lipschitz on } [d-\bar{b}, d]\text{]}
\end{aligned}
$$

Conclusion. Φ is nondecreasing in d' and nonincreasing in t, and $2\beta_c \le \bar{b}$ and $4\beta_r + 2\lambda\beta_c \le \bar{b}$ keep the arguments inside the box.

$$
\begin{aligned}
\widehat{W} - \beta_c \;&\le\; \max_{\phi \ne \psi}\ \sup_{\widehat{G}} J_{c_\phi} + \beta_c && \text{[uniform error for each } c_\phi\text{]}\\
&\le\; \Phi\big(d + 2\beta_c,\ V_\psi(d) - \varepsilon - 4\beta_r - 2\lambda\beta_c\big) + \beta_c && \text{[previous display and monotonicity of } \Phi\text{]}\\
\widehat{W} \;&\le\; \Phi\big(d,\ V_\psi(d) - \varepsilon\big) + \Lambda\big(2\beta_c + 4\beta_r + 2\lambda\beta_c\big) + 2\beta_c && \text{[}\Lambda\text{-Lipschitz]}\\
&\le\; (d - \kappa) + \mathrm{c_{inst}}\beta_n && \text{[}W^{\varepsilon}_{\psi} \le d - \kappa\text{; collect constants]}\\
&\le\; d && \text{[}\beta_n \le \kappa/\mathrm{c_{inst}}\text{]}
\end{aligned}
$$

so step (4) certifies ψ. □

Proof of Corollary S3, distance and information. Every decision-reversing model differs from the true one by at least $\rho(\kappa)$ in some row. Take every state–action pair covered, $\mathcal{K} = S \times A$, so $\Pi_{\mathcal{K}} = \Pi$ and the certificate's soundness and completeness concern the class defining $\mathcal{Q}_\psi$. Suppose some $Q \in \mathcal{Q}_\psi$ had $\max_z \|P_z - Q_z\|_1 = t < \rho(\kappa)$. Then $L_H t < \min\{\kappa/\mathrm{c_{inst}},\ \bar{b}/(2c_{\max}),\ \bar{b}/(4r_{\max} + 2\lambda c_{\max})\}$ by the definition of $\rho(\kappa)$, so the certificate run on P itself with uniform radius t, that is with $\beta_n = L_H t$, certifies ψ by completeness, its true margin being at most $-\kappa$. Soundness with Q as the true model, which lies within radius t of the input, then gives $\Gamma^{\varepsilon}_{\psi,Q}(C, d) \le 0$, contradicting $Q \in \mathcal{Q}_\psi$. Hence every $Q \in \mathcal{Q}_\psi$ has a row at L_1 distance at least $\rho(\kappa)$, and

$$
\sum_z w_z\, \mathrm{KL}(P_z \| Q_z) \;\ge\; w_{\min} \max_z \tfrac{1}{2}\|P_z - Q_z\|_1^2 \;\ge\; w_{\min}\rho(\kappa)^2/2 \qquad \text{[Pinsker's inequality; infimum over } Q\text{]}
$$

which is the stated bound on $I^{\star}_{\psi}(P)$. □

Proof of Corollary S3, tightness. A two-step example shows that no procedure can beat the $1/\kappa^2$ rate. Its states are $s_0, s_{\mathrm{good}}, s_{\mathrm{bad}}$, with actions a_1, a_2 at s_0 and horizon 2. Action a_1 earns reward 1 and moves to s_{bad} with probability p, else to s_{good}. Action a_2 earns $1/2$ and moves to s_{good}. The

costs are $c_\psi \equiv 0$ and $c_\phi = 1$ at s_{bad}. Write $\theta = \pi(a_1 \mid s_0)$.

$$
\begin{aligned}
J_r(\theta) &= \theta + \tfrac{1}{2}(1-\theta) = \tfrac{1+\theta}{2}, \qquad J_{c_\phi}(\theta) = p\,\theta && \text{[expectation over the two actions at } s_0\text{]} \\
V_\psi(d) &= \max_\theta J_r(\theta) = 1 && \text{[} c_\psi \equiv 0\text{, so the budget never binds]} \\
F^\varepsilon_\psi(d) &= \{\theta : \tfrac{1+\theta}{2} \geq 1-\varepsilon\} = \{\theta \geq 1-2\varepsilon\} \ni 1 && \text{[definition of the } \varepsilon\text{-face, } \varepsilon < 1/2\text{]} \\
\Gamma^\varepsilon_\psi(d) &= \max_{\theta \in F^\varepsilon_\psi(d)} J_{c_\phi}(\theta) - d = p - d && \text{[} J_{c_\phi} \text{ increasing, } \theta = 1 \text{ attainable]}
\end{aligned}
$$

Take $p = d - \kappa$ and $p = d + \kappa$ with $\kappa < \min\{d, 1-d\}$. The correct verdict then flips, while the two worlds differ only in the $\mathrm{Ber}(p)$ law of the (s_0, a_1) transition.

$$
\begin{aligned}
\mathrm{KL}\big(\mathrm{Ber}(d-\kappa) \| \mathrm{Ber}(d+\kappa)\big) &\leq \frac{(2\kappa)^2}{(d+\kappa)(1-d-\kappa)} && \text{[KL} \leq \chi^2 \text{ for Bernoulli, gap } 2\kappa\text{]} \\
P^{(n)}(\mathcal{A}^c) + Q^{(n)}(\mathcal{A}) &\geq \tfrac{1}{2}\, e^{-n\,\mathrm{KL}} && \text{[Bretagnolle–Huber inequality]} \\
n &\geq \frac{(d+\kappa)(1-d-\kappa)}{4\kappa^2} \ln \frac{1}{4\delta_{\mathrm{dec}}} && \text{[both errors at most } \delta_{\mathrm{dec}} < 1/4\text{]}
\end{aligned}
$$

for any procedure with error at most $\delta_{\mathrm{dec}} < 1/4$ in both worlds. Since $I^\star_\psi = \Theta(\kappa^2)$ on this instance, Theorem 4.2(b) recovers the same rate. In the constants, κ is the cost slack of the ε-face, λ the shadow price of the budget and Λ the growth rate of the worst competing cost as the face widens, which diverges as $\varepsilon \to 0$ near a tie. □

C.1.3 Theorem 4.3: safe stopping under clarification

Clarification can stop once some reading is sufficient for the survivors U_ξ, because truthful answers never remove the governing reading $\psi^\dagger$, the unknown reading that applies to the rule. The SAFE stop needs any such reading and gives safety under $\psi^\dagger$. The FREE stop also needs it to survive, and then loses no best return relative to protecting U_ξ.

Proof of Theorem 4.3. Any reading, the SAFE stop. Assume $F^\varepsilon_\psi(d) \subseteq \Pi_{U_\xi}(d)$ for some $\psi \in U_0$, surviving or not, and let $\pi \in F^\varepsilon_\psi(d)$.

$$
\begin{aligned}
\psi^\dagger &\in U_\xi && \text{[answers consistent with } \psi^\dagger\text{]} \\
J_{c_{\psi^\dagger}}(\pi) &\leq d && \text{[} \pi \in \Pi_{U_\xi}(d) \text{ by hypothesis; previous line]} \\
J_r(\pi) &\geq V_\psi(d) - \varepsilon && \text{[} \pi \text{ lies in the } \varepsilon\text{-optimal face]}
\end{aligned}
$$

Nothing here used whether ψ survived, nor $|U_\xi|$.

A surviving reading, the FREE stop. Assume in addition $\psi \in U_\xi$. Approximate $V_\psi(d)$ within $\rho \leq \varepsilon$. The approximating policy then lies in the face, the first part makes it feasible for the surviving set, and letting $\rho \downarrow 0$ gives the value equality.

$$
\begin{aligned}
\Pi_{U_\xi}(d) &\subseteq \Pi_\psi(d) && \text{[} \psi \in U_\xi\text{]} \\
V_{U_\xi}(d) &\leq V_\psi(d) && \text{[maximize over a smaller set]} \\
\rho &\in (0, \varepsilon], \text{ or } \rho = 0 \text{ when } \varepsilon = 0 && \text{[attainment in the second case]} \\
\exists\, \pi_\rho \in \Pi_\psi(d) : \; J_r(\pi_\rho) &\geq V_\psi(d) - \rho && \text{[supremum, or the maximizer when } \rho = 0\text{]} \\
\pi_\rho &\in F^\varepsilon_\psi(d) \subseteq \Pi_{U_\xi}(d) && \text{[} \rho \leq \varepsilon\text{, then the first part]} \\
V_{U_\xi}(d) &\geq V_\psi(d) - \rho && \text{[it competes in the } U_\xi \text{ program]} \\
V_\psi(d) &= V_{U_\xi}(d) && \text{[} \rho \downarrow 0\text{, with the second line]} \\
J_r(\pi) &\geq V_{U_\xi}(d) - \varepsilon && \text{[substitute into the first part]}
\end{aligned}
$$

Without $\psi \in U_\xi$ the first containment can fail and $V_\psi(d) < V_{U_\xi}(d)$ is possible. This is why Refine, the question-selection step of Algorithm 1, distinguishes SAFE from FREE. □

C.1.4 THEOREM 4.4: DEPLOYMENT SAFETY

Retention guarantee. SCORE-KEEP, the plausibility filter of Section 3.2, keeps a generated governing reading with probability at least $1 - \delta_{\text{sem}}$. It scores the generated pool Ψ_ℓ of rule ℓ with a plausibility score $\sigma(\ell, \psi)$ fixed before the deployment rule is seen and retains $U_0 = \{\psi \in \Psi_\ell : \sigma(\ell, \psi) \geq \hat{\tau}_{\text{sem}}\}$, where $\hat{\tau}_{\text{sem}}$ is the m-th smallest score of the governing readings of n_{ref} held-out reference rules, $m = \lfloor \delta_{\text{sem}}(n_{\text{ref}} + 1) \rfloor$ (Vovk et al., 2005; Angelopoulos & Bates, 2023). With no reference rules, or when $m = 0$, it keeps everything. Suppose the reference and deployment rules are exchangeable conditionally on $\psi^\dagger \in \Psi_\ell$ (their joint law is invariant to reordering), and write $z_i = \sigma(\ell_i, \psi_i^\dagger)$ for the reference rules and $z_{n_{\text{ref}}+1} = \sigma(\ell, \psi^\dagger)$ for the deployment rule, with $\text{rank}(z_j) = 1 + \#\{i \neq j : z_i \leq z_j\}$, so that ties count.

$$
\begin{aligned}
z_1, \ldots, z_{n_{\text{ref}}+1} &\text{ are exchangeable} && [\sigma \text{ is fixed before the draw; the rules are exchangeable}] \\
\{\psi^\dagger \notin U_0\} &= \{z_{n_{\text{ref}}+1} < z_{(m)}\} && [\text{definition of } U_0;\ \hat{\tau}_{\text{sem}} = z_{(m)}] \\
&\subseteq \{\text{rank}(z_{n_{\text{ref}}+1}) \leq m\} && [\text{at most } m-1 \text{ other scores are} \leq z_{n_{\text{ref}}+1}] \\
\Pr[\text{rank}(z_{n_{\text{ref}}+1}) \leq m] &\leq m/(n_{\text{ref}} + 1) && [\text{exchangeable; at most } m \text{ indices have rank} \leq m] \\
&\leq \delta_{\text{sem}} && [m = \lfloor \delta_{\text{sem}}(n_{\text{ref}} + 1) \rfloor]
\end{aligned}
$$

so $\Pr[\psi^\dagger \in U_0 \mid \psi^\dagger \in \Psi_\ell] \geq 1 - \delta_{\text{sem}}$. Ties at the threshold are retained and only enlarge U_0. The guarantee comes from the threshold rule, not from the score. A nontrivial threshold needs $n_{\text{ref}} \geq 1/\delta_{\text{sem}} - 1$, and if no candidate clears $\hat{\tau}_{\text{sem}}$ then $U_0 = \emptyset$ and Algorithm 1 abstains.

Proof of Theorem 4.4(a). An unsafe checked deployment needs a generation miss, a retention drop or a failed bound of CHECK (the fresh-episode cost bound of Section 3.3), so the three error levels add. Let U_Ξ be the survivors at the stopping round, fixed with $\hat{\pi}$ before the evaluation episodes are drawn. Let $G = \{\psi^\dagger \in U_0\}$ be retention of the governing reading, $E_{\text{ev}} = \{\max_{\psi \in U_\Xi} J_{c_\psi}(\hat{\pi}) \leq \mathcal{B}(\hat{\pi}, U_\Xi)\}$ validity of CHECK's bound, and $\text{DEPLOY} = \{\mathcal{B}(\hat{\pi}, U_\Xi) \leq d\}$ the deployment event, and write $\mathcal{U} = \{\text{DEPLOY} \wedge J_{c_{\psi^\dagger}}(\hat{\pi}) > d\}$ for an unsafe deployment. Let $\mathcal{F}_{\text{pre}}$ be the σ-field generated by everything determined before the evaluation episodes are drawn, among them Ψ_ℓ, U_0, the clarification history with U_Ξ, the log, the learner's randomness, $\hat{\pi}$ and $\psi^\dagger$. The hypothesis on the evaluator is $\Pr[E_{\text{ev}}^c \mid \mathcal{F}_{\text{pre}}] \leq \delta_{\text{ev}}$ almost surely.

$$
\begin{aligned}
\Pr[G^c \mid \psi^\dagger \in \Psi_\ell] &\leq \delta_{\text{sem}} && [\text{retention guarantee}] \\
\psi^\dagger &\in U_\Xi \text{ on } G && [\text{truthful answers never remove } \psi^\dagger] \\
J_{c_{\psi^\dagger}}(\hat{\pi}) &\leq \max_{\psi \in U_\Xi} J_{c_\psi}(\hat{\pi}) \leq \mathcal{B}(\hat{\pi}, U_\Xi) \leq d && [\text{on } G \cap E_{\text{ev}} \cap \text{DEPLOY, by the previous line}] \\
G \cap \mathcal{U} &\subseteq G \cap E_{\text{ev}}^c && [\text{contrapositive of the previous line}] \\
\Pr[G \cap E_{\text{ev}}^c \mid \psi^\dagger \in \Psi_\ell] &= \mathbb{E}\big[\mathbb{I}_G \Pr[E_{\text{ev}}^c \mid \mathcal{F}_{\text{pre}}] \,\big|\, \psi^\dagger \in \Psi_\ell\big] && [\mathbb{I}_G \text{ is } \mathcal{F}_{\text{pre}}\text{-measurable; tower property}] \\
&\leq \delta_{\text{ev}} && [\text{evaluator hypothesis}] \\
\Pr[\mathcal{U} \mid \psi^\dagger \in \Psi_\ell] &\leq \delta_{\text{sem}} + \delta_{\text{ev}} && [\text{split on } G\text{; the two previous bounds}] \\
\mathcal{U} &\subseteq \{\psi^\dagger \notin \Psi_\ell\} \cup \big(\{\psi^\dagger \in \Psi_\ell\} \cap \mathcal{U}\big) && [\text{split on generation}] \\
\Pr[\mathcal{U}] &\leq \delta_{\text{gen}} + \delta_{\text{sem}} + \delta_{\text{ev}} && [\Pr[\psi^\dagger \notin \Psi_\ell] \leq \delta_{\text{gen}}\text{; previous two lines}]
\end{aligned}
$$

The fifth line conditions on $\mathcal{F}_{\text{pre}}$ instead of multiplying the two probabilities, because G and E_{ev} are not independent. Both U_0 and $\hat{\pi}$ depend on the reference rules. The seventh line is the conditional bound of the theorem and the last line the unconditional one. □

The bound charges each stage its own error. It contains no δ_{dec}, which controls the representation decision, while δ_{ev} covers the validation of the realized policy. Nothing in the theorem constrains how $\hat{\pi}$ was produced, which is why learned and exactly solved policies sit in the same pipeline, and any evaluator meeting the hypothesis may be used (Appendix B.3 describes the one used in the experiments). The generation miss splits as $\Pr[\psi^\dagger \notin \Psi_\ell] = \Pr[A_\emptyset] + \Pr[N] \leq \delta_{\text{gen}}$ with $A_\emptyset = \{\Psi_\ell = \emptyset\}$ and $N = \{\Psi_\ell \neq \emptyset,\ \psi^\dagger \notin \Psi_\ell\}$. On $A_\emptyset$ the pipeline abstains, so $\Pr[\text{DEPLOY} \wedge J_{c_{\psi^\dagger}}(\hat{\pi}) > d] \leq \Pr[N] + \delta_{\text{sem}} + \delta_{\text{ev}}$ and the theorem's δ_{gen} adds $\Pr[A_\emptyset]$ as slack. On N every stage reasons inside the represented survivors, and truthful answers cannot restore a reading that was never a candidate. δ_{gen} is therefore a property of the generation stage on the deployment population, used conservatively by the theorem. Its estimation on annotated requirements is in Appendix D.4.

Redundancy fact. Dropping a reading that some retained reading dominates changes neither the feasible policies nor their best return. Let Ψ be a finite set of readings and $U' \subseteq \Psi$. If for every $\psi \in \Psi$ some $\phi \in U'$ has $J_{c_\psi}(\pi) \le J_{c_\phi}(\pi)$ for every $\pi \in \Pi$, then $\Pi_{U'}(d) = \Pi_\Psi(d)$ and $V_{U'}(d) = V_\Psi(d)$ for every d. Indeed, $U' \subseteq \Psi$ gives $\Pi_\Psi(d) \subseteq \Pi_{U'}(d)$. Conversely, for $\pi \in \Pi_{U'}(d)$ and $\psi \in \Psi$ the dominating ϕ gives $J_{c_\psi}(\pi) \le J_{c_\phi}(\pi) \le d$, so $\pi \in \Pi_\Psi(d)$, and maximizing the same reward over equal sets gives equal values. SAFE-KEEP removes a reading only when a retained reading dominates it in this sense, so it never changes the constrained problem.

Proof of Theorem 4.4(b). On the certified-offline route an unsafe deployment needs the planner's fixed-data guarantee to fail. Let A denote admission into the supported threshold-duration family (rules the parser reads as a load level held for a duration) and $\mathcal{E}_{\text{off}}$ the event on which the guarantee of CERTIFIED-PROTECT, the certified planner of Section 3.3, is valid, $\Pr[\mathcal{E}_{\text{off}}] \ge 1 - \delta_{\text{off}}$, and write $\mathcal{U} = \{\text{DEPLOY} \wedge J_{c_{\psi^\dagger_{\text{parsed}}}}(\hat\pi) > d\}$. On $A = 1$,

$$
\begin{aligned}
\psi^\dagger_{\text{parsed}} &\in \Psi && \text{[deterministic generator on an admitted rule]}\\
\Pi_{U_0}(d) &= \Pi_\Psi(d) && \text{[redundancy fact of this appendix]}\\
\hat\pi &\in \Pi_C(d) \text{ on } \mathcal{E}_{\text{off}} && \text{[the planner's guarantee for the set } C \text{ it is handed]}\\
C = U_0 &\Rightarrow \hat\pi \in \Pi_{U_0}(d) && \text{[previous line with } C = U_0\text{]}\\
C = \{\psi^\star\} &\Rightarrow \hat\pi \in \Pi_{U_0}(d) && \text{[kept only if certified for all of } U_0\text{, Algorithm 1]}\\
\hat\pi &\in \Pi_{U_0}(d) = \Pi_\Psi(d) \subseteq \Pi_{\psi^\dagger_{\text{parsed}}}(d) && \text{[either case; } \psi^\dagger_{\text{parsed}} \in \Psi\text{]}\\
\mathcal{U} &\subseteq \mathcal{E}^c_{\text{off}} && \text{[contrapositive of the previous line]}\\
\Pr[\mathcal{U}] &\le \delta_{\text{off}} && \text{[}\Pr[\mathcal{E}^c_{\text{off}}] \le \delta_{\text{off}}\text{]}
\end{aligned}
$$

On $A = 0$ the certified-offline route abstains, so rejected inputs reduce automation coverage but do not enter the unsafe-deployment event. Admission, deterministic generation and SAFE-KEEP add no probabilistic failure event. The certification error is covered by δ_{off}, and δ_{dec} enters neither deployment bound because a singleton is deployed on this route only when certified on $\mathcal{E}_{\text{off}}$. □

C.2 SUPPORTING THEORY USED BY THE PIPELINE

This subsection proves the supporting results the pipeline uses beyond the four main theorems. They are the dual certificate of fixed-tolerance sufficiency in finite constrained Markov decision processes (CMDPs, Proposition S4) and the composition of ARROW's decision with an approximate learner (Proposition S5). They also cover the dominance relation behind SAFE-KEEP and the single-cost surrogate handed to learners that accept one cost.

C.2.1 THE DUAL CERTIFICATE IN FINITE CMDPS

For stationary randomized policies, the fixed-tolerance test is a linear program whose dual certificate can be checked one state–action pair at a time. Theorem 4.1 is stated over the declared policy class Π. When Π is the stationary randomized class of a finite discounted CMDP, policies are represented by the normalized occupancy polytope $\mathcal{X} = \mathcal{X}(P)$ of Appendix B.3, returns and costs are linear in $x \in \mathcal{X}$, and the fixed-tolerance test becomes a linear program. Write $r, c_\psi, c_\phi \in \mathbb{R}^{S\times A}$ for the reward and cost vectors, so that $J_r(x) = r^\top x$ and $J_{c_\phi}(x) = c_\phi^\top x$.

The fixed-tolerance test reduces to one face program per competitor. For $\phi \in C \setminus \{\psi\}$ and $\varepsilon \ge 0$ let $W_\phi(\varepsilon) = \max\{c_\phi^\top x :\ x \in \mathcal{X},\ c_\psi^\top x \le d,\ r^\top x \ge V_\psi(d) - \varepsilon\}$ be the face program of Equation (2), the largest cost of ϕ over the acceptable policies of ψ, so that $\Gamma^\varepsilon_\psi(C, d) = \max_{\phi \ne \psi} W_\phi(\varepsilon) - d$.

Proposition S4 (Dual characterization of fixed-tolerance sufficiency)**.** *Let Π be represented by $\mathcal{X}$ and let $\Pi_\psi(d) \ne \emptyset$. Then $W_\phi(\varepsilon) \le d$ if and only if there exist $\mu \ge 0$, $\nu \ge 0$ and a potential $h : S \to \mathbb{R}$ such that, for every (s, a),*

$$
c_\phi(s,a) \ \le\ \mu\, c_\psi(s,a) \ -\ \nu\, r(s,a) \ +\ h(s) \ -\ \gamma \sum_{s'} P(s' \mid s,a)\, h(s'), \tag{6}
$$

and $(1-\gamma)\,\mu_0^\top h + \mu\, d - \nu\,(V_\psi(d) - \varepsilon) \le d$. Hence $\Gamma^\varepsilon_\psi(C, d) \le 0$ if and only if such a certificate exists for every competitor $\phi \in C \setminus \{\psi\}$.

Proof. Inequality (6) is the dual constraint of the face program of Equation (2) written over $\mathcal{X}$, with multipliers h for the flow equations, μ for the budget and ν for the return floor, and the second inequality is its dual objective. The primal is feasible, since an optimizer of $V_\psi(d)$ lies in the face, and bounded, since $\mathcal{X}$ is compact, so strong duality gives $W_\phi(\varepsilon) = \min$ of the dual objective. Weak duality gives the other direction, and the last claim is the definition of $\Gamma^{\varepsilon}_{\psi}(C, d)$. □

Unlike the scalar margin, the triple (h, μ, ν) can be re-verified pointwise on any transition model, which is what the robust-dual certificate of Appendix C.1.2 exploits. The potential term of Inequality (6) integrates to the constant $(1-\gamma)\mu_0^\top h$ against every occupancy, so it shifts every policy's value by the same constant and changes no comparison between policies.

C.2.2 Composition with an approximate learner

The learner contract. The contract asks the learner for a policy within a possibly tightened budget and with near-best return under it. Let $\mathcal{L}$ be the constrained learner and $\eta \in [0, d]$ an optional tightening of the limit. PROTECT, the training step of Algorithm 1, computes $\hat{\pi} = \mathcal{L}(\mathcal{D}, \{c_\phi\}_{\phi \in C}, d-\eta)$ if $\mathcal{L}$ accepts one cost per reading and $\hat{\pi} = \mathcal{L}(\mathcal{D}, c_C, d-\eta)$ with $c_C = \max_{\phi \in C} c_\phi$ otherwise, restricted to covered pairs (those observed often enough in the log) when the offline certificate is used. $\mathcal{L}$ meets the (ε_r, η) *learner contract* on C when $J_{c_\phi}(\hat{\pi}) \le d - \eta$ for every $\phi \in C$ and $J_r(\hat{\pi}) \ge V_C(d-\eta) - \varepsilon_r$. Here $\varepsilon_r \ge 0$ is the learner's reward error under the tightened limit. When a single reading ψ is used for training, the contract is on $\{\psi\}$ and the sufficiency test against a candidate set C uses $\varepsilon' = \varepsilon_r + V_\psi(d) - V_\psi(d-\eta)$.

Proposition S5 (Composition with an approximate learner)**.** *Every policy meeting the (ε_r, η) learner contract on the singleton $\{\psi\}$ lies in $F^{\varepsilon'}_{\psi}(d)$ with $\varepsilon' = \varepsilon_r + V_\psi(d) - V_\psi(d-\eta)$. Hence, for a candidate set $C \subseteq U$, if $\Gamma^{\varepsilon'}_{\psi}(C, d) \le 0$ then $J_{c_\phi}(\hat{\pi}) \le d$ for every $\phi \in C$, and with $\Delta_{C,\psi}(d) = V_C(d) - V_\psi(d)$,*

$$V_C(d) - J_r(\hat{\pi}) \;\le\; \underbrace{\Delta_{C,\psi}(d)}_{\text{anchor}} + \underbrace{\varepsilon_r}_{\text{learner}} + \underbrace{V_\psi(d) - V_\psi(d-\eta)}_{\text{tightening}},$$

where, under this condition, $\Delta_{C,\psi}(d) = 0$ whenever $\psi \in C$.

The effective tolerance ε', the learner's reward error plus the return given up by tightening, is what the sufficiency radius must cover. In the LP case the third term is at most $\eta\, \lambda_\psi(d-\eta)$, where $\lambda_\psi(b)$, the right derivative of V_ψ at limit b, is the shadow price of safety, the return gained per unit of extra budget. By Theorem 4.1(a), $\Gamma^{\varepsilon'}_{\psi}(C, d) \le 0$ whenever $\varepsilon' < \varepsilon^\star_\psi(C, d)$, the equality case being decided by the fixed-tolerance test of Theorem 4.1(b).

Proof. Fix $C \subseteq U$ and recall $\varepsilon' = \varepsilon_r + V_\psi(d) - V_\psi(d-\eta)$. The contract gives $J_{c_\psi}(\hat{\pi}) \le d - \eta \le d$ and $J_r(\hat{\pi}) \ge V_\psi(d-\eta) - \varepsilon_r = V_\psi(d) - \varepsilon'$, so $\hat{\pi} \in F^{\varepsilon'}_{\psi}(d)$. If $\Gamma^{\varepsilon'}_{\psi}(C, d) \le 0$ then $F^{\varepsilon'}_{\psi}(d) \subseteq \Pi_C(d)$ by Theorem 4.1(b), so $J_{c_\phi}(\hat{\pi}) \le d$ for every $\phi \in C$. For the return, if $\varepsilon' > 0$ then for every $0 < \rho \le \varepsilon'$ the definition of the supremum gives $\pi_\rho \in \Pi_\psi(d)$ with $J_r(\pi_\rho) \ge V_\psi(d) - \rho$, so $\pi_\rho \in F^{\varepsilon'}_{\psi}(d) \subseteq \Pi_C(d)$ and $V_C(d) \ge V_\psi(d) - \rho$. Letting $\rho \downarrow 0$ gives $V_C(d) \ge V_\psi(d)$. If $\varepsilon' = 0$ the contract itself gives $J_{c_\psi}(\hat{\pi}) \le d$ and $J_r(\hat{\pi}) \ge V_\psi(d)$, so $\hat{\pi} \in F^0_\psi(d) \subseteq \Pi_C(d)$ and again $V_C(d) \ge V_\psi(d)$. Hence $\Delta_{C,\psi}(d) \ge 0$, and $V_C(d) - J_r(\hat{\pi}) \le V_C(d) - V_\psi(d) + \varepsilon' = \Delta_{C,\psi}(d) + \varepsilon_r + [V_\psi(d) - V_\psi(d-\eta)]$. In the LP case concavity of V_ψ in the budget bounds the bracket by $\eta\, \partial_+ V_\psi(d-\eta) = \eta \lambda_\psi(d-\eta)$. When $\psi \in C$, $\Pi_C(d) \subseteq \Pi_\psi(d)$ gives $V_C(d) \le V_\psi(d)$, hence $\Delta_{C,\psi}(d) = 0$. From a log, $V_\psi(d)$ and $V_\psi(d-\eta)$ are two LPs on the estimated model. Bounding them with the confidence set of Appendix C.1.2 gives an upper confidence bound on ε', and running the certificate at that tolerance keeps Theorem 4.2(a). □

C.2.3 Safety-preserving semantic compression (Safe-Keep)

SAFE-KEEP never changes the constrained problem, because it removes a reading only when a retained reading dominates it in the sense of the redundancy fact of Appendix C.1.4, and its implication relation has that property. For threshold-duration readings compiled to a load level lvl and a duration cap cap (the number of recent observations at that level needed before the reading fires),

$\phi \succeq_{\text{sem}} \psi$ iff $\text{lvl}_\phi \leq \text{lvl}_\psi$ and $\text{cap}_\phi \leq \text{cap}_\psi$. If ψ fires on a history, its last cap_ψ observations are at level at least lvl_ψ. The last cap_ϕ of them are a suffix at level at least $\text{lvl}_\psi \geq \text{lvl}_\phi$, so ϕ fires too. The firing language of ψ is thus contained in that of ϕ and $J_{c_\psi}(\pi) \leq J_{c_\phi}(\pi)$ for every policy and environment consistent with the monitor semantics. The relation is stated on the executable parameters because of how the compiler maps a duration to a counter cap. It rounds to 60 s steps and, when two durations round to the same cap, falls back to their ordinal rank, which need not be monotone in the duration. Pointwise dominance $c_\phi \geq c_\psi$ on every compiled state is sufficient but not necessary, since the compiled product contains counter tuples that no load history realizes. The implication audits are reported in Appendix D.2.

C.2.4 THE SINGLE-COST SURROGATE

For learners that accept one cost signal, the pipeline supplies the pointwise maximum $c_\vee = \max_{\psi \in U_0} c_\psi$, written c_C with $C = U_0$ in Appendix B.3, a sound but incomplete surrogate for full-set protection $\Pi_{U_0}(d)$. Since $c_\psi \leq c_\vee$ pointwise, $J_{c_\psi}(\pi) \leq J_{c_\vee}(\pi)$ for every $\psi \in U_0$ and π, so $J_{c_\vee}(\pi) \leq d$ implies $\pi \in \Pi_{U_0}(d)$. The converse fails because the inequality exchanges a maximum with an expectation. The readings' worst trajectories need not coincide, and a policy can hold every $J_{c_\psi}(\pi)$ below d while $J_{c_\vee}(\pi)$ exceeds it. The surrogate is sound and incomplete, so every exact result in Section 5 solves full-set protection itself. A certified sufficient reading recovers exactly this representation-induced gap for a learner that accepts one cost, and the price is measured in Appendix D.3.

D ADDITIONAL EXPERIMENTAL RESULTS

This appendix gives the full evidence behind Section 5, and Table 9 maps each study to its question, evidence, theory and conclusion. A *reading* is one executable meaning of a written rule, with its own cost (Section 2.1). *Full-set protection* enforces every reading of the *retained set* U_0, the candidates kept after filtering. A reading is *sufficient* when every *acceptable policy* under it, one satisfying it with return within ε of its best, satisfies every retained reading (Section 2.2). Data sources, decision models, protocols and proofs are in Appendices B.1, B.2, B.3 and C. Each result gives why it was run, its setup and its findings, which open with the main takeaway.

D.1 DECISION RELEVANCE AND CLARIFICATION

This subsection asks when different readings change the training decision and how far clarification, targeted questions to the operator (Section 3.2), must go. Appendix D.1.1 solves exactly the compiled monitoring rules, whose readings admit an exact finite model, and the admission pools, candidate sets of Kyverno admission policies. It replicates the verdicts on eight further repositories and scores simple alternatives. Appendix D.1.2 checks the consequence with a live agent and learned maintenance policies, and Appendix D.1.3 counts the questions clarification needs.

D.1.1 DECISION GEOMETRY, REPLICATION AND SIMPLE ALTERNATIVES

Why. This study traces the counts of Table 1 to the cost geometries behind them, replicates them on eight further repositories and scores simple alternatives to certification. The value test, which compares a reading's best return with the best under all retained readings (Section 3.1), disagrees with ARROW's test only when a reading has several optimal policies, some unsafe under another reading. Each single reading is therefore credited with its most favorable tie-break and tested on its whole optimal face, the set of its optimal policies, at several budgets. The alternatives are the top-ranked reading, the strictest reading, the pointwise-maximum cost (the largest retained cost at each state–action pair) and full-set protection.

Setup. Every result is an exact occupancy program on the declared load model, a linear program over the discounted state–action visit frequencies, with no learning. Every distinct compiled geometry of the 28 compiled rules is solved at four budgets. The 465 retained monitoring pools of the eight further repositories of Appendix B.1 are run under the value test and ARROW's test without retuning. The alternatives protocol of Appendix B.3 runs on the 28 compiled monitoring rules and the 44 admission pools at the operating budget.

Table 9: **Appendix roadmap.** Question, evidence, theory link and what each result establishes.

Question	Evidence	Theory link	What it establishes
When do different readings change the training decision?	Table 10, the text of Appendices D.1.1 and D.1.2, Table 12 (exact geometry, eight-repository replication, live agent, learned maintenance policies)	Theorem 4.1	Equal optimal value can hide decision-relevant ambiguity. Sufficiency depends on the policies the learner may return. The learned full-retained-set policy is safer than a single-reading policy at every budget.
Can clarification stop before identification?	Table 13 (decision-focused clarification, return recovery)	Theorem 4.3	Clarification stops once the remaining distinctions no longer change the training representation.
Can simpler alternatives replace the decision?	Table 11 (top-1, strictest reading, pointwise maximum, full-set protection)	Theorem 4.1, Appendix C.2.4	Each alternative addresses one piece of the problem. ARROW decides whether the retained constraints can be reduced.
Can sufficiency be certified from a fixed log?	Table 14 and Figure 2 (three certificates against the witness-based information floor)	Theorem 4.2, Corollary S3	Certification stays one-sided. The decision-specific certificate needs orders of magnitude less data, and the requirement left on the hardest class is intrinsic.
Can redundant readings be removed?	Table 15 (SAFE-KEEP retention and implication audit)	Redundancy fact (Appendix C.1.4), Theorem 4.4(b)	Structured implication shrinks candidate sets without changing the feasible policy set.
What does a sufficient single reading buy downstream?	Figure 3, Tables 16, 17, 18 and 19 (surrogate price, agent-service gains, native learners, public-benchmark portability)	Theorem 4.1, Proposition S5, Appendix C.2.4	Exact for native optimization, simplifies some approximate native learners, removes conservative aggregation for learners that accept one cost.
Can the returned policy be validated before deployment?	Table 20 (CHECK on 3,960 learned policies)	Theorem 4.4(a)	CHECK validates the policy actually returned, for any learner.
Do real requirements keep several plausible meanings?	Table 21 (ARTEMIS coverage and post-redundancy retained sets)	Retention guarantee (Appendix C.1.4), redundancy fact (Appendix C.1.4)	Real requirements retain several distinct readings, the input to ARROW's decision.

Findings. Some rules cannot be reduced to one reading no matter how favorably ties are broken, and whether a rule can be reduced changes with the budget. Eight further repositories show the same picture, and none of the simple alternatives does the whole job. Table 10 lists 17 of the 28 compiled rules, one row per cost geometry (rules sharing one cost structure). The verdict is constant within a geometry and agrees with the value-test column of Table 1. In admission control the value test rules out every single reading on only 2 of 44 pools, while ARROW finds no sufficient reading on 9. So 7 pools pass the value test yet have a tied optimal policy that violates another reading. Across the eight further repositories, 57 of 465 retained pools contain a reading whose constrained value matches full-set protection, yet none is sufficient at $d{=}0.005$. Under the duration-only class one reading is sufficient for 59 of the 60 analyzable rules at the operating budget, and under the flexible class for 5. In Table 11 the top-ranked reading is unsafe on 20 of 44 admission pools and 20 of 28 monitoring rules. A strictest reading exists on 2/44 admission pools and on no compiled monitoring rule. The pointwise maximum is safe but gives up 2.3% of return at the median, the same population priced at five budgets in Table 16. ARROW attains V_U, the full-set optimum, and certifies one reading on 27 of 28 rules and 35 of 44 pools, returning a counterexample on the rest.

Table 10: **What single-reading optimization hides, solved exactly.** h is the pair of duration-counter caps of the compiled model and ν_Π the non-dominance margin, positive when no reading is uniformly strictest. no-feas counts instances with no full-set-feasible single-reading optimum at d=0.01, and orc. worst is the best single reading's worst cost across the retained set, in budget multiples. ✓ marks budgets at which the value test alone requires full-set protection, no single reading matching V_U. ARROW's face test, Table 1, can require it where the value test does not.

						full-set protection at $d =$			
h (steps)	rules	example readings	ν_Π	no-feas	orc. worst	.005	.01	.02	.05
$(2, 1)$	1	(θ=1, 120 s), (θ=2, 0 s)	0.0273	1/1	$3.31d$	✓	✓	✓	✓
$(5, 1)$	4	(θ=0.45, 300 s), (θ=0.5, 0 s)	0.0156	4/4	$1.71d$	✓	✓	✗	✗
$(8, 1)$	12	(θ=0, 900 s), (θ=0.01, 0 s)	0.0045	0/12	$1.00d$	✗	✗	✗	✗

Table 11: **Simple alternatives address different parts of the problem**, on 28 monitoring rules and 44 admission pools at the operating budget.

Alternative	Monitoring (28 rules)	Admission control (44 pools)
top-ranked reading	untied on 10 rules, 2 unsafe; 18 tie at the top: 20/28 unsafe with ties broken toward the lower-threshold reading, 3/28 the other way	unsafe 20/44
single strictest reading (when it exists)	none of 28: the readings cross	exists in 2/44
pointwise-maximum cost	safe, gives up 2.3% median, 6.9% worst	safe, gives up 0.0% median, 16.7% worst
full-set protection	safe, never decides sufficiency	safe, never decides sufficiency
ARROW	one reading in 27/28, otherwise full set	one reading in 35/44, otherwise full set

D.1.2 OPERATIONAL CONSEQUENCES, A LIVE AGENT AND LEARNED MAINTENANCE POLICIES

Why. This study checks that the gap between one reading and the full retained set shows in deployed behavior. A live agent following one reading of a rate limit can obey it exactly while violating other retained readings, and learned maintenance policies should show the gap across domains and budgets.

Setup. The live check runs a GPT-4o-mini agent for 50 paired sessions per condition, posting to a sandboxed endpoint under a published per-channel posting-rate policy with retained burst-cap readings, at most k posts in a burst for $k \in \{3, 4, 5, 6, 8\}$. The agent receives either the single most plausible reading, k=5, or a guard enforcing the full retained set, in which k=3 binds. Every post is a real billed call, and the live-agent experiments billed $0.19 in total. Task value is a fixed keyword score, summing over the 12 messages of a session the share of each message's required points that the posted or fallback text covers. The maintenance sweep re-learns single-reading and full-retained-set policies at five budgets on NASA's C-MAPSS engine-degradation data and a synthetic task (Appendix B.2). The learner is BCQ-FQI-Lagrangian, offline fitted-Q iteration on logged actions with a Lagrange multiplier on cost, run with 15 seeds. Every policy is evaluated against all retained readings.

Findings. An agent given one plausible reading of a rate limit obeyed it and broke the stricter plausible readings in every session, and protecting the whole retained set removed the violations at a lower task value. In both maintenance domains the retained-set policy violated the plausible readings less often than the single-reading policy at every budget. The single-reading agent violated the retained k=3, 4 readings in every session, at a mean task value of 8.02. The retained-set guard violated none, at 6.93. Table 12 shows the learned-policy advantage in all 10 cells.

Table 12: **Budget sweep on the maintenance domains** (return, worst retained cost, realized violation rate).

		single reading			full retained set		
Domain	d	ret.	worst	viol.	ret.	worst	viol.
C-MAPSS	0.02	472.3	0.974	0.600	356.2	0.005	0.058
C-MAPSS	0.05	472.3	0.974	0.600	356.2	0.005	0.058
C-MAPSS	0.10	472.3	0.974	0.600	356.2	0.005	0.058
C-MAPSS	0.20	472.3	0.974	0.600	356.2	0.014	0.064
C-MAPSS	0.50	472.3	0.975	0.600	356.3	0.043	0.084
Maint. (synthetic)	0.02	368.3	0.378	0.423	366.2	0.010	0.045
Maint. (synthetic)	0.05	368.3	0.381	0.423	366.2	0.023	0.048
Maint. (synthetic)	0.10	368.3	0.389	0.423	366.4	0.041	0.054
Maint. (synthetic)	0.20	368.3	0.403	0.423	366.5	0.097	0.080
Maint. (synthetic)	0.50	368.3	0.479	0.423	366.6	0.198	0.184

D.1.3 CLARIFICATION BEFORE IDENTIFICATION

Why. This study tests whether asking only about distinctions that matter for the decision reaches a sufficient reading long before the governing reading $\psi^\dagger$, the one that actually applies, is identified. Such a selector should then stop well before balanced splitting, which divides the survivors evenly until one remains, and its first question should recover most of the return of identification. The study extends Section 5.2 with the remaining selectors, the return recovered per question and the SAFE stop, reached when any retained reading is sufficient for the survivors (the FREE stop needs a surviving one).

Setup. The clarification protocol of Appendix B.3, with simulated truthful answers, runs on the admission pools, the flexible-class monitoring pools and the compiled rules of Table 13. The flexible class decides per monitoring test scenario, the duration-only controller from load level and duration counters alone (Appendix B.2). The random rule is averaged over 20 draws.

Findings. Asking only about the distinctions that matter for the decision reaches a reading that is safe to train on after far fewer questions than identifying the exact reading, and a single well-chosen question recovers most of the return. Decision regions, each the readings under which all acceptable policies of one reading are safe, are few, a median of 4 per admission pool, 13 on the flexible monitoring class and 3 on the duration-only controller. Table 13 reports hyperedge cutting, our selector that splits the readings still blocking sufficiency, at its FREE stop. The compiled rules are taken at d=0.005 because at d=0.05 27/28 of them already have a sufficient reading. One price-guided question, chosen by its effect on return, recovers 89% of the oracle return, the gain from identifying $\psi^\dagger$, on the compiled rules and 65% on the flexible-class pools, more than the balanced split in all three populations. Hyperedge cutting reaches a sufficient reading strictly before its surviving set shrinks to one reading on 56/61 of the flexible-class monitoring pools that need clarification, the count quoted in Section 5.2.

D.2 FIXED-DATA CERTIFICATION AND RETENTION

This subsection certifies sufficiency, with high confidence, from a fixed log with unknown dynamics. Appendix D.2.1 compares three certificates against the witness-based information floor, a lower bound on the data any sound certificate needs. Appendix D.2.2 audits the removals of SAFE-KEEP, which drops readings already enforced by others, and runs the certified-offline pipeline end to end.

D.2.1 CERTIFICATE COMPARISON AND INFORMATION FLOOR

Why. This study tests three predictions of Theorems 4.2 and S2. Every certificate is one-sided, never certifying an insufficient reading. The robust-dual certificate needs far fewer transitions than the generic uniform certificate, which bounds every row's model error uniformly (Corollary S3). It pays for uncertainty only through the spread of its witness, the dual solution proving sufficiency, over each row's successors. The remaining requirement is governed by the decision-information

Table 13: **Stopping versus identification by question count** (fraction of pools), every rule at the FREE stop. Hyperedge cutting's SAFE stop differs only on the flexible class, reaching 95% and 100% after three and four questions against 94% and 99%. It does not pursue identification (–).

		stopped after q					identified after q			
Population	Rule	0	1	2	3	4	1	2	3	4
Admission (44)	hyperedge cutting	80%	100%	100%	100%	100%	–	–	–	–
Admission (44)	price-guided	80%	93%	100%	100%	100%	5%	75%	100%	100%
Admission (44)	witness-restricted	80%	93%	100%	100%	100%	5%	75%	100%	100%
Admission (44)	most balanced split	80%	80%	100%	100%	100%	5%	75%	100%	100%
Admission (44)	random fixture	80%	87%	97%	100%	100%	1%	22%	81%	98%
Monitoring, flexible class (87)	hyperedge cutting	30%	38%	45%	94%	99%	–	–	–	–
Monitoring, flexible class (87)	price-guided	30%	34%	34%	60%	91%	17%	20%	24%	31%
Monitoring, flexible class (87)	witness-restricted	30%	34%	36%	62%	92%	17%	20%	24%	31%
Monitoring, flexible class (87)	most balanced split	30%	37%	37%	43%	48%	13%	18%	30%	37%
Monitoring, flexible class (87)	random fixture	30%	35%	36%	37%	40%	16%	19%	23%	28%
Compiled monitoring, d=0.005 (28)	hyperedge cutting	0%	100%	100%	100%	100%	–	–	–	–
Compiled monitoring, d=0.005 (28)	price-guided	0%	100%	100%	100%	100%	4%	100%	100%	100%
Compiled monitoring, d=0.005 (28)	witness-restricted	0%	100%	100%	100%	100%	4%	100%	100%	100%
Compiled monitoring, d=0.005 (28)	most balanced split	0%	100%	100%	100%	100%	0%	100%	100%	100%
Compiled monitoring, d=0.005 (28)	random fixture	0%	100%	100%	100%	100%	5%	100%	100%	100%

radius $I^\star_\psi$ of Equation (4). The study measures the first two on the compiled rules and bounds the third through a witness chain.

Setup. The fixed-data protocol of Appendix B.3 runs the generic uniform, the row-specific occupancy-weighted and the robust-dual certificate on identical load-chain draws over the 28 compiled rules. The margin κ is how far the true ambiguity margin of Equation (1) lies below zero. At d=0.05 the 27 sufficient readings form two classes, 22 with κ=0.0337 and 5 sharing one compiled geometry with κ=0.0028, reported by class with 50 additional draws for the hard class. 29 readings are insufficient. The witness-based information floor is computed on the true chain from a decision-reversing chain Q, with $\Gamma^{\varepsilon}_{\psi,Q} > 0$, found by directional bisection and local refinement. Its weighted relative entropy $\bar{I}_\psi$ (row-wise Kullback–Leibler divergence weighted by log share) upper-bounds $I^\star_\psi(P)$, so the reported floor $\mathrm{kl}(1-\delta_{\mathrm{dec}}, \delta_{\mathrm{dec}})/\bar{I}_\psi$ is at most the bound of Theorem 4.2(b). It still lower-bounds the transitions any δ_{dec}-sound procedure (wrong with probability at most δ_{dec}) needs to certify ψ under the true chain with probability at least $1-\delta_{\mathrm{dec}}$.

Findings. No certificate ever approved a reading wrongly. The decision-specific certificate needs roughly three thousand to more than ten thousand times less data than the generic one, and the data the hardest readings still require is set by the decision itself, not by the certificate. No insufficient reading, at any budget, log size or draw, is certified by any of the three certificates. A certificate is reliable at a log size when at least 90% of draws certify. The decision-specific robust dual reliably certifies the 22 easier readings from 3.2×10^3 transitions and all 27 from 10^6, against 3.2×10^7 and 10^{10} for the generic certificate, the latter from the archived generic sweep and beyond the grid of Table 14. Occupancy weighting alone reaches only the easier class, so the gain comes from robustifying the dual witness. Against the witness-based information floor (Table 14 and Figure 2), the robust dual needs 14–32 times the floor across the 6 classes and the generic certificate 7.1×10^4–1.5×10^5 times on the 3 classes it reaches. At d=0.10 the one-reading class has the smaller cost margin but the larger witness divergence. The robust dual certifies it in 20% of draws at 10^4 against 0% for the five-reading class, so certification order follows the floor, not the cost margin. For the hard class, certification succeeds in 0% of 60 draws at 10^5 and 100% at 10^6. Sampled load-chain paths in place of fixed row proportions reproduce every conclusion.

D.2.2 SAFE-KEEP AUDIT AND CERTIFIED-OFFLINE DEPLOYMENT

Why. This study checks that SAFE-KEEP shrinks the candidate set before compilation without changing the feasible policy set, the policies satisfying every generated reading. It also runs end to end the certified-offline route (Section 3.3), which deploys without fresh data, on the rules the threshold–duration parser admits.

Table 14: **Three data scales per sufficient-reading class.** Columns give the readings, the margin κ, the witness divergence $\bar{I}_\psi$, the floor $\mathrm{kl}(1-\delta_{\mathrm{dec}}, \delta_{\mathrm{dec}})/\bar{I}_\psi$ and each certificate's smallest reliably certifying grid size, with $> 10^8$ meaning not reached.

d	readings	κ	$\bar{I}_\psi$	floor	uniform	occupancy	robust dual
0.02	22	0.0092	3.64×10^{-3}	728	$> 10^8$	10^7	10^4
0.05	5	0.0028	7.21×10^{-5}	36,734	$> 10^8$	$> 10^8$	10^6
0.05	22	0.0337	1.25×10^{-2}	212	3.2×10^7	3.2×10^6	3.2×10^3
0.10	1	0.0146	2.70×10^{-3}	980	$> 10^8$	3.2×10^7	3.2×10^4
0.10	5	0.0260	1.89×10^{-3}	1,405	10^8	3.2×10^7	3.2×10^4
0.10	22	0.0745	1.88×10^{-2}	141	10^7	3.2×10^6	3.2×10^3

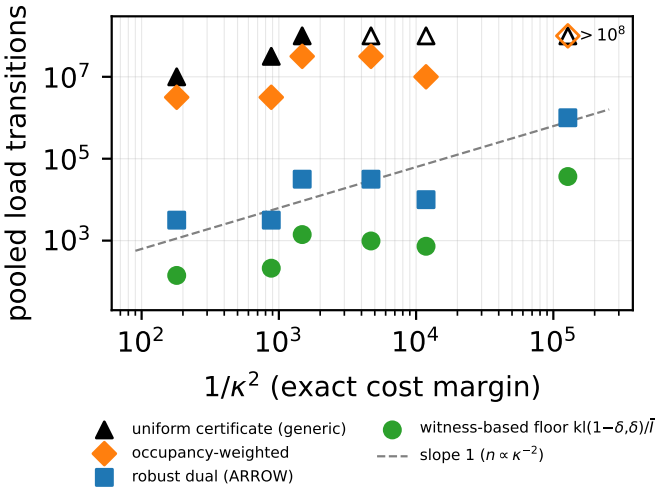

Figure 2: **Certificates and information floor against** $1/\kappa^2$, the classes of Table 14. Open markers denote sizes not reached within the grid, and the dashed line has slope 1.

Setup. SAFE-KEEP retention runs on the 579 admitted families of Appendix B.1, each removal checked by an implication certificate and a linear-program redundancy check. The deployment audit covers 1,600 rule-budget combinations with the penalized occupancy planner of Appendix B.3.

Findings. Removing readings that are implied by other retained readings cuts the typical candidate set from 8 readings to 1, every removal was checked and confirmed, and the full offline pipeline deployed no unsafe policy in 1,600 rule-budget combinations. Table 15 gives the retained-set sizes. By contrast SCORE-KEEP, the plausibility filter of Section 3.2, makes uncovered removals, dropping a reading no retained reading implies, on 46/579 families. It drops the governing reading on 11 of them and retains no reading on 53 families, where the pipeline abstains. The reduction makes the families compilable. Raw pools reach a median of 544,321 joint monitor states and a maximum of 535,692,529, while after SAFE-KEEP the maximum is 55. Every removal was confirmed by the linear-program check, and on the 28 compiled rules SAFE-KEEP leaves the candidate sets unchanged. The certified-offline pipeline, run on 400 seeded rules at 4 budgets, deployed no policy exceeding its budget under the parsed governing reading $\psi^{\dagger}_{\mathrm{parsed}}$, the one the parser produces.

D.3 DOWNSTREAM LEARNING AND DEPLOYMENT

This subsection measures what a sufficient reading buys a learner. Appendix D.3.1 prices exactly the single-cost surrogate, the pointwise-maximum cost a one-cost learner needs to protect all retained readings. It trains six learners under the surrogate and under the selected reading, the one ARROW certifies, and tests native learners that accept one constraint per reading. Appendix D.3.2 ports the

Table 15: **SAFE-KEEP removes redundant readings without losing safety coverage** (579 families). An uncovered removal drops a reading that no retained reading implies. Keep-all and SCORE-KEEP have no implications to confirm (—).

Retention	uncovered removals	median retained	max retained	implications confirmed by LP
keep-all	0	8	10	—
SCORE-KEEP	46/579 (7.9%)	7	10	—
SAFE-KEEP	0/579	1	2	3,969/3,969

interface to five released offline safe-RL learners, and Appendix D.3.3 validates every learned policy before deployment.

D.3.1 EXACT SURROGATE PRICE, LEARNED AGENT-SERVICE GAINS AND NATIVE LEARNERS

Why. This study measures how much return a certified single reading saves, exactly and for learned policies. A one-cost learner gains most, since otherwise it must use the pointwise-maximum cost, whose feasible set can be strictly smaller (Appendix C.2.4). The study prices that loss and trains the six learners of Table 2 under the surrogate and under the selected reading on the agent-service benchmark, whose readings fire on disjoint regions the learner values. For a learner enforcing the retained constraints separately, Theorem 4.1 makes the certified reduction lossless. The study asks whether the reduction still changes finite-sample learning, instantiating the learner-aware route of Proposition S5, which widens ε by the learner's errors.

Setup. The exact price, the gap $V_U - V_{\text{surr}}$ between the full-set optimum and the optimum under the pointwise-maximum cost, is computed on every eligible compiled instance at every budget and on the admission pools. The agent-service domain and learning protocol of Appendix B, where reading ψ_k charges each call to external service k, make the candidate geometry and the exact optimum known. On its 36 (variant, K, d) instances ARROW at tolerance $\varepsilon = 0.01$ finds the primary reading sufficient on 9. These are exactly the instances where every auxiliary demand plus the ε-slack fits within d on its own, and the primary limit binds on every one. The native-learner protocol of Appendix B.3 enforces the K retained constraints separately against the selected reading, for four learners at four data sizes.

Exact price. A learner that accepts only one cost signal loses return when the plausible readings are merged into one worst-case cost, up to 6.91% of the achievable return at the operating budget, and a certified single reading removes that loss. Figure 3 and Table 16 give the exact loss at five budgets over the 28 compiled rules. At the operating budget the gap is positive on all 27 rules with a sufficient reading, 2.26% of V_U at the median. On the 5 rules attaining the maximum, the tightened occupancy planner learns 6.3% less return under the surrogate, and training under the uncertified reading alone is unsafe on every run.

Learned agent-service gains. On every instance where one reading was certified sufficient, all six learners earned more return when trained on that reading than when trained on the merged worst-case cost. BCQ-FQI-Lagrangian, PID-Lagrangian and O3SRL, the three multiplier learners, price cost with a Lagrange multiplier, tuned in the latter two by a proportional-integral-derivative controller and by a bandit. CQL-Lagrangian (conservative Q-learning) runs only at the larger data size. CPQ (constraints penalized Q-learning) and CAPS (constraint-adaptive policy switching) complete the six. Table 2 aggregates by learner. The surrogate is safe on 100% of runs, the selected reading on 99%–100% at 74.5%–97.5% of V_U, and the paired gain is positive on 9/9 instances for every learner (sign test $p = 0.002$). At $n = 2{,}000$ transitions the three multiplier learners and CPQ stay positive on every instance, with median gains of +15.7 to +24.7 points. CAPS is positive on 8/9 and its selected-reading policies are safe on 64% of runs, because the zero-cost local action is absent from the log at the states its switching rule reaches. Table 17 gives the gain per instance, from +4.9 to +53.1 points. It can exceed the exact price because the cost representation also changes the optimization. Under the surrogate the learners fall a further 5.6% to 27.3% short of the exact surrogate optimum.

Sensitivity to alternative settings. We also evaluate the settings of the maintenance domain in full, its limit grids with a 5% action-support threshold alone, and PID-Lagrangian gains at half and twice their value. All runs are released with the outputs. Every learner's gain keeps its sign on every instance under every alternative, except one instance under doubled PID gains. On the maintenance grid the three multiplier learners return the same policy, at 54.6% under the surrogate and 74.3% under the selected reading, against 74.8% and 97.5% for BCQ-FQI-Lagrangian at the main settings. The smallest non-zero multiplier on that grid already makes every charged call unprofitable. Under the 5% support threshold CAPS's selected-reading policies are safe on 90% of runs, against 100% at the main settings.

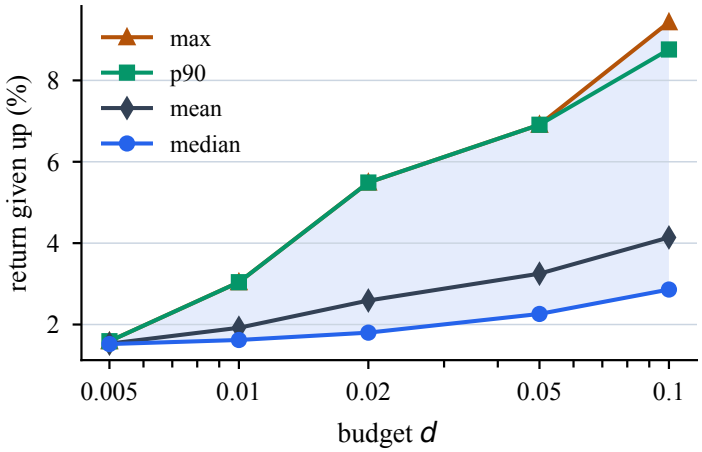

Figure 3: **Exact price of the single-cost surrogate** by budget over the 28 compiled rules. The band runs from the median to the maximum.

Table 16: **Exact price of the single-cost surrogate** by budget over the 28 compiled rules, with the rules whose price exceeds 1%, 2% and 5%.

d	rules	median	mean	p90	max	$> 1\%$	$> 2\%$	$> 5\%$
0.005	28	1.52%	1.53%	1.59%	1.59%	28	0	0
0.01	28	1.62%	1.92%	3.04%	3.04%	28	6	0
0.02	28	1.80%	2.59%	5.49%	5.49%	28	6	6
0.05	28	2.26%	3.25%	6.91%	6.91%	28	28	6
0.1	28	2.86%	4.14%	8.76%	9.43%	28	28	6

Table 17: **Per-instance gains of the ARROW-selected reading over the single-cost surrogate** ($n = 20{,}000$, mean over 10 seeds), in percentage points of the full-set optimum.

Auxiliary demand	K	d	Exact price	BCQ-FQI	PID	O3SRL	CPQ	CAPS	CQL
light	2	0.10	10.2%	+6.3	+11.2	+6.3	+4.9	+11.5	+19.5
light	2	0.15	8.7%	+10.4	+19.0	+10.9	+16.2	+10.4	+24.9
light	3	0.10	18.5%	+15.0	+31.9	+15.0	+19.5	+20.2	+23.4
light	3	0.15	15.9%	+13.9	+20.9	+14.3	+20.7	+18.4	+35.7
light	4	0.10	25.5%	+45.9	+41.5	+45.9	+23.4	+26.5	+42.2
light	4	0.15	21.9%	+22.0	+22.0	+22.0	+21.4	+24.2	+42.1
medium	2	0.15	13.7%	+14.6	+23.8	+14.6	+13.5	+16.0	+37.3
medium	3	0.15	23.8%	+22.7	+19.2	+22.7	+21.1	+26.8	+50.0
medium	4	0.15	28.9%	+53.1	+53.1	+53.1	+31.5	+29.0	+46.2

Native multi-constraint learners. For learners that enforce several readings at once, a certified single reading loses nothing. The exact optimum is unchanged, three of the four learners return the same policies once given moderate data, and the fourth (CPQ) learns better with

Table 18: **Native multi-constraint learners, K separate constraints versus the sufficient reading**, by learner over $n \in \{2{,}000, 5{,}000, 20{,}000, 50{,}000\}$. Gain is the median per-instance difference singleton minus native, in points of V_U.

Learner	native versus singleton	gain	deployment route
vector Lagrangian FQI	identical policies at every n	0	checked deployment
vector PID-Lagrangian FQI	identical policies from n=5,000; median gap 0 at every n	0	checked deployment
vector CPQ	singleton better on 9/9 instances at every n	+3.5 to +5.0 (median)	checked deployment
occupancy LP on $\hat{P}$	same solution from n=5,000	0	learner-aware composition: 6/36 certify at n=5,000, 24/36 from n=20,000

one constraint than with several. Exact optimization agrees on the 9 sufficient instances. In Table 18, vector Lagrangian FQI, with one multiplier per constraint, returns identical policies under the K constraints and under the singleton (the selected reading alone) at every data size, and vector PID-Lagrangian does so from n=5,000. Vector CPQ favors the singleton on every sufficient instance, by +3.5 to +5.0 points at the median, since K cost critics (learned cost estimates) under a common limit forbid auxiliary calls that one critic admits. The occupancy planner converges to the same solution from n=5,000, and its learner-aware route certifies 6/36 instances there and 24/36 from n=20,000.

D.3.2 Portability to released safe-RL learners

Why. ARROW's interface should work unmodified with released offline safe-RL implementations on a public benchmark, since it changes only the constraint representation supplied to the learner.

Setup. The learners are BCQ-Lagrangian (Fujimoto et al., 2019; Stooke et al., 2020), the release's own implementation, distinct from the tabular BCQ-FQI-Lagrangian of Appendix B.3, CPQ (Xu et al., 2022), COptiDICE (Lee et al., 2022), the constrained decision transformer (CDT) (Liu et al., 2023b) and constraint-adaptive policy switching (CAPS) (Chemingui et al., 2025b). They run on the DSRL benchmark datasets, using its public OSRL learner library (Liu et al., 2024) and, for CAPS, the authors' release built on OSRL. Only their standard cost-limit (c) and training controls change (20 seeds per cell for the c=40 and retained-set columns, 3 seeds for the c=20 column). The benchmark's paired readings are nested, one stricter than the other, so the experiment isolates learner portability. The non-nested geometry is evaluated in the exact and agent-service studies.

Findings. Five released offline safe-RL learners work with retained-set protection unmodified, and in 20 of the 23 task–learner cells (of 25) where the single-reading limit breaks a plausible reading, protection lowers the worst cost. Table 19 reports reward and worst normalized cost per task and learner. In 23 cells the c=40 policy violates the strictest retained reading. Protection lowers that cost in 20 of them, the 3 exceptions being BCQ-Lagrangian on AntRun (9.07→9.12), BCQ-Lagrangian on BallCircle (3.71→3.78), COptiDICE on DroneCircle (1.06→1.14). CPQ fails on DroneCircle under every limit, which is why its returns there are negative on DSRL's scale.

D.3.3 Deployment checking

Why. This study tests whether the checked route of Theorem 4.4 validates the policy an approximate learner actually returns. On that route CHECK bounds each retained reading's cost from fresh episodes and deploys only if every bound is within budget (Section 3.3). It is evaluated on every learned agent-service policy, including, as a stress test, wrong-reading policies trained under a single auxiliary-service reading that ARROW's test found insufficient.

Setup. The CHECK protocol of Appendix B.3, with empirical-Bernstein bounds (variance-adaptive confidence bounds) Bonferroni-corrected across readings, runs on all 3,960 policies of the single-cost study.

Table 19: **Portability across five released offline safe-RL learners on DSRL/OSRL.** Reward is on DSRL's normalized scale, where 0 is the task's reference minimum return and 1 its maximum, so a negative entry marks a learner that returns less than the reference minimum. Cost is divided by the reading's limit, so a value above 1 violates it, and worst is the maximum over the retained readings.

		Norm. Reward ↑			Worst Norm. Cost ↓		
Task	Learner	c=40	c=20	retained set	c=40	c=20	retained set
AntRun	BCQ-Lagrangian	0.77 ± 0.07	0.78 ± 0.02	0.78 ± 0.09	9.07 ± 2.46	8.79 ± 0.64	9.12 ± 3.03
AntRun	CPQ	0.03 ± 0.02	0.02 ± 0.01	0.03 ± 0.03	0.03 ± 0.08	0.00 ± 0.00	0.02 ± 0.08
AntRun	COptiDICE	0.61 ± 0.02	0.61 ± 0.00	0.60 ± 0.02	1.85 ± 0.70	1.53 ± 0.59	1.75 ± 0.70
AntRun	CDT	0.73 ± 0.01	0.73 ± 0.00	0.72 ± 0.01	3.30 ± 0.51	1.61 ± 0.10	1.03 ± 0.10
AntRun	CAPS	0.69 ± 0.02	0.60 ± 0.04	0.57 ± 0.08	3.19 ± 0.73	1.16 ± 0.30	1.13 ± 0.58
BallCircle	BCQ-Lagrangian	0.70 ± 0.04	0.69 ± 0.01	0.70 ± 0.03	3.71 ± 0.69	3.61 ± 0.66	3.78 ± 0.50
BallCircle	CPQ	0.66 ± 0.07	0.59 ± 0.02	0.51 ± 0.21	3.11 ± 0.69	1.86 ± 1.10	1.70 ± 3.41
BallCircle	COptiDICE	0.71 ± 0.02	0.71 ± 0.01	0.70 ± 0.02	4.51 ± 0.46	4.75 ± 0.26	4.48 ± 0.46
BallCircle	CDT	0.85 ± 0.01	0.76 ± 0.01	0.72 ± 0.01	3.86 ± 0.09	2.04 ± 0.14	1.40 ± 0.16
BallCircle	CAPS	0.80 ± 0.01	0.68 ± 0.02	0.61 ± 0.01	3.38 ± 0.04	1.35 ± 0.07	0.37 ± 0.03
CarCircle	BCQ-Lagrangian	0.74 ± 0.04	0.61 ± 0.01	0.58 ± 0.07	5.09 ± 1.16	4.10 ± 0.04	3.46 ± 0.99
CarCircle	CPQ	0.72 ± 0.02	0.70 ± 0.01	0.70 ± 0.02	2.01 ± 0.93	0.04 ± 0.06	0.32 ± 0.80
CarCircle	COptiDICE	0.50 ± 0.02	0.49 ± 0.02	0.49 ± 0.02	5.85 ± 0.61	6.73 ± 0.47	5.77 ± 0.56
CarCircle	CDT	0.78 ± 0.02	0.72 ± 0.00	0.71 ± 0.01	3.30 ± 0.24	1.70 ± 0.15	0.96 ± 0.26
CarCircle	CAPS	0.73 ± 0.02	0.68 ± 0.01	0.60 ± 0.04	3.49 ± 0.11	1.52 ± 0.07	0.46 ± 0.07
CarRun	BCQ-Lagrangian	0.93 ± 0.08	0.91 ± 0.09	0.94 ± 0.05	1.22 ± 2.74	0.02 ± 0.02	0.98 ± 3.03
CarRun	CPQ	0.94 ± 0.06	0.95 ± 0.04	0.86 ± 0.27	4.87 ± 2.83	3.15 ± 1.32	0.99 ± 1.97
CarRun	COptiDICE	0.89 ± 0.05	0.93 ± 0.02	0.90 ± 0.05	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
CarRun	CDT	0.99 ± 0.00	0.99 ± 0.00	0.99 ± 0.00	2.90 ± 0.90	1.45 ± 0.20	0.91 ± 0.33
CarRun	CAPS	0.98 ± 0.01	0.98 ± 0.01	0.97 ± 0.01	1.36 ± 1.25	0.66 ± 0.85	0.42 ± 0.27
DroneCircle	BCQ-Lagrangian	0.93 ± 0.02	0.83 ± 0.08	0.70 ± 0.06	9.30 ± 0.31	6.97 ± 1.57	4.35 ± 1.62
DroneCircle	CPQ	-0.22 ± 0.03	-0.23 ± 0.02	-0.23 ± 0.03	1.50 ± 0.80	1.28 ± 1.15	1.08 ± 0.68
DroneCircle	COptiDICE	0.24 ± 0.03	0.27 ± 0.02	0.25 ± 0.03	1.06 ± 0.81	1.34 ± 0.14	1.14 ± 0.76
DroneCircle	CDT	0.69 ± 0.01	0.61 ± 0.00	0.57 ± 0.01	3.93 ± 0.12	2.30 ± 0.05	1.09 ± 0.16
DroneCircle	CAPS	0.62 ± 0.01	0.55 ± 0.01	0.51 ± 0.02	3.54 ± 0.07	1.44 ± 0.08	0.54 ± 0.08

Findings. The pre-deployment check let no unsafe policy through among all 3,960 learned policies tested, and, with one unsafe exception, every selected-reading policy it rejected at the largest data size was within the budget. Table 20 shows every unsafe policy rejected, the 1 unsafe CQL selected-reading policy and the wrong-reading policies included. At n=20,000 the passing selected-reading policies deploy at 74.5%–96.7% of V_U by learner. The 92 safe selected-reading policies rejected among 540 runs cost up to 99.8% of the budget, too close for a bound from n_{ev}=20,000 fresh samples to certify.

Table 20: **Checked deployment on all learned agent-service policies**, per learner at n=20,000 (90 policies per training cost) and, pooled, the wrong-reading policies. Returns are those of Table 2. The pooled row's unsafe count and pass share stand in its first two columns. Of the 900 n=2,000 policies, not listed, CHECK rejected all 33 unsafe ARROW-reading policies, and all 450 surrogate policies were safe and passed.

	ARROW reading		single-cost surrogate		
Learner	unsafe	pass	unsafe	pass	unsafe deployed
BCQ-FQI-Lagrangian	0	71%	0	100%	0
PID-Lagrangian	0	89%	0	100%	0
O3SRL	0	71%	0	100%	0
CPQ	0	96%	0	100%	0
CAPS	0	100%	0	100%	0
CQL-Lagrangian (neural)	1	70%	0	100%	0
wrong readings, all learners (1,980 policies)	1,815	5%	—	—	0

D.4 REAL REQUIREMENTS

Why. This study asks two questions behind Table 3. Does SCORE-KEEP keep plausible readings while discarding most candidates, and do the retained sets still contain several distinct readings after redundancy removal? Section 5.4 makes this comparison with language-to-specification generators on ARTEMIS, an external benchmark whose plausible readings were enumerated independently of our pipeline, using candidate pools produced by the prior translation methods themselves.

Setup. The ARTEMIS protocol of Appendix B.3 runs on 182 expert-annotated ventilator, human–robot teamwork and Lockheed Martin cyber-physical (LMCPS) requirements. The pool is the union of 9 generators' archived translations into linear temporal logic (LTL), a formal language for properties of executions over time, matched by exact LTL equivalence. SCORE-KEEP runs at δ_{sem}=0.10, its allowed probability of dropping the governing reading.

Findings. Keeping a small set of candidate translations preserves a reading the domain experts accept for 73.1% of real requirements, against 32.4% when only ARTEMIS + reflection's first translation is kept, and most of these sets still contain several different readings. Table 3 reports, per domain group, the share of requirements on which a method's committed (single output) translation is expert-plausible, 35.7% for the best method against 73.1% for SCORE-KEEP. Table 21 pools the groups and adds expert recall, the share of expert-plausible readings retained. The retained set keeps a plausible reading on 73.1% of the requirements against 80.2% for the unfiltered pool, at a median size of 11.5 readings against 60.5. It recovers 37.1% of the expert-plausible readings against 13.0% for the first translation. SAFE-KEEP's redundancy removal, dropping every retained reading that another entails (satisfying the other guarantees it) as decided by the same exact checker, leaves the SCORE-KEEP sets with a median of 4 readings (at most 8). They hold one reading on 12.6% of the requirements and at most two on 31.3%, and the strongest readings are jointly satisfiable on 75.8%.

Table 21: **Expert-plausible readings preserved** on 182 ARTEMIS requirements, the union pool of all archived translations, with generation misses counted against every method.

Method	any plausible	expert recall	median retained classes
First translation, ARTEMIS + reflection	32.4%	13.0%	1
Self-consistency singleton	28.0%	13.5%	1
ARROW SCORE-KEEP	73.1%	37.1%	11.5
Unfiltered candidate pool	80.2%	57.5%	60.5

Per-method validation. Our any-of-ten computation reproduces ARTEMIS's published over-best-baseline ratios except the first (1.95/1.57/1.25/1.60/1.73/1.08 here versus 1.90/1.57/1.25/1.60/1.73/1.08 reported), a requirement counting as covered when any of ten samples is plausible. SCORE-KEEP is selective mainly on the pooled union. On one method's samples the choice-frequency score, each equivalence class's sample frequency, takes few values, so SCORE-KEEP's median retained size equals the pool's on 12 of the 15 archives.

Generation error. Generation misses, requirements with no plausible candidate, number 36/182 and shrink as candidates are pooled ($\hat{\delta}_{\text{gen}} = 19.8\%$, one-sided 95% Clopper–Pearson exact binomial upper bound 25.3%). Coverage rises from 28.0% with one candidate to 69.8% with ten and 80.2% with all archived samples. Pooling generators raises all-sample coverage from 67.0% to 80.2%. In the admitted threshold–duration family, the rules the parser accepts, deterministic parsing gives $\delta_{\text{gen}} = 0$ conditional on admission, after which SAFE-KEEP preserves the feasible policy set.