BUDGET-FIRST CONFORMAL ROBUSTNESS CONTROL UNDER HARD DEPLOYMENT BUDGETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Decision-making usually relies on uncertainty sets for robust decisions, and existing methods like conformal robustness control (CRC) typically return an uncertainty set at a fixed coverage or robustness level. However, the uncertainty set may violate pointwise deployment requirements such as per-decision cost limits or set-size budgets. In this paper, we propose *Budget-First Conformal Robustness Control* (BCRC) to satisfy deployment budgets under both continuous and discrete regimes. Our main contribution is a budget-first conformal robustness framework that learns the deployment rule induced by the hard budget. In contrast to this risk-first order, BCRC works backward from the hard budget to construct the maximal feasible candidate set for an observed input. For the resulting budget-feasible deployment, we provide a finite-sample upper bound on the failure probability of the exact deployed rule using an independent certification set. Across various experiments, BCRC consistently satisfies the deployment budget and achieves lower failure risk than several baseline methods under the same budget.

1 INTRODUCTION

Uncertainty-aware decision-making is increasingly deployed to support downstream actions under ambiguity, rather than producing a single output (Vovk et al., 2005; Bates et al., 2021). In robust decision-making, a decision-maker constructs an uncertainty set for possible outcomes and then chooses an action that remains safe against all outcomes in the set (Bertsimas et al., 2011). In human-in-the-loop systems, a model may return a shortlist of plausible labels, answers, or actions for downstream inspection (Sadinle et al., 2019; Cresswell et al., 2024). However, real deployment environments often impose hard budgets: a human reviewer may inspect at most T candidates for each case, while a robust decision system may require every decision certificate to stay below a prescribed cost limit. These constraints are pointwise, requiring the budget to hold for each deployed output rather than only on average. Hence, the central question is how to retain decision-relevant uncertainty under a hard output budget.

Recent decision-aware methods connect uncertainty sets to downstream robust decisions (Johnstone & Cox, 2021; Sun et al., 2023; Chenreddy & Delage, 2024; Patel et al., 2024). Kiyani et al. (2025) formulate the underlying risk-averse decision problem through Risk-Averse Decision Policy Optimization (RA-DPO), in which a decision maker chooses an action policy together with a utility certificate under a prescribed risk tolerance. A recent follow-up, Conformal Robustness Control (CRC) (Hu et al., 2026), builds on this decision formulation and uses parameterized prediction sets to obtain a tractable optimization method for continuous decision spaces. Let $z(X)$ denote the decision, $\phi(Y, z)$ the downstream loss, and $r(X)$ the corresponding loss certificate. The robustness requirement is

$$\mathbb{P}\{\phi(Y, z(X)) \leq r(X)\} \geq 1 - \alpha, \tag{1}$$

where α is a prescribed failure tolerance, and the decision maker seeks to minimize $r(X)$ to reduce potential worst-case losses. These risk-first formulations prescribe a risk or coverage level, while the deployment cost is induced afterward.

A hard deployment budget introduces an additional pointwise feasibility constraint. For a prescribed α, the risk requirement and the deployment budget need not be simultaneously feasible: achieving the desired reliability may require a certificate value or output size that exceeds the available budget.

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.

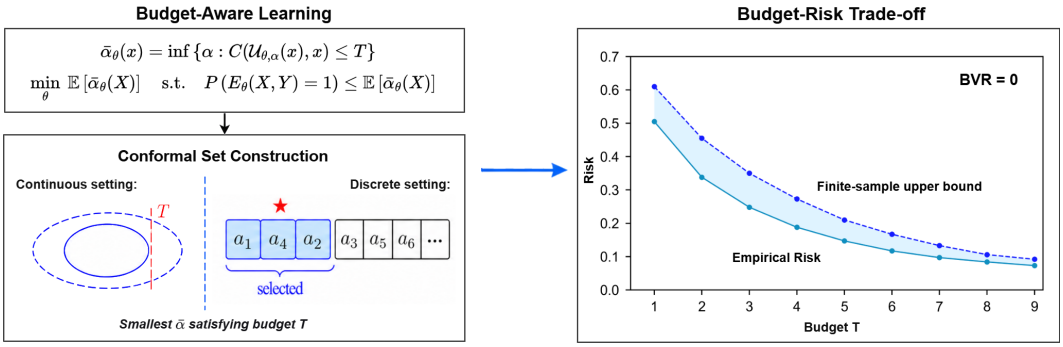


Figure 1: Overview of BCRC under a hard deployment budget T. Left: BCRC learns a parameterized candidate-set family for the budget-induced deployed rule and constructs a pointwise budget-feasible output for each input, covering continuous uncertainty sets and discrete shortlists. Right: after the deployed rule is fixed, independent data provide a finite-sample upper bound on its failure risk. Pointwise budget feasibility holds by construction.

We therefore consider the complementary budget-first setting, in which the hard budget is fixed first and reliability is optimized over decision rules that satisfy it pointwise.

To address this budget-first problem, we develop *Budget-First Conformal Robustness Control* (BCRC). For each input, the hard budget selects the largest feasible member of a parameterized nested candidate family, which directly determines the deployed decision and its failure event. BCRC learns the family parameters for this budget-induced deployed rule. After training, we reconstruct the exact budget-feasible rule and certify its deployment failure risk on independent data.

2 Related Work

Conformal methods for robust decision-making. Conformal uncertainty sets have been incorporated into robust and contextual decision problems (Johnstone & Cox, 2021; Sun et al., 2023; Patel et al., 2024; Chenreddy et al., 2022; Chenreddy & Delage, 2024). More recent decision-aware methods optimize or calibrate the robustness event induced by the final decision and certificate rather than coverage alone (Kiyani et al., 2025; Hu et al., 2026). BCRC instead considers deployment under a hard pointwise budget and certifies the risk of the resulting budget-feasible rule.

Size-constrained and finite-candidate set prediction. Conformal methods improve set efficiency by learning scores or set families and by optimizing expected set size or length (Stutz et al., 2022; Bai et al., 2022; Tumu et al., 2024; Dhillon et al., 2024; Kiyani et al., 2024; Noorani et al., 2026). Set-valued prediction and top-k learning also study prediction under cardinality constraints (Sadinle et al., 2019; Lei, 2014; Chzhen et al., 2021; Lapin et al., 2016; Berrada et al., 2018; Cortes et al., 2024). Backward Conformal Prediction adapts the miscoverage level to a prescribed set-size constraint using a scoring model fixed during calibration (Gauthier et al., 2025a). BCRC instead learns the candidate family for a prescribed pointwise budget, which may represent either shortlist cardinality or a decision-dependent robustness certificate, and separately certifies the resulting deployment failure risk.

Additional related work on conformal risk control and post-hoc certification is discussed in Appendix A.

3 Budget-First Conformal Robustness Control

BCRC unifies continuous robust decision-making and discrete hard-budget shortlisting by treating the deployment budget as the primary constraint. Instead of fixing a nominal level α and accepting the induced deployment cost, we enforce the deployment budget first and infer the truncation level

required by that budget. The resulting budget-feasible rule is learned using differentiable surrogates and certified post-hoc on an independent certification set.

3.1 Problem Setup

Risk-first and budget-first decision problems. Let $\mathcal{X}$ denote the feature space. For each $x \in \mathcal{X}$, let $\mathcal{Y}(x)$ denote the corresponding outcome space, and let $(X, Y) \sim P$ with $Y \in \mathcal{Y}(X)$. Let $z : \mathcal{X} \to \mathcal{Z}$ denote the decision policy, $\phi(Y, z(X))$ the downstream loss, and $r : \mathcal{X} \to \mathbb{R}$ the associated loss-certificate function. For each input x, $r(x)$ is an input-dependent loss threshold that is optimized jointly with the decision $z(x)$. Kiyani et al. (2025) formulate Risk-Averse Decision Policy Optimization (RA-DPO) as

$$\min_{z(\cdot),\, r(\cdot)} \mathbb{E}[r(X)] \qquad \text{s.t.} \qquad \mathbb{P}\{\phi(Y, z(X)) \le r(X)\} \ge 1 - \alpha, \tag{2}$$

where α is a prescribed failure tolerance.

We additionally consider a deployment cost $\mathcal{C}$ and a hard budget T. A deployed decision must satisfy

$$\mathcal{C}(z(x), r(x), x) \le T, \qquad \forall x \in \mathcal{X}. \tag{3}$$

The function $\mathcal{C}$ measures the quantity constrained at deployment. For example, if each decision is required to keep its loss certificate below a fixed limit, then $\mathcal{C}(z(x), r(x), x) = r(x)$, and (3) reduces to $r(x) \le T$.

The risk requirement in (2) is statistical and marginal, whereas (3) is operational and pointwise. For a prescribed α, the two requirements need not be simultaneously feasible. We therefore define the budget-first risk level as

$$\alpha^\star(T) := \inf_{\substack{z(\cdot),\, r(\cdot) \\ \mathcal{C}(z(x), r(x), x) \le T,\ \forall x \in \mathcal{X}}} \mathbb{P}\{\phi(Y, z(X)) > r(X)\}. \tag{4}$$

Equation (4) defines the oracle budget-first risk. Thus, $\alpha^\star(T)$ is the smallest achievable failure level under the pointwise budget T. If $T_1 \le T_2$, then every rule feasible under T_1 is also feasible under T_2, so $\alpha^\star(T_2) \le \alpha^\star(T_1)$. Because BCRC restricts deployment to a parameterized nested family, $\alpha^\star(T)$ serves as an oracle benchmark, while BCRC optimizes within the specified family.

Prediction-set representation. Directly optimizing over arbitrary decision and certificate functions is generally difficult. We therefore represent the decision rule through an uncertainty set $\mathcal{U}(x) \subseteq \mathcal{Y}(x)$. In the continuous setting, $\mathcal{Y}(x) \subseteq \mathbb{R}^q$, and $\mathcal{U}(x)$ can be a structured region such as a box or an ellipsoid. In the discrete setting, $\mathcal{Y}(x) = \{a_1, \ldots, a_{m(x)}\}$, and $\mathcal{U}(x)$ is a shortlist of candidate outputs.

Given $\mathcal{U}(x)$, the robust decision and its loss certificate are

$$z_{\mathcal{U}}(x) \in \arg\min_{z \in \mathcal{Z}} \max_{y \in \mathcal{U}(x)} \phi(y, z), \qquad r_{\mathcal{U}}(x) = \max_{y \in \mathcal{U}(x)} \phi(y, z_{\mathcal{U}}(x)). \tag{5}$$

Hence, $r_{\mathcal{U}}(x)$ is the worst-case loss of the robust decision over $\mathcal{U}(x)$. We denote the induced deployment cost by $\mathcal{C}(\mathcal{U}(x), x) := \mathcal{C}(z_{\mathcal{U}}(x), r_{\mathcal{U}}(x), x)$. When the deployed output is a discrete shortlist, the corresponding cost is its cardinality, $\mathcal{C}(\mathcal{U}(x), x) = |\mathcal{U}(x)|$. When the deployment cost depends only on the loss certificate and is nondecreasing in r, this prediction-set representation does not restrict the budget-first problem. A feasible pair (z, r) induces the set $\mathcal{U}_{z,r}(x) = \{y \in \mathcal{Y}(x) : \phi(y, z(x)) \le r(x)\}$, while any uncertainty set $\mathcal{U}$ induces $(z_{\mathcal{U}}, r_{\mathcal{U}})$ through (5). We give the formal statement in Appendix C.

Budget-induced deployment rule. We now restrict the uncertainty sets to a parameterized nested family $\{\mathcal{U}_{\theta,\alpha}(x) : \alpha \in (0,1),\ \theta \in \Theta\}$, where Θ is the parameter space and θ controls the set geometry or candidate ranking. The level α indexes the nested path, and we assume $\alpha_1 \le \alpha_2 \implies \mathcal{U}_{\theta,\alpha_1}(x) \supseteq \mathcal{U}_{\theta,\alpha_2}(x)$. We also assume that the deployment cost is monotone with respect to set inclusion.

Throughout this section, T is fixed and omitted from the notation. For a given θ, the budget determines the smallest level that makes the set feasible:

$$\bar{\alpha}_\theta(x) := \inf\{\alpha \in (0,1) : \mathcal{C}(\mathcal{U}_{\theta,\alpha}(x), x) \le T\}. \tag{6}$$

When the feasible set in (6) is nonempty, let $\bar{\alpha}_\theta(x)$ denote its smallest feasible level. In numerical implementation, we use the feasible-side approximation returned by the backward search. The deployed set is then $\mathcal{U}^\star_\theta(x) := \mathcal{U}_{\theta,\bar{\alpha}_\theta(x)}(x)$. Under the nesting and monotonicity assumptions, $\mathcal{U}^\star_\theta(x)$ is the largest member of the candidate family that satisfies the budget. If the feasible set is empty, we set $\bar{\alpha}_\theta(x) = 1$ for the learning objective and use a predefined budget-feasible fallback as $\mathcal{U}^\star_\theta(x)$. Here, the value 1 is an objective convention; the deployed set is the fallback rather than $\mathcal{U}_{\theta,1}(x)$.

Except for this fallback convention, $\bar{\alpha}_\theta(x)$ is an input-dependent set level induced by the budget; it is not the failure probability of the deployed rule. Since θ changes the set geometry or candidate ranking, it also changes $\mathcal{U}^\star_\theta(x)$ under the same budget. BCRC learns θ for this budget-feasible rule. In the discrete case with $\mathcal{C}(\mathcal{U}(x), x) = |\mathcal{U}(x)|, \mathcal{U}^\star_\theta(x)$ reduces to the corresponding Top-T candidate set. Thus, cardinality-constrained shortlisting is a special case of the same budget-induced construction used for general pointwise deployment costs.

3.2 Budget-Aware Learning

Deployment failure. The budget-feasible deployed rule induces a task-specific failure indicator $E_\theta(x, y) \in \{0, 1\}$. In the continuous robust-decision setting, define $z_\theta(x) := z_{\mathcal{U}^\star_\theta}(x)$ and $r_\theta(x) := r_{\mathcal{U}^\star_\theta}(x)$ according to (5). In the discrete setting, let $\mathcal{G}(x, y) \subseteq \mathcal{Y}(x)$ denote the acceptable-candidate set for example (x, y), with $\mathcal{G}(x, y) = \{y\}$ in single-label classification or multiple-choice tasks. The failure indicator is

$$E_\theta(x, y) = \begin{cases} \mathbf{1}\{\phi(y, z_\theta(x)) > r_\theta(x)\}, & \text{continuous robust decision-making,} \\ \mathbf{1}\{\mathcal{U}^\star_\theta(x) \cap \mathcal{G}(x, y) = \varnothing\}, & \text{finite-candidate shortlisting.} \end{cases} \tag{7}$$

Rather than directly optimizing the unrestricted oracle risk in (4), BCRC searches over the parameterized nested family and uses $\mathbb{E}[\bar{\alpha}_\theta(X)]$ as a budget-induced learning criterion. Within a fixed nested family, $\bar{\alpha}_\theta(x)$ records the level at which the hard budget becomes feasible. We minimize this quantity while explicitly requiring the actual deployment failure risk not to exceed it:

$$\min_{\theta\in\Theta} \mathbb{E}[\bar{\alpha}_\theta(X)] \quad \text{s.t.} \quad \mathbb{P}(E_\theta(X, Y) = 1) \le \mathbb{E}[\bar{\alpha}_\theta(X)]. \tag{8}$$

Within the specified nested family, $\mathbb{E}[\bar{\alpha}_\theta(X)]$ serves as the budget-induced learning target. Because this target need not rank candidate rules identically to deployment failure risk, BCRC explicitly couples it to the actual failure constraint above. Appendix C.1 establishes an oracle inequality relating the budget-induced target to deployment risk. The distribution-free post-hoc certificate is provided separately after the deployed rule has been fixed.

Given an i.i.d. training set $\mathcal{D}_{\mathrm{tr}} = \{(x_i, y_i)\}_{i=1}^n$ drawn from P, its empirical counterpart is

$$\min_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^n \bar{\alpha}_\theta(x_i) \quad \text{s.t.} \quad \frac{1}{n}\sum_{i=1}^n E_\theta(x_i, y_i) \le \frac{1}{n}\sum_{i=1}^n \bar{\alpha}_\theta(x_i). \tag{9}$$

The empirical problem in (9) is exact but nonsmooth: $\bar{\alpha}_\theta(x)$ is defined by a budget threshold, and $E_\theta(x, y)$ is a binary failure indicator. We therefore use smooth surrogates only for learning the model parameter θ. After training, the surrogates are discarded, and the final deployed rule is constructed from the exact budget-induced level in (6).

Lagrangian surrogate training. As both the budget-induced level $\bar{\alpha}_\theta(x)$ and the failure indicator $E_\theta(x, y)$ involve discontinuous thresholding operations, we use training relaxations that are differentiable almost everywhere, $\widetilde{\bar{\alpha}}_\theta(x)$ and $\widetilde{E}_\theta(x, y)$. In the continuous robust-decision setting, we use sigmoid relaxations, while in the discrete finite-candidate setting we use task-specific ranking relaxations. The precise surrogate definitions are given in Appendix D. We optimize a primal-dual training relaxation with the following Lagrangian objective, where $\lambda \ge 0$ is the multiplier:

$$\min_\theta \max_{\lambda\ge 0} \frac{1}{n}\sum_{i=1}^n \widetilde{\bar{\alpha}}_\theta(x_i) + \lambda \left[\frac{1}{n}\sum_{i=1}^n \widetilde{E}_\theta(x_i, y_i) - \frac{1}{n}\sum_{i=1}^n \widetilde{\bar{\alpha}}_\theta(x_i)\right]. \tag{10}$$

We denote the expression inside the min–max problem in (10) by $\mathcal{L}(\theta, \lambda)$. The multiplier penalizes violations of the relaxed empirical constraint, so the model parameter θ is optimized with respect to the budget-induced failure criterion.

Algorithm 1: Budget-Aware Training and Budget-Feasible Rule Construction

Input: Training set $\mathcal{D}_{\mathrm{tr}} = \{(x_i, y_i)\}_{i=1}^n$, candidate family $\mathcal{U}_{\theta,\alpha}$, budget function $\mathcal{C}$, budget threshold T, smooth surrogates $\widetilde{\bar{\alpha}}_\theta, \widetilde{E}_\theta$, learning rates $\eta_\theta, \eta_\lambda$, initial parameter θ_0, iterations N_{iter}

Output: Learned parameter $\hat{\theta}$ and budget-feasible deployed rule $x \mapsto \mathcal{U}^\star_{\hat{\theta}}(x)$

Training stage;
Initialize $\theta \leftarrow \theta_0$, $\lambda \leftarrow 0$;
for $t = 1, \ldots, N_{\mathrm{iter}}$ **do**
  Compute $\widetilde{\bar{\alpha}}_\theta(x_i)$ and $\widetilde{E}_\theta(x_i, y_i)$ on $\mathcal{D}_{\mathrm{tr}}$;
  Compute $\mathcal{L}(\theta, \lambda) = \frac{1}{n}\sum_{i=1}^n \widetilde{\bar{\alpha}}_\theta(x_i) + \lambda\left(\frac{1}{n}\sum_{i=1}^n \widetilde{E}_\theta(x_i, y_i) - \frac{1}{n}\sum_{i=1}^n \widetilde{\bar{\alpha}}_\theta(x_i)\right)$;
  Update $\theta \leftarrow \theta - \eta_\theta \nabla_\theta \mathcal{L}(\theta, \lambda)$;
  Update $\lambda \leftarrow \max\left\{0,\ \lambda + \eta_\lambda\left(\frac{1}{n}\sum_{i=1}^n \widetilde{E}_\theta(x_i, y_i) - \frac{1}{n}\sum_{i=1}^n \widetilde{\bar{\alpha}}_\theta(x_i)\right)\right\}$;
Set $\hat{\theta} \leftarrow \theta$;

Budget-feasible rule construction;
For any input x, search for a feasible level in (6);
if *a feasible level exists* **then**
  Set $\bar{\alpha}_{\hat{\theta}}(x)$ to the smallest feasible level;
  Set $\mathcal{U}^\star_{\hat{\theta}}(x) = \mathcal{U}_{\hat{\theta}, \bar{\alpha}_{\hat{\theta}}(x)}(x)$;
else
  Set $\bar{\alpha}_{\hat{\theta}}(x) \leftarrow 1$;
  Set $\mathcal{U}^\star_{\hat{\theta}}(x)$ to the predefined budget-feasible fallback;
return $\hat{\theta}$ *and the deployed rule* $x \mapsto \mathcal{U}^\star_{\hat{\theta}}(x)$;

3.3 Training-Stage Generalization

We next bound the population failure risk of the exact budget-feasible rule reconstructed after surrogate training. The result applies to any measurable training output through its exact empirical feasibility residual; the modality-specific regularity conditions and deviation terms are given in Appendix E.

Fix $\delta \in (0, 1)$, and in the continuous setting fix a tolerance $\varepsilon > 0$. Let Δ_n denote the modality-specific uniform deviation for the constraint gap $E_\theta - \bar{\alpha}_\theta$. For any $\theta \in \Theta$, define its exact empirical feasibility residual as

$$\xi_n(\theta) := \left[\frac{1}{n}\sum_{i=1}^n \left(E_\theta(x_i, y_i) - \bar{\alpha}_\theta(x_i)\right)\right]_+, \qquad [u]_+ := \max\{u, 0\}.$$

Both E_θ and $\bar{\alpha}_\theta$ in this definition refer to the exact budget-induced rule rather than its smooth training surrogates.

Theorem 3.1 (Training-stage population deployment bound)**.** *Under the corresponding modality-specific regularity conditions stated in Appendix E, let $\hat{\theta} \in \Theta$ be any measurable output of the training procedure. Then, with probability at least $1 - \delta$ over the training sample,*

$$\mathbb{P}\big(E_{\hat{\theta}}(X, Y) = 1\big) \le \mathbb{E}[\bar{\alpha}_{\hat{\theta}}(X)] + \Delta_n + \xi_n(\hat{\theta}). \tag{11}$$

The uniform generalization gap is

$$\Delta_n = \begin{cases} 3\sqrt{\dfrac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon + 4\rho_0 L_V \varepsilon, & \text{continuous,} \\ \sqrt{\dfrac{2\big(\log|\mathcal{F}^{\mathrm{disc}}_{\mathrm{gap}}| + \log(4/\delta)\big)}{n}}, & \text{discrete,} \end{cases}$$

Here $N_\Theta(\varepsilon)$, L_α, L_V, ρ_0, and $\mathcal{F}_{\mathrm{gap}}^{\mathrm{disc}}$ are the covering-number, regularity, boundary-density, and discrete constraint-gap quantities, respectively, defined precisely in Appendix E. If the reconstructed rule is feasible for the exact empirical constraint, then $\xi_n(\hat{\theta}) = 0$, and the result reduces to the exact empirical-feasibility bound. In general, $\xi_n(\hat{\theta})$ records the empirical mismatch of the reconstructed exact rule and therefore need not vanish for a parameter obtained by surrogate optimization.

Appendix E additionally establishes target optimality for an exact optimizer of the nonsmooth empirical problem. This idealized result is not required for post-hoc certification.

Remark 3.2 (Training-stage and post-hoc guarantees). The training-stage bound controls the population risk of the reconstructed exact rule through Δ_n and $\xi_n(\hat{\theta})$. After training, the fixed rule is certified independently on the certification set, without requiring exact empirical feasibility or exact optimization of the training objective.

4 Post-Hoc Certification of the Budget-Feasible Deployed Rule

After training, BCRC fixes the exact budget-feasible deployed rule constructed in Section 3.1 and certifies its population failure risk on an independent certification split. Certification does not modify the rule or affect its pointwise budget feasibility, and it does not require exact optimization of the training objective or a finite training-rule class. For a trained parameter $\hat{\theta}$, we use the budget-induced rule $x \mapsto \mathcal{U}_{\hat{\theta}}^\star(x)$ together with the fallback and numerical-search conventions specified above. The deployed rule is fixed before certification and satisfies the deployment budget pointwise.

The fixed rule induces the task-specific binary deployment failure loss $E_{\hat{\theta}}(x, y) \in \{0, 1\}$ defined in (7). In the continuous setting, this is the event that the realized outcome violates the robustness certificate induced by $\mathcal{U}_{\hat{\theta}}^\star(x)$. In the discrete setting, it is the event that the returned shortlist contains no acceptable action. Hence $E_{\hat{\theta}}$ measures deployment failure under the hard budget, rather than generic prediction error.

Let $R(\hat{\theta}) = \mathbb{E}\big[E_{\hat{\theta}}(X, Y)\big]$ be the population deployment risk of the fixed budget-feasible rule. Throughout this section, probabilities and expectations are understood conditional on the training data, the selected parameter, the budget, and all implementation choices made before certification.

Given an independent certification set $\mathcal{D}_{\mathrm{cert}} = \{(X_i, Y_i)\}_{i=1}^m$, we evaluate the deployment failures on the certification set using exactly the same construction as at test time, and define $\bar{E}_m(\hat{\theta}) = \frac{1}{m}\sum_{i=1}^m E_{\hat{\theta}}(X_i, Y_i)$.

Proposition 4.1 (High-probability deployment-risk certificate)**.** *Assume the certification samples are i.i.d. from the deployment distribution after the deployed rule has been fixed. Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$R(\hat{\theta}) \le \bar{E}_m(\hat{\theta}) + \sqrt{\frac{1}{2m}\log\frac{1}{\delta}}. \tag{12}$$

This yields a distribution-free finite-sample certificate for the exact fixed deployed rule; its validity does not require exact optimization of the training objective. For the binary deployment failures used in our experiments, Appendix F.4 also reports the tighter one-sided Clopper–Pearson bound. An inverse-conformal interpretation of the corresponding add-one risk estimate is given in Appendix F.

5 Experiments

We compare BCRC with task-specific baseline methods. In continuous robust control, we consider three decision-aware baselines: (i) conditional robust optimization (CRO) (Sun et al., 2023); (ii) conformal robustness control (CRC) (Hu et al., 2026); and (iii) an end-to-end conditional robust optimization method (E2E) (Chenreddy & Delage, 2024; Yeh et al., 2025). In discrete prediction tasks, we compare against APS (Romano et al., 2020), RAPS (Angelopoulos et al., 2021), and SAPS (Huang et al., 2024). We additionally use Direct Top-K as a hard-budget reference in our matched-budget evaluation, which directly returns the top K candidates.

We report the following evaluation metrics. Let $\mathcal{U}_i$ denote the deployed prediction set or shortlist for x_i. (i) Failure Risk $= n^{-1}\sum_{i=1}^n E_{\hat{\theta}}(x_i, y_i)$. (ii) For a budget function $\mathcal{C}(\cdot, \cdot)$ and threshold T, the

budget violation rate is $\mathrm{BVR} = n^{-1}\sum_{i=1}^{n}\mathbf{1}\{\mathcal{C}(\mathcal{U}_i, x_i) > T\}$. (iii) For continuous robust-control tasks, Avg. Certificate is $n^{-1}\sum_{i=1}^{n} r_i$, and Decision Loss is $n^{-1}\sum_{i=1}^{n}\phi(y_i, z_i)$. (iv) In finite-candidate tasks, Size is $n^{-1}\sum_{i=1}^{n}|\mathcal{U}_i|$. (v) Risk Upper Bound denotes the post-hoc high-probability certificate $\widehat{R}_\delta(\hat{\theta})$ from Proposition 4.1. The post-hoc risk certificates are computed on an independent certification split after the deployed rule is fixed, while all empirical metrics are reported on a separate held-out test split. The corresponding risk upper bounds across budgets are reported in Appendix G.2.

As our setting requires every deployed output to satisfy the pointwise hard budget, we use budget-feasible baselines as the primary comparison. In discrete tasks, APS, RAPS, and SAPS are truncated and re-certified, together with Direct Top-K as a matched-budget reference. The original forward baselines are reported separately to show the risk–feasibility trade-off without enforcing the budget.

5.1 Continuous Robust Control under Hard Budgets

We first evaluate BCRC on a synthetic portfolio-optimization task. We set $\mathcal{Y} = \mathbb{R}^2$, $\mathcal{Z} = \{z \in [0,1]^2 : \|z\|_1 = 1\}$, and use the loss $\phi(y, z) = -y^\top z$. The labeled and test data are generated by

$$Y_1 = -5X_1 - 2X_2^2 - e_1, \qquad Y_2 = -3X_1^2 - X_2 - e_2,$$

where $X \sim N((1,1)^\top, 2.25I_2)$, $e \sim N(0, I_2)$, and e is independent of X. We use ellipsoidal uncertainty sets in this experiment. The deployment cost is the risk certificate $r(\mathcal{U}, x)$.

Table 1: Continuous robust-control results across hard deployment budgets. The upper block reports BVR for the original forward baselines and BCRC; the lower block reports BCRC's average certificate, empirical failure risk, and post-hoc risk upper bound.

Metric	Method	$T = 5$	$T = 8$	$T = 10$	$T = 12$
BVR ↓	CRC	0.576 ± 0.012	0.425 ± 0.015	0.352 ± 0.017	0.290 ± 0.023
	CRO	0.620 ± 0.006	0.447 ± 0.015	0.376 ± 0.012	0.310 ± 0.018
	E2E	0.821 ± 0.053	0.530 ± 0.031	0.449 ± 0.019	0.386 ± 0.022
	BCRC	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$
BCRC	Avg. Certificate	4.883 ± 0.077	7.060 ± 0.376	8.978 ± 0.602	10.417 ± 0.519
	Failure risk ↓	0.489 ± 0.010	0.375 ± 0.013	0.308 ± 0.016	0.249 ± 0.020
	Risk Upper Bound ↓	0.521 ± 0.012	0.399 ± 0.008	0.336 ± 0.008	0.280 ± 0.007

Table 1 shows that the original forward CRC, CRO, and E2E baselines have mean BVRs ranging from 29.0% to 82.1%, depending on method and budget. We therefore compare the methods under the same pointwise hard budget below. Full method-level results and the real-world stock experiment are reported in Appendix G.2. Under the same pointwise hard budget, BCRC achieves lower mean failure risk than all three truncated baselines at each of the four tested budgets.

Table 2: Matched hard-budget failure risk in the continuous synthetic experiment. Values are mean ± standard deviation across the evaluation splits; lower is better.

T	BCRC	CRC + truncation	CRO + truncation	E2E + truncation
5	$\mathbf{0.489 \pm 0.010}$	0.515 ± 0.013	0.509 ± 0.012	0.508 ± 0.007
8	$\mathbf{0.375 \pm 0.013}$	0.409 ± 0.013	0.393 ± 0.013	0.389 ± 0.011
10	$\mathbf{0.308 \pm 0.016}$	0.347 ± 0.018	0.325 ± 0.020	0.324 ± 0.016
12	$\mathbf{0.249 \pm 0.020}$	0.295 ± 0.018	0.268 ± 0.021	0.265 ± 0.022

Aggregated across the four tested budgets, BCRC reduces failure risk relative to truncated CRC, CRO, and E2E by 3.62, 1.86, and 1.62 percentage points, respectively. The corresponding pairwise 95% hierarchical paired-bootstrap confidence intervals exclude zero, with two-sided $p < 10^{-4}$.

5.2 Image Classification under Hard Budgets

We evaluate BCRC on ImageNet (Deng et al., 2009) using nine pretrained classifiers (He et al., 2016; Xie et al., 2017; Huang et al., 2017; Simonyan & Zisserman, 2015; Szegedy et al., 2016; Ma et al.,

2018) under the pointwise cardinality constraint $|\mathcal{U}_i| \leq T$. BCRC maintains zero BVR for every tested backbone and budget, whereas the original forward APS, RAPS, and SAPS rules exhibit budget violations. The full forward results and risk–BVR trade-offs are reported in Appendix G.3.2.

For failure-risk comparisons under the same pointwise hard budget, we use Direct Top-K, hard-truncate APS, RAPS, and SAPS, and independently re-certify the resulting rules.

Table 3: Matched-budget image-classification results under the common pointwise constraint $|\mathcal{U}_i| \leq 2$. All deployed rules satisfy the same per-input upper bound; realized average set sizes may differ across methods.

Model	Method	Size	Failure risk	Model	Method	Size	Failure risk
ResNet18	Direct Top-K	2.000 ± 0.000	0.199 ± 0.005	DenseNet161	Direct Top-K	2.000 ± 0.000	0.134 ± 0.003
	Trunc. APS	1.571 ± 0.005	0.214 ± 0.006		Trunc. APS	1.463 ± 0.008	0.152 ± 0.004
	Trunc. RAPS	1.732 ± 0.076	0.204 ± 0.006		Trunc. RAPS	1.630 ± 0.041	0.141 ± 0.004
	Trunc. SAPS	1.785 ± 0.147	0.202 ± 0.007		Trunc. SAPS	1.525 ± 0.026	0.143 ± 0.004
	BCRC	2.000 ± 0.000	$\mathbf{0.197 \pm 0.004}$		BCRC	2.000 ± 0.000	$\mathbf{0.128 \pm 0.004}$
ResNet50	Direct Top-K	2.000 ± 0.000	0.142 ± 0.003	VGG16	Direct Top-K	2.000 ± 0.000	$\mathbf{0.180 \pm 0.002}$
	Trunc. APS	1.469 ± 0.007	0.161 ± 0.002		Trunc. APS	1.545 ± 0.009	0.195 ± 0.003
	Trunc. RAPS	1.601 ± 0.025	0.150 ± 0.003		Trunc. RAPS	1.755 ± 0.141	0.183 ± 0.004
	Trunc. SAPS	1.620 ± 0.078	0.149 ± 0.003		Trunc. SAPS	1.706 ± 0.165	0.184 ± 0.004
	BCRC	2.000 ± 0.000	$\mathbf{0.139 \pm 0.003}$		BCRC	2.000 ± 0.000	0.181 ± 0.002
ResNet101	Direct Top-K	2.000 ± 0.000	0.131 ± 0.003	Inception-v3	Direct Top-K	2.000 ± 0.000	0.199 ± 0.003
	Trunc. APS	1.450 ± 0.005	0.150 ± 0.004		Trunc. APS	1.503 ± 0.005	0.221 ± 0.003
	Trunc. RAPS	1.583 ± 0.075	0.140 ± 0.005		Trunc. RAPS	1.767 ± 0.180	0.205 ± 0.005
	Trunc. SAPS	1.525 ± 0.055	0.141 ± 0.005		Trunc. SAPS	1.754 ± 0.192	0.205 ± 0.006
	BCRC	2.000 ± 0.000	$\mathbf{0.126 \pm 0.003}$		BCRC	2.000 ± 0.000	$\mathbf{0.192 \pm 0.004}$
ResNet152	Direct Top-K	2.000 ± 0.000	0.122 ± 0.002	ShuffleNet-V2	Direct Top-K	2.000 ± 0.000	0.204 ± 0.004
	Trunc. APS	1.431 ± 0.006	0.141 ± 0.001		Trunc. APS	1.543 ± 0.009	0.222 ± 0.005
	Trunc. RAPS	1.508 ± 0.053	0.134 ± 0.005		Trunc. RAPS	1.726 ± 0.156	0.211 ± 0.006
	Trunc. SAPS	1.504 ± 0.057	0.132 ± 0.003		Trunc. SAPS	1.966 ± 0.054	0.204 ± 0.004
	BCRC	2.000 ± 0.000	$\mathbf{0.119 \pm 0.003}$		BCRC	2.000 ± 0.000	$\mathbf{0.200 \pm 0.003}$
ResNeXt101	Direct Top-K	2.000 ± 0.000	0.116 ± 0.002				
	Trunc. APS	1.399 ± 0.009	0.139 ± 0.002				
	Trunc. RAPS	1.475 ± 0.044	0.131 ± 0.004				
	Trunc. SAPS	1.455 ± 0.017	0.131 ± 0.003				
	BCRC	2.000 ± 0.000	$\mathbf{0.112 \pm 0.003}$				

At $T = 2$, BCRC reduces mean failure risk relative to Direct Top-K, truncated APS, RAPS, and SAPS by 0.36, 2.24, 1.15, and 1.08 percentage points, respectively. The corresponding pairwise 95% hierarchical paired-bootstrap confidence intervals exclude zero ($p = 3.0 \times 10^{-4}$ for Direct Top-K and $p < 10^{-4}$ for the truncated conformal baselines).

Across $T \in \{1, 2, 3, 5, 10\}$, BCRC has lower mean failure risk than Direct Top-K in $42/45$ backbone–budget settings, with an overall paired reduction of 0.364 percentage points ($p < 10^{-4}$). BCRC also has lower mean failure risk than each of truncated APS, RAPS, and SAPS in $44/45$ settings. The corresponding confidence interval and full matched-budget results are reported in Appendix G.3.2.

5.3 LLM Finite-Candidate Shortlisting

We finally evaluate BCRC on MMLU-Pro (Wang et al., 2024), using Qwen-7B, Qwen2.5-14B, and Llama3.1-8B as base models. MMLU-Pro provides up to ten candidate answers for each question, so the deployment task is to return a shortlist $\mathcal{U}_i$ of answer options under budget $|\mathcal{U}_i| \leq T$. We use the same evaluation metrics and forward conformal baselines as in image classification.

The subject-wise curves in Figure 2 reveal heterogeneous budget–risk trade-offs across MMLU-Pro subjects, with miss risk generally decreasing as T increases in all three LLMs. Larger budgets permit larger shortlists and therefore increase the likelihood that the correct answer is retained. Across domains, easier subjects reach low failure risk with relatively small shortlists, whereas harder subjects require larger budgets to achieve target reliability. Together, the curves expose the budget–risk trade-off while preserving pointwise budget feasibility. For the matched-budget comparison, we use truncated and re-certified APS, RAPS, and SAPS together with Direct Top-K. The original forward-baseline results are reported in Appendix G.3.3.

At $T = 5$, BCRC reduces failure risk relative to Direct Top-K and the three truncated conformal baselines by 0.62–0.67 percentage points. The corresponding pairwise 95% hierarchical paired-

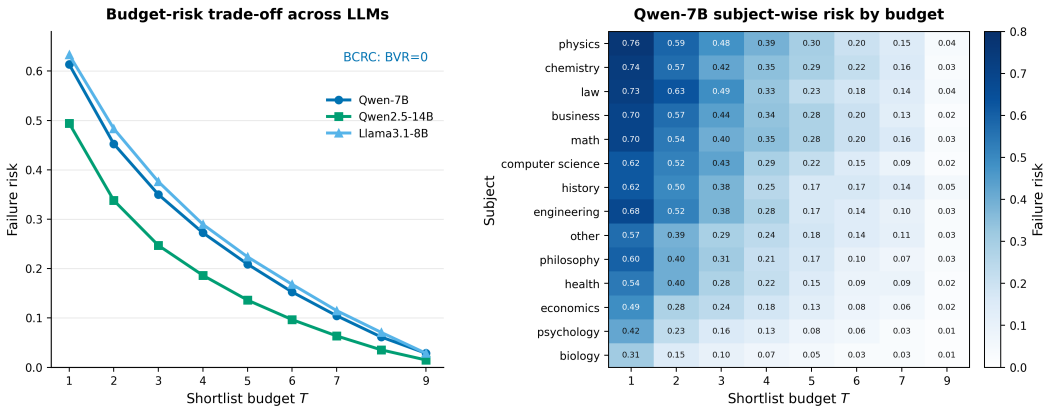


Figure 2: LLM shortlisting budget–risk trade-offs under BCRC. The left panel shows decreasing failure risk as T increases, with zero BVR across the tested LLMs. The right panel shows subject-wise miss-risk curves for Qwen-7B on MMLU-Pro, illustrating heterogeneous risk levels and decreasing miss risk as the budget increases.

Table 4: LLM shortlisting results under the common pointwise constraint $|\mathcal{U}_i| \leq 5$. BCRC achieves the lowest mean failure risk on base models. Values are mean $\pm$ standard deviation over five seeds.

Method	Qwen-7B		Qwen2.5-14B		Llama3.1-8B	
	Failure risk ↓	Size	Failure risk ↓	Size	Failure risk ↓	Size
Direct Top-K	0.214 ($\pm$0.005)	4.948 ($\pm$0.003)	0.136 ($\pm$0.007)	4.948 ($\pm$0.003)	0.225 ($\pm$0.006)	4.948 ($\pm$0.003)
Trunc. APS	0.215 ($\pm$0.006)	4.940 ($\pm$0.002)	0.136 ($\pm$0.007)	4.943 ($\pm$0.005)	0.226 ($\pm$0.005)	4.944 ($\pm$0.003)
Trunc. RAPS	0.214 ($\pm$0.005)	4.948 ($\pm$0.003)	0.136 ($\pm$0.007)	4.948 ($\pm$0.003)	0.225 ($\pm$0.006)	4.948 ($\pm$0.003)
Trunc. SAPS	0.214 ($\pm$0.005)	4.948 ($\pm$0.003)	0.136 ($\pm$0.007)	4.948 ($\pm$0.003)	0.226 ($\pm$0.005)	4.916 ($\pm$0.028)
BCRC	**0.202** (**$\pm$0.010**)	4.948 ($\pm$0.003)	**0.131** (**$\pm$0.009**)	4.948 ($\pm$0.003)	**0.223** (**$\pm$0.007**)	4.948 ($\pm$0.003)

bootstrap confidence intervals exclude zero, with two-sided $p = 0.025$–0.029. Across the three LLMs and five evaluated budgets, BCRC has lower mean failure risk than Top-K in $12/15$ model–budget settings overall. For example, the failure risk for Qwen-7B decreases from 0.214 to 0.202, with corresponding reductions for Qwen2.5-14B and Llama3.1-8B. More comprehensive results across budgets and additional diagnostics are reported in Appendix G.3.3.

6 Conclusion

We introduced Budget-First Conformal Robustness Control (BCRC), a framework for decision-making under pointwise hard deployment budgets. BCRC reverses the usual risk-first conformal order by constructing a budget-feasible deployed rule before certifying the task-specific failure risk incurred by that fixed rule. Across continuous robust control, image classification, and LLM shortlisting, BCRC satisfies the pointwise hard budget by construction and generally achieves lower failure risk than truncated forward baselines under the same budget. These results highlight the value of treating the deployment budget as a primary constraint rather than a secondary consequence of risk calibration.

AI USE STATEMENT

In this work, Qwen-7B, Qwen2.5-14B, and Llama3.1-8B were used as fixed pretrained base models solely for the MMLU-Pro finite-candidate shortlisting evaluation described in Section 5.3; they are not components of the proposed BCRC method. AI-assisted tools were used for text editing and formatting. The authors verified all AI-assisted text and model-produced experimental outputs and take responsibility for the final content of this work.

ETHICS STATEMENT

This work uses synthetic data, existing public datasets, and existing pretrained models for evaluation. It does not involve human-subject research or crowdsourcing, and it does not release new datasets or pretrained models. The principal deployment concern is that satisfying a hard budget should not be interpreted as a safety guarantee: overly small budgets may exclude plausible or useful outcomes. These limitations, together with the need for domain-specific oversight, monitoring, and recalibration under distribution shift, are discussed in Appendix B. The authors have read and adhere to the ICLR Code of Ethics and take responsibility for the ethical conduct of this work.

REPRODUCIBILITY STATEMENT

The main text specifies the exact budget-feasible deployment rules, training objectives, post-hoc certification procedure, evaluation metrics, and budget settings. Appendix E and Appendix F provide the assumptions and complete proofs for the theoretical results. Appendix D describes the surrogate relaxations, while Appendix G reports the datasets and base models, data-split and calibration protocols, hyperparameters, optimization details, and compute resources needed to reproduce the experiments.

REFERENCES

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.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025. doi: 10.1214/24-AOAS1998.

Anastasios Nikolas Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I. Jordan. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*, 2021.

Yu Bai, Song Mei, Huan Wang, Yingbo Zhou, and Caiming Xiong. Efficient and differentiable conformal prediction with general function classes. In *International Conference on Learning Representations*, 2022.

Alexander A. Balinsky and Alexander David Balinsky. Enhancing conformal prediction using e-test statistics. In *Proceedings of The 13th Symposium on Conformal and Probabilistic Prediction with Applications*, volume 230 of *Proceedings of Machine Learning Research*, pp. 65–72. PMLR, 2024.

Stephen Bates, Anastasios N. Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):43:1–43:34, 2021. doi: 10.1145/3478535.

Leonard Berrada, Andrew Zisserman, and M. Pawan Kumar. Smooth loss functions for deep Top-k classification. In *International Conference on Learning Representations*, 2018.

Dimitris Bertsimas, David B. Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011. doi: 10.1137/080734510.

Abhilash Reddy Chenreddy and Erick Delage. End-to-end conditional robust optimization. *arXiv preprint arXiv:2403.04670*, 2024.

Abhilash Reddy Chenreddy, Nymisha Bandi, and Erick Delage. Data-driven conditional robust optimization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 9525–9537, 2022.

Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, and Titouan Lorieul. Set-valued classification: Overview via a unified framework. *arXiv preprint arXiv:2102.12318*, 2021.

Corinna Cortes, Anqi Mao, Christopher Mohri, Mehryar Mohri, and Yutao Zhong. Cardinality-aware set prediction and top-k classification. In *Advances in Neural Information Processing Systems*, 2024.

Jesse C. Cresswell, Yi Sui, Bhargava Kumar, and Noël Vouitsis. Conformal prediction sets improve human decision making. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9439–9457. PMLR, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Guneet S. Dhillon, George Deligiannidis, and Tom Rainforth. On the expected size of conformal prediction sets. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1549–1557. PMLR, 2024.

Etienne Gauthier, Francis Bach, and Michael I. Jordan. Backward conformal prediction. In *Advances in Neural Information Processing Systems*, 2025a.

Etienne Gauthier, Francis Bach, and Michael I. Jordan. E-values expand the scope of conformal prediction. *arXiv preprint arXiv:2503.13050*, 2025b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Yang Hu, Jieren Tan, Changliang Zou, Yajie Bao, and Haojie Ren. Conformal robustness control: A new strategy for robust decision. In *International Conference on Learning Representations*, 2026.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

Jianguo Huang, Huajun Xi, Linjun Zhang, Huaxiu Yao, Yue Qiu, and Hongxin Wei. Conformal prediction for deep classifier via label ranking. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20331–20347. PMLR, 2024.

Chancellor Johnstone and Bruce Cox. Conformal uncertainty sets for robust optimization. In *Proceedings of The Tenth Symposium on Conformal and Probabilistic Prediction and Applications*, volume 152 of *Proceedings of Machine Learning Research*, pp. 72–90. PMLR, 2021.

Shayan Kiyani, George Pappas, and Hamed Hassani. Length optimization in conformal prediction. In *Advances in Neural Information Processing Systems*, 2024.

Shayan Kiyani, George J. Pappas, Aaron Roth, and Hamed Hassani. Decision theoretic foundations for conformal prediction: Optimal uncertainty quantification for risk-averse agents. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30943–30965. PMLR, 2025.

Maksim Lapin, Matthias Hein, and Bernt Schiele. Loss functions for Top-k error: Analysis and insights. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1468–1477, 2016.

Jing Lei. Classification with confidence. *Biometrika*, 101(4):755–769, 2014. doi: 10.1093/biomet/asu038.

Bo Lin, Erick Delage, and Timothy C. Y. Chan. Conformal inverse optimization. In *Advances in Neural Information Processing Systems*, 2024.

Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision*, pp. 116–131, 2018.

Sima Noorani, Orlando Romero, Nicolo Dal Fabbro, Hamed Hassani, and George J. Pappas. Conformal risk minimization with variance reduction. In *Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence*, volume 337 of *Proceedings of Machine Learning Research*, pp. 4972–5002. PMLR, 2026.

Yash P. Patel, Sahana Rayan, and Ambuj Tewari. Conformal contextual robust optimization. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 2485–2493. PMLR, 2024.

Drew Prinster, Anqi Liu, and Suchi Saria. Jaws: Auditing predictive uncertainty under covariate shift. In *Advances in Neural Information Processing Systems*, volume 35, pp. 35907–35920, 2022.

Drew Prinster, Suchi Saria, and Anqi Liu. JAWS-x: Addressing efficiency bottlenecks of conformal prediction under standard and feedback covariate shift. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28167–28190. PMLR, 2023.

Yaniv Romano, Matteo Sesia, and Emmanuel Candès. Classification with valid and adaptive coverage. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3581–3591, 2020.

Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525):223–234, 2019.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9:371–421, 2008.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Sukrita Singh, Neeraj Sarna, Yuanyuan Li, Yang Lin, Agni Orfanoudaki, and Michael Berger. Distribution-free risk assessment of regression-based machine learning algorithms. In *Proceedings of The 13th Symposium on Conformal and Probabilistic Prediction with Applications*, volume 230 of *Proceedings of Machine Learning Research*, pp. 44–64. PMLR, 2024.

David Stutz, Krishnamurthy Dvijotham, A. Taylan Cemgil, and Arnaud Doucet. Learning optimal conformal classifiers. In *International Conference on Learning Representations*, 2022.

Chunlin Sun, Linyu Liu, and Xiaocheng Li. Predict-then-calibrate: A new perspective of robust contextual LP. In *Advances in Neural Information Processing Systems*, volume 36, pp. 17713–17741, 2023.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016.

Renukanandan Tumu, Matthew Cleaveland, Rahul Mangharam, George J. Pappas, and Lars Lindemann. Multi-modal conformal prediction regions by optimizing convex shape templates. In *Proceedings of the 6th Annual Learning for Dynamics and Control Conference*, volume 242 of *Proceedings of Machine Learning Research*, pp. 1343–1356. PMLR, 2024.

Vladimir Vovk. Conformal e-prediction. *Pattern Recognition*, 166:111674, 2025.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer-Verlag, 2005.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, 2024.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1492–1500, 2017.

Christopher Yeh, Nicolas Christianson, Alan Wu, Adam Wierman, and Yisong Yue. End-to-end conformal calibration for optimization under uncertainty. *Transactions on Machine Learning Research*, 2025.

Wenbin Zhou and Shixiang Zhu. Calibrating decision robustness via inverse conformal risk control. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Wenbin Zhou, Agni Orfanoudaki, and Shixiang Zhu. Conformalized decision risk assessment. In *International Conference on Learning Representations*, 2026.

A Related Work

Conformal prediction and risk control. Conformal prediction provides distribution-free finite-sample guarantees for set-valued prediction under exchangeability (Vovk et al., 2005; Shafer & Vovk, 2008; Angelopoulos & Bates, 2023). Classical and adaptive conformal classification methods construct prediction sets with valid marginal coverage, including APS, RAPS, and SAPS for modern deep classifiers (Romano et al., 2020; Angelopoulos et al., 2021; Huang et al., 2024). Risk-controlling prediction sets and conformal risk control extend conformal guarantees from marginal coverage to general bounded or monotone losses (Bates et al., 2021; Angelopoulos et al., 2024), while Learn-then-Test gives a related multiple-testing view of calibrating predictive algorithms to satisfy finite-sample risk constraints (Angelopoulos et al., 2025). These works provide the statistical foundation for BCRC, but they are primarily target-first: a coverage or risk level is specified in advance, and the resulting set size, volume, or deployment cost is induced afterward. The cardinality-aware objective of Cortes et al. (2024) balances classification error and set-size cost in expectation, while also studying fixed-k prediction; BCRC instead takes the pointwise deployment cap as an input constraint.

Inverse and post-hoc conformal certification. BCRC is also related to methods that certify a predictive or decision object after it has been chosen. Work on conformal e-values and post-hoc validity provides tools for data-dependent calibration levels and selected predictive objects (Vovk, 2025; Balinsky & Balinsky, 2024; Gauthier et al., 2025b). Related inverse uses of conformal prediction estimate the error or risk of a fixed predictor or prediction set (Prinster et al., 2022; 2023; Singh et al., 2024). In robust decision-making, inverse conformal risk control estimates risk or regret across fixed robustness levels for robust predict-then-optimize policies (Zhou & Zhu, 2026). Conformal inverse optimization and conformalized decision risk assessment certify risks for fixed candidate decisions using optimization structure (Lin et al., 2024; Zhou et al., 2026). These works share the post-hoc view that certification should apply to the object actually under consideration. Our use of this view is tied to deployment: once a budget-feasible set-valued rule has been learned and fixed, the certificate is attached to that exact rule.

B Limitations

BCRC treats the deployment budget as a hard constraint and therefore inherits the trade-off imposed by that constraint. When the budget is too small, or when the underlying score model ranks candidates poorly, a budget-feasible set may exclude important outcomes and lead to higher deployment risk. The method also assumes that the budget function is known and evaluable at deployment time; budgets that depend on changing system load, human availability, or other external factors are not modeled here. Finally, the post-hoc certificate relies on an independent certification split from the deployment distribution. Distribution shift or adaptive reuse of certification data may weaken the reported guarantee. The discrete formulation selects from a predefined finite candidate pool; token or sequence-length budgets in open-ended generation are outside the present setting.

The main practical motivation of BCRC is to make uncertainty-aware systems usable when outputs must fit a fixed operational budget, such as limited shortlist length, review capacity, or robust-decision cost. This can improve deployability by preventing uncertainty sets from becoming too large for the downstream interface or decision pipeline. At the same time, strict feasibility should not be mistaken for safety: a small budget can force the system to discard plausible or useful candidates. In high-stakes domains, the budget should therefore be chosen with domain-specific risk requirements, human oversight, and monitoring after deployment. The reported certificate describes the fixed deployed rule under the deployment distribution represented by the certification split, and should be recomputed when the deployment environment changes.

C Decision-Theoretic Justification

This section formalizes the prediction-set representation used in Section 3.1. The result concerns the continuous decision-and-certificate setting in which the deployment cost depends only on the loss certificate.

Proposition C.1 (Prediction-set representation)**.** *Suppose that $\mathcal{C}(z, r, x) = c(r, x)$, where $c(r, x)$ is nondecreasing in r. For any measurable pair (z, r), assume that $\mathcal{U}_{z,r}(x) = \{y \in \mathcal{Y}(x) :$*

$\phi(y, z(x)) \le r(x)\}$ *is nonempty and that* (5) *admits a measurable solution. Then, for every* x, $r_{\mathcal{U}_{z,r}}(x) \le r(x)$ *and* $\mathcal{C}(z_{\mathcal{U}_{z,r}}(x), r_{\mathcal{U}_{z,r}}(x), x) \le \mathcal{C}(z(x), r(x), x)$. *Moreover,*

$$\mathbb{P}\big\{\phi\big(Y, z_{\mathcal{U}_{z,r}}(X)\big) > r_{\mathcal{U}_{z,r}}(X)\big\} \le \mathbb{P}\{\phi(Y, z(X)) > r(X)\}\,. \tag{13}$$

Consequently, the unrestricted prediction-set and decision-and-certificate formulations have the same optimal value under the hard budget.

Proof. Fix x. Since $z(x)$ is feasible for the robust problem induced by $\mathcal{U}_{z,r}(x)$,

$$r_{\mathcal{U}_{z,r}}(x) \le \max_{y \in \mathcal{U}_{z,r}(x)} \phi(y, z(x)) \le r(x).$$

The pointwise cost inequality follows from the monotonicity of c. Also, $\phi(Y, z(X)) \le r(X)$ is equivalent to $Y \in \mathcal{U}_{z,r}(X)$. On this event, the definition of the robust certificate gives $\phi(Y, z_{\mathcal{U}_{z,r}}(X)) \le r_{\mathcal{U}_{z,r}}(X)$, which proves (13). Conversely, every uncertainty-set rule induces a decision and certificate through (5); hence the two unrestricted formulations have the same optimal value. □

Proposition C.1 does not remove the modeling restriction imposed by the parameterized family $\{\mathcal{U}_{\theta,\alpha}\}$, and it does not apply directly to finite-candidate shortlisting, where the shortlist itself is deployed and its cost is its cardinality.

C.1 Relation between the Budget-Induced Target and Deployment Risk

For this subsection only, define

$$R(\theta) := \mathbb{P}(E_\theta(X, Y) = 1), \qquad A(\theta) := \mathbb{E}[\bar{\alpha}_\theta(X)],$$

and write $S(\theta) := A(\theta) - R(\theta)$. Let

$$\mathcal{F} := \{\theta \in \Theta : R(\theta) \le A(\theta)\}$$

denote the population-feasible parameter set in (8). Assume that $\mathcal{F}$ is nonempty and that the following minima are attained. Let

$$\theta_A \in \arg\min_{\theta \in \mathcal{F}} A(\theta), \qquad \theta_R \in \arg\min_{\theta \in \mathcal{F}} R(\theta).$$

Proposition C.2 (Target-risk relation)**.**

$$0 \le R(\theta_A) - R(\theta_R) \le S(\theta_R).$$

Proof. The optimality of θ_R gives $R(\theta_R) \le R(\theta_A)$. Population feasibility and the optimality of θ_A give

$$R(\theta_A) \le A(\theta_A) \le A(\theta_R) = R(\theta_R) + S(\theta_R).$$

Combining the two inequalities proves the result. □

Corollary C.3. *If* $S(\theta_R) \le \varepsilon_{\mathrm{tar}}$*, then*

$$R(\theta_A) - R(\theta_R) \le \varepsilon_{\mathrm{tar}}.$$

In particular, the condition holds if $\sup_{\theta \in \mathcal{F}} S(\theta) \le \varepsilon_{\mathrm{tar}}$.

In the continuous decision-and-certificate setting of Equation (4), define

$$\alpha^\star_\Theta(T) := \inf_{\theta \in \Theta} R(\theta), \qquad \alpha^\star_{\mathcal{F}}(T) := \inf_{\theta \in \mathcal{F}} R(\theta) = R(\theta_R).$$

The unrestricted oracle, parameterized family, and population-feasible family satisfy

$$\alpha^\star(T) \le \alpha^\star_\Theta(T) \le \alpha^\star_{\mathcal{F}}(T).$$

Proposition C.2 therefore gives

$$R(\theta_A) - \alpha^\star(T) \le S(\theta_R) + \big[\alpha^\star_{\mathcal{F}}(T) - \alpha^\star_\Theta(T)\big] + \big[\alpha^\star_\Theta(T) - \alpha^\star(T)\big].$$

The three terms are the comparator target slack, the population-constraint restriction gap, and the family approximation gap, respectively.

Single-label cardinality specialization. In the discrete single-label setting, let $B_\theta(x) := \mathcal{Y}(x) \setminus \mathcal{U}^\star_\theta(x)$. By Equation (16),

$$A(\theta) = \mathbb{E}_X[\Pi_\theta(B_\theta(X) \mid X)], \qquad R(\theta) = \mathbb{E}_X\left[P_{Y|X}(B_\theta(X) \mid X)\right].$$

Hence

$$|A(\theta) - R(\theta)| \le \mathbb{E}_X \operatorname{TV}\big(P_{Y|X}(\cdot \mid X), \Pi_\theta(\cdot \mid X)\big).$$

For $\theta_R \in \mathcal{F}$, this yields

$$0 \le S(\theta_R) \le \mathbb{E}_X \operatorname{TV}\big(P_{Y|X}(\cdot \mid X), \Pi_{\theta_R}(\cdot \mid X)\big).$$

Thus, in the single-label cardinality setting, $A(\theta)$ is the learned conditional model's counterpart of the miss probability for the exact deployed Top-T set.

The value of $\bar{\alpha}_\theta$ depends on the parameterization of the nested family. It is therefore a training criterion for the specified family rather than an intrinsic quantity of the deployed rule. Deployment risk and post-hoc certification are evaluated on the exact rule $\mathcal{U}^\star_\theta$ itself.

D Candidate-Set Construction and Surrogate Relaxations

This section specifies the modality-specific candidate families and the surrogates used in (10). The surrogates are used only during training; deployment and certification use the exact rule from (6).

Score-induced candidate family. Let $s_\theta(x, u)$ be a score for $u \in \mathcal{Y}(x)$, with larger values indicating greater model-assigned plausibility. It induces a probability measure $\Pi_\theta(\cdot \mid x)$. In the continuous setting, for a reference measure μ_x, its density is $\pi_\theta(u \mid x) = \exp(s_\theta(x, u))/Z_\theta(x)$, where $Z_\theta(x) = \int_{\mathcal{Y}(x)} \exp(s_\theta(x, u))\, d\mu_x(u) < \infty$. In the discrete setting, $\pi_\theta(a \mid x) = \exp(s_\theta(x, a))/\sum_{u \in \mathcal{Y}(x)} \exp(s_\theta(x, u))$.

For continuous outcomes, define

$$L_{\theta,\alpha}(x) := \sup\{\ell \in \mathbb{R} : \Pi_\theta(\{u \in \mathcal{Y}(x) : s_\theta(x, u) \ge \ell\} \mid x) \ge 1 - \alpha\}. \tag{14}$$

The candidate set is $\mathcal{U}_{\theta,\alpha}(x) = \{u \in \mathcal{Y}(x) : s_\theta(x, u) \ge L_{\theta,\alpha}(x)\}$. If $\alpha_1 \le \alpha_2$, then $L_{\theta,\alpha_1}(x) \le L_{\theta,\alpha_2}(x)$, which gives the nesting assumed in Section 3.1. Gaussian scores yield ellipsoidal superlevel sets, as used in the continuous experiments.

For finite $\mathcal{Y}(x)$, order the candidates by decreasing score using a fixed deterministic tie-breaking rule, and write the resulting sequence as $a_{\theta,(1)}(x), \ldots, a_{\theta,(m(x))}(x)$. Define

$$\begin{aligned} k_{\theta,\alpha}(x) &:= \min\left\{k \in \{1, \ldots, m(x)\} : \sum_{j=1}^{k} \pi_\theta(a_{\theta,(j)}(x) \mid x) \ge 1 - \alpha\right\}, \\ \mathcal{U}_{\theta,\alpha}(x) &= \{a_{\theta,(1)}(x), \ldots, a_{\theta,(k_{\theta,\alpha}(x))}(x)\}. \end{aligned} \tag{15}$$

This family is also nested. Under the cardinality cost $\mathcal{C}(\mathcal{U}, x) = |\mathcal{U}|$ and an integer budget T, let $T_x = \min\{T, m(x)\}$. With the endpoint convention $\bar{\alpha}_\theta(x) = 0$ when $T_x = m(x)$, the exact deployed set and its induced level are

$$\mathcal{U}^\star_\theta(x) = \{a_{\theta,(1)}(x), \ldots, a_{\theta,(T_x)}(x)\}, \qquad \bar{\alpha}_\theta(x) = 1 - \sum_{j=1}^{T_x} \pi_\theta(a_{\theta,(j)}(x) \mid x). \tag{16}$$

For the score-induced families used here, α is fixed as a probability-mass coordinate under $\Pi_\theta(\cdot \mid x)$.

Cardinality-budget specialization. When $\mathcal{C}(\mathcal{U}, x) = |\mathcal{U}|$, the exact deployed set is

$$\mathcal{U}^\star_\theta(x) = \operatorname{Top}_{T_x}\{s_\theta(x, a) : a \in \mathcal{Y}(x)\}, \qquad T_x = \min\{T, m(x)\}.$$

Thus, cardinality-constrained shortlisting is an exact specialization of the general budget-induced construction, with hard Top-T_x deployment.

Continuous relaxation. For implementation, we work in cost coordinates $c = -y$. The quantities $\mu_\theta(x)$ and $L_\theta(x)$ below parameterize the conditional distribution of c, so that $\phi(y, z) = c^\top z$. For the continuous branch, the quantities below are training-only relaxations that are differentiable almost everywhere with respect to the trainable parameters. They are discarded after optimization. We use the fixed grid

$$a_j = 10^{-4} + (j-1)\frac{1-10^{-4}}{80}, \qquad \varrho_j = \min\{6, \sqrt{-2\log a_j}\}, \qquad j = 1, \dots, 81.$$

For a fixed input x, suppress its argument and write μ_θ and L_θ for the predicted Gaussian location and factor in cost coordinates. At each level a_j, the training-time decision is relaxed on the 301-point simplex grid $v_g = (g/300, 1 - g/300)$, $g = 0, \dots, 300$, as

$$q_{jg} = \mu_\theta^\top v_g + \varrho_j \|L_\theta^\top v_g\|_2, \qquad p_{jg} = \frac{\exp(-q_{jg}/\tau_z)}{\sum_h \exp(-q_{jh}/\tau_z)}, \qquad \widetilde{z}_j = \sum_g p_{jg} v_g,$$

where $\tau_z = 0.05$. The corresponding training certificate is

$$\widetilde{r}_j = \mu_\theta^\top \widetilde{z}_j + \varrho_j \|L_\theta^\top \widetilde{z}_j\|_2.$$

Let $\sigma(t) = 1/(1 + \exp(-t))$. The soft budget-feasibility weights are

$$f_j = \sigma\left(\frac{T - \widetilde{r}_j}{\tau_b(e)}\right), \qquad f_0 = 0, \qquad d_j = [f_j - f_{j-1}]_+, \qquad d_{\mathrm{fb}} = [1 - f_{81}]_+,$$
$$Z = \sum_j d_j + d_{\mathrm{fb}} + 10^{-8}, \qquad w_j = d_j/Z, \qquad w_{\mathrm{fb}} = d_{\mathrm{fb}}/Z.$$

They define the training-only level, decision, and certificate

$$\widetilde{\alpha}_\theta^{\mathrm{cont}} = \sum_j w_j a_j + w_{\mathrm{fb}}, \qquad \widetilde{z}_\theta = \sum_j w_j \widetilde{z}_j + w_{\mathrm{fb}}(1/2, 1/2), \qquad \widetilde{r}_\theta = \sum_j w_j \widetilde{r}_j + w_{\mathrm{fb}} T.$$

The continuous failure relaxation is

$$\widetilde{E}_\theta^{\mathrm{cont}}(x, y) = \sigma\left(\frac{\phi(y, \widetilde{z}_\theta(x)) - \widetilde{r}_\theta(x)}{\tau_f(e)}\right). \tag{17}$$

For the pre-specified 200-epoch schedule, the temperatures are

$$\tau_b(e) = 0.6 + \frac{e-1}{199}(0.12 - 0.6), \qquad \tau_f(e) = 0.4 + \frac{e-1}{199}(0.10 - 0.4).$$

ImageNet relaxation. The frozen classifier supplies a feature vector h and raw logits ℓ. We form the adjusted scores

$$h' = \mathrm{Dropout}_{0.05}(\mathrm{GELU}(W_d\,\mathrm{LayerNorm}(h) + b_d)),$$
$$d_\theta = 8\tanh\left(\frac{W_u h' + b_u}{8}\right), \qquad u = \ell + d_\theta.$$

During training, u is standardized within each row to $\widehat{u}$. The soft Top-T weights satisfy

$$w_a = \sigma\left(\frac{\widehat{u}_a - b}{\tau}\right), \qquad \sum_a w_a = T,$$

The threshold b is computed using 28 bisection steps. A custom backward pass applies implicit differentiation to $\sum_a w_a = T$, including the dependence of b on $\widehat{u}$. With $\pi_a = \mathrm{softmax}(u)_a$, the training quantities are

$$\widetilde{E} = 1 - w_y, \qquad \widetilde{\alpha} = \left[1 - \sum_a w_a \pi_a\right]_{[0,1]}.$$

The temperature τ is annealed from 1 to 0.1 over 35 epochs. Deployment uses exact hard Top-T.

MMLU-Pro relaxation. For valid option a in example i, the adjusted score is

$$u_{ia} = \ell_{ia} + c_{\text{res}} \tanh(\text{MLP}_H(f_{ia}) + b_a + \langle e_{v_i}, q_a \rangle),$$

Here v_i denotes the subject index, $e_{v_i}, q_a \in \mathbb{R}^4$ are trainable subject and option embeddings, and b_a is an option-specific bias. The vector f_{ia} contains 13 standardized option features, $H \in \{0, 16, 32\}$, and the subject–option interaction has rank 4. The case $H = 0$ denotes the linear ranker. Let

$$\pi^{\text{opt}} = \text{softmax}_{\text{valid}}(\beta_\theta u), \qquad \beta_\theta > 0.$$

When $m(x) > T$, let b_y^{wrong} be the T-th largest score among the incorrect valid options and define

$$\widetilde{E} = \sigma\left(\frac{b_y^{\text{wrong}} - u_y}{t}\right), \qquad b^{\text{gate}} = \frac{u_{(T)} + u_{(T+1)}}{2},$$
$$w_a = \sigma\left(\frac{u_a - b^{\text{gate}}}{t}\right), \qquad \widetilde{\bar{\alpha}} = 1 - \sum_a w_a \pi_a^{\text{opt}}.$$

If $m(x) \le T$, both quantities are zero. Deployment uses exact hard Top-$\min\{T, m(x)\}$.

E Proofs for Training-Stage Generalization

This appendix proves Theorems 3.1 and E.1. Throughout this appendix, the training sample is assumed to be i.i.d. The first result applies to any measurable training output through its exact empirical feasibility residual; the second concerns the exact empirical problem in (9).

E.1 Exact-optimizer target optimality

For the population counterpart, define the Δ_n-margin population optimizer

$$\theta^\star_{\Delta_n} \in \arg\min_{\theta \in \Theta} \mathbb{E}[\bar{\alpha}_\theta(X)] \quad \text{s.t.} \quad \mathbb{P}(E_\theta(X, Y) = 1) \le \mathbb{E}[\bar{\alpha}_\theta(X)] - \Delta_n. \tag{18}$$

The margin keeps the population optimizer empirically feasible on the uniform-deviation event.

Theorem E.1 (Budget-induced target optimality gap)**.** *Under the same modality-specific conditions as Theorem 3.1, let $\hat{\theta}$ be a measurable optimal solution of the exact empirical problem* (9)*, and assume that the margin population optimizer $\theta^\star_{\Delta_n}$ in* (18) *exists. Then, with probability at least $1 - \delta$ over the training sample,*

$$\mathbb{E}[\bar{\alpha}_{\hat{\theta}}(X)] - \mathbb{E}[\bar{\alpha}_{\theta^\star_{\Delta_n}}(X)] \le 2\Gamma_n. \tag{19}$$

This result concerns optimality of the budget-induced target within the parameterized family and assumes an exact optimizer of the nonsmooth empirical problem. It is not a guarantee that the surrogate training procedure attains this optimizer, and it is not required for the independent post-hoc certificate.

For each $\theta \in \Theta$, write $a_\theta(x) := \bar{\alpha}_\theta(x)$, $g_\theta(x, y) := E_\theta(x, y) - a_\theta(x)$. By construction, $a_\theta(x) \in [0, 1]$, $E_\theta(x, y) \in \{0, 1\}$, and therefore $g_\theta(x, y) \in [-1, 1]$. Let P denote the population distribution, P_X its X-marginal, and P_n the empirical measure over the training sample. For any measurable function f, write $Pf := \mathbb{E}[f(X, Y)]$, $P_n f := \frac{1}{n}\sum_{i=1}^n f(x_i, y_i)$. For functions of x only, $Pf = \mathbb{E}[f(X)]$ and $P_n f = n^{-1}\sum_i f(x_i)$. With this notation, the empirical constraint in (9) is simply $P_n g_\theta \le 0$, and the objective is $P_n a_\theta$.

E.2 Regularity assumptions

The following assumptions are used to bridge the training-stage generalization gaps in the continuous robust-control setting and the discrete setting.

Assumption E.2 (Continuous deployed-rule Lipschitz regularity)**.** *In the continuous robust-control setting, assume that there exist constants $L_\alpha, L_V > 0$ and a measurable set $\mathcal{X}_0 \subseteq \mathcal{X}$ with $P_X(\mathcal{X}_0) = 1$ such that, for all $x \in \mathcal{X}_0$, the budget-induced level $a_\theta(x) = \bar{\alpha}_\theta(x)$ satisfies $|a_\theta(x) - a_{\theta'}(x)| \le L_\alpha \|\theta - \theta'\|_\infty$, and the certificate margin $V_\theta(x, y) = \phi(y, z_\theta(x)) - r_\theta(x)$ satisfies $|V_\theta(x, y) - V_{\theta'}(x, y)| \le L_V \|\theta - \theta'\|_\infty$ for all $y \in \mathcal{Y}(x)$ and all $\theta, \theta' \in \Theta$.*

This assumption is a condition on the exact deployed rule, including its fallback branch. Discontinuous fallback switches or changes in the selected robust optimizer may violate this regularity. The training-stage bound is therefore conditional on the stated assumption, whereas the independent post-hoc certificate does not require it.

Assumption E.3 (Continuous boundary density)**.** *In the continuous robust-control setting, for every $\theta \in \Theta$, the random variable $V_\theta(X, Y)$ admits a density f_θ with respect to the Lebesgue measure. Moreover, there exists a constant $\rho_0 > 0$ such that*

$$\sup_{\theta\in\Theta}\sup_{t\in\mathbb{R}} f_\theta(t) \le \rho_0.$$

Assumption E.4 (Continuous parameter-space complexity)**.** *In the continuous robust-control setting, for every $\varepsilon > 0$ considered below, the parameter space $(\Theta, \|\cdot\|_\infty)$ admits a finite ε-cover. Denote its covering number by*

$$N_\Theta(\varepsilon) := N(\Theta, \|\cdot\|_\infty, \varepsilon) < \infty.$$

Here, $N_\Theta(\varepsilon)$ is the minimum cardinality of an ε-cover of $(\Theta, \|\cdot\|_\infty)$.

Assumption E.5 (Discrete induced-class complexity)**.** *In the discrete finite-candidate setting, define the induced constraint-gap class $\mathcal{F}_{\mathrm{gap}}^{\mathrm{disc}} = \{(x, y) \mapsto E_\theta(x, y) - \bar{\alpha}_\theta(x) : \theta \in \Theta\}$ and the induced budget-level class $\mathcal{F}_\alpha^{\mathrm{disc}} = \{x \mapsto \bar{\alpha}_\theta(x) : \theta \in \Theta\}$. Assume that Θ is nonempty, that every function in $\mathcal{F}_{\mathrm{gap}}^{\mathrm{disc}} \cup \mathcal{F}_\alpha^{\mathrm{disc}}$ is measurable, and that both induced classes are finite:*

$$\left|\mathcal{F}_{\mathrm{gap}}^{\mathrm{disc}}\right| < \infty, \qquad \left|\mathcal{F}_\alpha^{\mathrm{disc}}\right| < \infty.$$

The Lipschitz conditions transfer closeness in parameter space to closeness of the budget-induced level and the certificate margin. The boundary-density condition controls the amount of probability mass near the decision boundary $V_\theta = 0$, where the binary failure indicator $E_\theta(x, y) = \mathbf{1}\{V_\theta(x, y) > 0\}$ can change under small perturbations of θ.

Assumption E.5 is a finite induced-class condition, not a consequence of finite candidate sets alone. It holds, for example, when the deployed discrete rules and the corresponding budget-induced levels are selected from a finite rule library, a finite parameter grid, or another finite collection that induces only finitely many functions in the two classes. If θ ranges over an infinite or continuous parameter class, one may replace the finite-cardinality bound with a VC, covering-number, or Rademacher-complexity bound.

The finite-class specialization is not used as a numerical complexity bound for the continuously parameterized neural rankers in the ImageNet and MMLU-Pro experiments. For such models, an appropriate complexity control for the induced function class is required to instantiate the training-stage deviation term. The independent post-hoc certificate does not rely on this training-stage complexity assumption.

E.3 Modality-specific deviation terms

Under Assumptions E.2–E.4 in continuous settings, define

$$\begin{cases} \Delta_n^{\mathrm{cont}}(\varepsilon, \delta) = 3\sqrt{\dfrac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon + 4\rho_0 L_V \varepsilon, \\ \Gamma_n^{\mathrm{cont}}(\varepsilon, \delta) = \sqrt{\dfrac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon. \end{cases} \tag{20}$$

In the discrete setting, under Assumption E.5, define

$$\begin{cases} \Delta_n^{\mathrm{disc}}(\delta) = \sqrt{\dfrac{2\left\{\log\left|\mathcal{F}_{\mathrm{gap}}^{\mathrm{disc}}\right| + \log(4/\delta)\right\}}{n}}, \\ \Gamma_n^{\mathrm{disc}}(\delta) = \sqrt{\dfrac{\log|\mathcal{F}_\alpha^{\mathrm{disc}}| + \log(4/\delta)}{2n}}. \end{cases} \tag{21}$$

Throughout the remaining proofs, Δ_n and Γ_n denote the corresponding modality-specific quantities:

$$(\Delta_n, \Gamma_n) = \begin{cases} \left(\Delta_n^{\mathrm{cont}}(\varepsilon, \delta), \Gamma_n^{\mathrm{cont}}(\varepsilon, \delta)\right), & \text{continuous setting,} \\ \left(\Delta_n^{\mathrm{disc}}(\delta), \Gamma_n^{\mathrm{disc}}(\delta)\right), & \text{discrete setting.} \end{cases}$$

E.4 UNIFORM DEVIATION BOUNDS

The next lemma is the only probabilistic ingredient needed for both training theorems. It states that the empirical constraint gap and the empirical objective uniformly approximate their population counterparts. We assume that the induced function classes are separable so that the suprema appearing below are measurable. Alternatively, all probability statements may be interpreted in outer probability.

Lemma E.6 (Modality-specific uniform deviations)**.** *Assume that the training sample consists of i.i.d. observations. Under the corresponding modality-specific conditions above, with probability at least* $1-\delta$*,*

$$\sup_{\theta\in\Theta} |Pg_\theta - P_n g_\theta| \le \Delta_n, \tag{22}$$

$$\sup_{\theta\in\Theta} |Pa_\theta - P_n a_\theta| \le \Gamma_n. \tag{23}$$

Proof. We prove the two settings separately.

Continuous setting. Let $\{\theta_1,\dots,\theta_N\}$ be an ε-cover of $(\Theta, \|\cdot\|_\infty)$, where $N = N_\Theta(\varepsilon)$. For each cover point define

$$B_j(x,y) := \mathbf{1}\{|V_{\theta_j}(x,y)| \le L_V\varepsilon\}.$$

By Assumption E.3, $PB_j \le 2\rho_0 L_V\varepsilon$.

Hoeffding's inequality and union bounds over the N cover points imply that, with probability at least $1-3\delta/4$, the following three bounds hold simultaneously:

$$\max_{1\le j\le N} |Pa_{\theta_j} - P_n a_{\theta_j}| \le \sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}},$$

$$\max_{1\le j\le N} |Pg_{\theta_j} - P_n g_{\theta_j}| \le 2\sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}},$$

$$\max_{1\le j\le N} (P_n B_j - PB_j) \le \sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}}.$$

Indeed, for the first and third displays we use Hoeffding's inequality for $[0,1]$-valued variables, while for the second display we use the same inequality for $[-1,1]$-valued variables, which produces the extra factor 2. With the choice

$$t = \sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}},$$

a union bound over the N cover points gives failure probability at most $\delta/4$ for each of the three events.

On this event, fix any $\theta\in\Theta$ and choose a cover point θ_j satisfying $\|\theta-\theta_j\|_\infty \le \varepsilon$. By Assumption E.2,

$$|a_\theta(x) - a_{\theta_j}(x)| \le L_\alpha\varepsilon \quad \text{for all } x.$$

Therefore,

$$\begin{aligned} |Pa_\theta - P_n a_\theta| &\le |Pa_{\theta_j} - P_n a_{\theta_j}| + P|a_\theta - a_{\theta_j}| + P_n|a_\theta - a_{\theta_j}| \\ &\le |Pa_{\theta_j} - P_n a_{\theta_j}| + 2L_\alpha\varepsilon. \end{aligned}$$

Taking the supremum over $\theta\in\Theta$ yields

$$\sup_{\theta\in\Theta} |Pa_\theta - P_n a_\theta| \le \sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon = \Gamma_n^{\mathrm{cont}}(\varepsilon,\delta),$$

as defined in (20). It remains to control the constraint gap. By Assumption E.2, if $E_\theta(x,y) \ne E_{\theta_j}(x,y)$, then $|V_{\theta_j}(x,y)| \le L_V\varepsilon$. Hence

$$|E_\theta(x,y) - E_{\theta_j}(x,y)| \le B_j(x,y),$$

and consequently

$$|g_\theta(x,y) - g_{\theta_j}(x,y)| \le B_j(x,y) + L_\alpha\varepsilon.$$

Therefore,

$$\begin{aligned}|Pg_\theta - P_n g_\theta| &\le |Pg_{\theta_j} - P_n g_{\theta_j}| + P|g_\theta - g_{\theta_j}| + P_n|g_\theta - g_{\theta_j}| \\ &\le 2\sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon + PB_j + P_nB_j.\end{aligned}$$

Using $PB_j \le 2\rho_0 L_V \varepsilon$ and the boundary concentration bound,

$$P_nB_j \le PB_j + \sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}},$$

we obtain

$$\sup_{\theta\in\Theta}|Pg_\theta - P_n g_\theta| \le 3\sqrt{\frac{\log N_\Theta(\varepsilon) + \log(8/\delta)}{2n}} + 2L_\alpha\varepsilon + 4\rho_0 L_V\varepsilon,$$

which is (20). Since $1 - 3\delta/4 \ge 1 - \delta$, the two continuous uniform deviation bounds hold simultaneously with probability at least $1 - \delta$.

Discrete setting. In the discrete finite-candidate setting, the exact top-K rule is ranking-based and may be discontinuous in θ. We therefore control the induced function classes directly. For a fixed $g \in \mathcal{F}_{\text{gap}}^{\text{disc}}$, we have $g(X, Y) \in [-1, 1]$, so Hoeffding's inequality gives

$$\mathbb{P}(|Pg - P_n g| \ge t) \le 2\exp\left(-\frac{nt^2}{2}\right).$$

Applying a union bound over $\left|\mathcal{F}_{\text{gap}}^{\text{disc}}\right|$ functions and taking

$$t = \sqrt{\frac{2\left\{\log\left|\mathcal{F}_{\text{gap}}^{\text{disc}}\right| + \log(4/\delta)\right\}}{n}}$$

gives the constraint-gap bound (21). Similarly, for each $a \in \mathcal{F}_\alpha^{\text{disc}}$, $a(X) \in [0, 1]$, so Hoeffding's inequality gives

$$\mathbb{P}(|Pa - P_n a| \ge t) \le 2\exp(-2nt^2).$$

A union bound over $\left|\mathcal{F}_\alpha^{\text{disc}}\right|$ functions gives (21). A final union bound over the constraint-gap and objective events shows that both discrete bounds hold simultaneously with probability at least $1 - \delta$. □

E.5 Proof of Theorem 3.1

Proof. Let $\hat{\theta} \in \Theta$ be any measurable output of the training procedure. On the event in Lemma E.6,

$$\begin{aligned}Pg_{\hat\theta} &\le P_n g_{\hat\theta} + \Delta_n \\ &\le [P_n g_{\hat\theta}]_+ + \Delta_n \\ &= \xi_n(\hat\theta) + \Delta_n.\end{aligned}$$

Expanding $g_{\hat\theta} = E_{\hat\theta} - a_{\hat\theta}$, we obtain

$$\mathbb{P}(E_{\hat\theta}(X, Y) = 1) \le \mathbb{E}[\bar\alpha_{\hat\theta}(X)] + \Delta_n + \xi_n(\hat\theta).$$

Since $E_{\hat\theta}$ is a binary failure indicator,

$$\mathbb{P}(E_{\hat\theta}(X, Y) = 0) = 1 - \mathbb{P}(E_{\hat\theta}(X, Y) = 1),$$

which proves

$$\mathbb{P}(E_{\hat\theta}(X, Y) = 0) \ge 1 - \mathbb{E}[\bar\alpha_{\hat\theta}(X)] - \Delta_n - \xi_n(\hat\theta).$$

□

E.6 Proof of Theorem E.1

Proof. Let $\theta^\star_{\Delta_n}$ be the Δ_n-margin population optimizer defined in (18). By definition,

$$Pg_{\theta^\star_{\Delta_n}} = \mathbb{P}(E_{\theta^\star_{\Delta_n}}(X,Y)=1) - \mathbb{E}[\bar{\alpha}_{\theta^\star_{\Delta_n}}(X)] \le -\Delta_n.$$

On the event in Lemma E.6,

$$P_n g_{\theta^\star_{\Delta_n}} \le Pg_{\theta^\star_{\Delta_n}} + \Delta_n \le 0.$$

Thus the margin population optimizer is feasible for the empirical constraint.

Since $\hat{\theta}$ is an optimal solution of the exact empirical problem and $\theta^\star_{\Delta_n}$ is empirically feasible, empirical optimality gives

$$P_n a_{\hat{\theta}} \le P_n a_{\theta^\star_{\Delta_n}}.$$

Using the objective uniform deviation in Lemma E.6 twice, we obtain

$$Pa_{\hat{\theta}} \le P_n a_{\hat{\theta}} + \Gamma_n \le P_n a_{\theta^\star_{\Delta_n}} + \Gamma_n \le Pa_{\theta^\star_{\Delta_n}} + 2\Gamma_n.$$

Recalling that $a_\theta(x) = \bar{\alpha}_\theta(x)$, this is equivalent to

$$\mathbb{E}[\bar{\alpha}_{\hat{\theta}}(X)] - \mathbb{E}[\bar{\alpha}_{\theta^\star_{\Delta_n}}(X)] \le 2\Gamma_n.$$

This proves the theorem. □

E.7 Exact-rule training diagnostics

Table 5: Exact empirical feasibility residuals of the final BCRC checkpoints, evaluated on the fitting split after reconstructing the exact deployment rule.

Setting	Checkpoints	Median ξ_n	Mean ξ_n
Continuous	20	0.0058	0.0079
ImageNet	225	0.0074	0.0175
MMLU-Pro	75	0.0000	0.0224

The residuals are small on average and in median across the final checkpoints. The training-stage bound nevertheless retains the explicit residual term rather than assuming exact empirical feasibility. All values use the exact paper-defined rule on the fitting split, not the smooth training surrogates. Pointwise budget feasibility and the independent post-hoc certificate do not require this residual to vanish.

F Proofs for Post-Hoc Risk Certification

This appendix proves the post-hoc certification results in Section 4. The deployed rule is fixed before the certification split is used. Let $\mathcal{F}$ denote all information available before certification, including the training data, the learned parameter $\hat{\theta}$, the budget T, the candidate-set construction, and all implementation choices. Conditional on $\mathcal{F}$, the map $x \mapsto \mathcal{U}^\star_{\hat{\theta}}(x)$ and the binary failure loss $E_{\hat{\theta}}(x,y) \in \{0,1\}$ are fixed. We write $R_{\mathcal{F}}(\hat{\theta}) = \mathbb{E}[E_{\hat{\theta}}(X,Y) \mid \mathcal{F}]$ for the conditional deployment risk. The main text suppresses this conditioning for readability.

F.1 Inverse-conformal interpretation of the add-one estimate

The quantity below provides an inverse-conformal interpretation of the add-one risk estimate; the realized high-probability certificate reported in the main text is obtained separately by applying Hoeffding's inequality directly to the empirical failure rate.

Define the inverse risk level assigned to the fixed deployed rule as

$$\widehat{R}_{\mathrm{IC}}(\hat{\theta}) = \inf_{r\in[0,1]} \left\{ r : 1 - \bar{E}_m(\hat{\theta}) \ge \frac{\lceil (m+1)(1-r) \rceil}{m} \right\}. \tag{24}$$

Algorithm 2: Post-Hoc Risk Certification for a Fixed Budget-Feasible Rule

Input: Fixed trained parameter $\hat{\theta}$, candidate family $\{\mathcal{U}_{\hat{\theta},\alpha}\}_{\alpha\in(0,1)}$, budget function $\mathcal{C}$, budget threshold T, certification set $\mathcal{D}_{\mathrm{cert}} = \{(X_i, Y_i)\}_{i=1}^{m}$, confidence level $\delta \in (0,1)$

Output: Budget-feasible deployment map $x \mapsto \mathcal{U}^{\star}_{\hat{\theta}}(x)$, and high-probability deployment-risk certificate $\widehat{R}_{\delta}(\hat{\theta})$

Data-use condition: $\hat{\theta}$, T, and all implementation choices are fixed before $\mathcal{D}_{\mathrm{cert}}$ is used. If the deployed rule is selected post hoc, then $\mathcal{D}_{\mathrm{cert}}$ denotes a held-out certification split not used for selection;

Budget-feasible deployment rule;
For any input x, compute

$$\bar{\alpha}_{\hat{\theta}}(x) = \inf\{\alpha \in (0,1) : \mathcal{C}(\mathcal{U}_{\hat{\theta},\alpha}(x), x) \le T\}, \qquad \mathcal{U}^{\star}_{\hat{\theta}}(x) = \mathcal{U}_{\hat{\theta},\bar{\alpha}_{\hat{\theta}}(x)}(x).$$

When the infimum is implemented on a grid, let $\mathcal{U}^{\star}_{\hat{\theta}}(x)$ denote the returned feasible set;

Certification stage;
Set $S \leftarrow 0$;
for $i = 1, \dots, m$ **do**
  Construct $\mathcal{U}^{\star}_{\hat{\theta}}(X_i)$ using the budget-feasible deployment rule above;
  Evaluate the exact deployment failure loss

$$e_i = E_{\hat{\theta}}(X_i, Y_i) = \begin{cases} \mathbf{1}\{\phi(Y_i, z_{\hat{\theta}}(X_i)) > r_{\hat{\theta}}(X_i)\}, & \text{continuous setting,} \\ \mathbf{1}\{\mathcal{U}^{\star}_{\hat{\theta}}(X_i) \cap \mathcal{G}(X_i, Y_i) = \varnothing\}, & \text{discrete setting.} \end{cases}$$

  Update $S \leftarrow S + e_i$;
Compute the empirical failure rate

$$\bar{E}_m(\hat{\theta}) = \frac{S}{m}.$$

Compute the high-probability deployment-risk certificate

$$\widehat{R}_{\delta}(\hat{\theta}) = \min\left\{1,\ \bar{E}_m(\hat{\theta}) + \sqrt{\frac{1}{2m}\log\frac{1}{\delta}}\right\}.$$

Deployment stage;
For a new input x, return $\mathcal{U}^{\star}_{\hat{\theta}}(x)$ using the same budget-feasible deployment rule;
return $x \mapsto \mathcal{U}^{\star}_{\hat{\theta}}(x)$, $\widehat{R}_{\delta}(\hat{\theta})$;

Lemma F.1 (Closed form). *Let $Z_i = E_{\hat{\theta}}(X_i, Y_i)$ for $i = 1, \dots, m$, and let $K_m = \sum_{i=1}^{m} Z_i$. Then the inverse risk level in* (24) *satisfies*

$$\widehat{R}_{\mathrm{IC}}(\hat{\theta}) = \frac{K_m + 1}{m+1} = \frac{1 + \sum_{i=1}^{m} E_{\hat{\theta}}(X_i, Y_i)}{m+1}. \tag{25}$$

Proof. Since $Z_i \in \{0,1\}$, we have $\bar{E}_m(\hat{\theta}) = K_m/m$ and $1 - \bar{E}_m(\hat{\theta}) = (m - K_m)/m$. Hence the constraint in (24) is equivalent to

$$\lceil (m+1)(1-r) \rceil \le m - K_m.$$

Because $m - K_m$ is an integer, this holds if and only if $(m+1)(1-r) \le m - K_m$, or equivalently $r \ge (K_m + 1)/(m+1)$. Taking the infimum over $r \in [0,1]$ gives

$$\widehat{R}_{\mathrm{IC}}(\hat{\theta}) = \frac{K_m + 1}{m+1}.$$

Substituting $K_m = \sum_{i=1}^{m} E_{\hat{\theta}}(X_i, Y_i)$ proves the stated closed form. □

Theorem F.2 (Expected inverse conformal conservativeness)**.** *Suppose the deployed rule $x \mapsto \mathcal{U}^{\star}_{\hat{\theta}}(x)$ is fixed before $\mathcal{D}_{\mathrm{cert}}$ is used, and conditional on $\mathcal{F}$, $(X_1, Y_1), \ldots, (X_m, Y_m), (X_{m+1}, Y_{m+1})$ are exchangeable. Then*

$$\mathbb{P}\big(E_{\hat{\theta}}(X_{m+1}, Y_{m+1}) = 0\big) \geq 1 - \mathbb{E}\Big[\widehat{R}_{\mathrm{IC}}(\hat{\theta})\Big]. \tag{26}$$

This gives an expected-conservative interpretation of the add-one estimate. The realized certificate in the main text is obtained separately from concentration of the empirical failure rate.

Proof. Let $Z_i = E_{\hat{\theta}}(X_i, Y_i)$ for $i = 1, \ldots, m+1$. Conditional on $\mathcal{F}$, the failure map $E_{\hat{\theta}}$ is fixed and the samples are exchangeable by assumption. Hence $Z_1, \ldots, Z_m, Z_{m+1}$ are exchangeable conditional on $\mathcal{F}$. In particular, $\mathbb{E}[Z_i \mid \mathcal{F}] = \mathbb{E}[Z_{m+1} \mid \mathcal{F}] = R_{\mathcal{F}}(\hat{\theta})$ for all $i = 1, \ldots, m$.

By Lemma F.1,

$$\widehat{R}_{\mathrm{IC}}(\hat{\theta}) = \frac{1 + \sum_{i=1}^{m} Z_i}{m+1}.$$

Taking conditional expectation gives

$$\mathbb{E}\Big[\widehat{R}_{\mathrm{IC}}(\hat{\theta}) \mid \mathcal{F}\Big] = \frac{1 + m R_{\mathcal{F}}(\hat{\theta})}{m+1}.$$

Since $0 \leq R_{\mathcal{F}}(\hat{\theta}) \leq 1$,

$$\frac{1 + m R_{\mathcal{F}}(\hat{\theta})}{m+1} - R_{\mathcal{F}}(\hat{\theta}) = \frac{1 - R_{\mathcal{F}}(\hat{\theta})}{m+1} \geq 0.$$

Thus

$$\mathbb{E}\Big[\widehat{R}_{\mathrm{IC}}(\hat{\theta}) \mid \mathcal{F}\Big] \geq R_{\mathcal{F}}(\hat{\theta}).$$

Moreover,

$$\mathbb{P}\big(E_{\hat{\theta}}(X_{m+1}, Y_{m+1}) = 0 \mid \mathcal{F}\big) = 1 - R_{\mathcal{F}}(\hat{\theta}).$$

Combining the two displays yields

$$\mathbb{P}\big(E_{\hat{\theta}}(X_{m+1}, Y_{m+1}) = 0 \mid \mathcal{F}\big) \geq 1 - \mathbb{E}\Big[\widehat{R}_{\mathrm{IC}}(\hat{\theta}) \mid \mathcal{F}\Big].$$

Taking expectation over $\mathcal{F}$ proves

$$\mathbb{P}\big(E_{\hat{\theta}}(X_{m+1}, Y_{m+1}) = 0\big) \geq 1 - \mathbb{E}\Big[\widehat{R}_{\mathrm{IC}}(\hat{\theta})\Big].$$

□

F.2 Proof of Proposition 4.1

Proof. Conditional on $\mathcal{F}$, the deployed rule and $E_{\hat{\theta}}$ are fixed. Under the i.i.d. certification assumption, let $Z_i = E_{\hat{\theta}}(X_i, Y_i)$ for $i = 1, \ldots, m$. Conditional on $\mathcal{F}$, the variables $Z_1, \ldots, Z_m$ are i.i.d. Bernoulli with $\mathbb{E}[Z_i \mid \mathcal{F}] = R_{\mathcal{F}}(\hat{\theta})$, and $\bar{E}_m(\hat{\theta}) = m^{-1} \sum_{i=1}^{m} Z_i$. By the one-sided Hoeffding inequality, for any $t > 0$,

$$\mathbb{P}\Big(R_{\mathcal{F}}(\hat{\theta}) - \bar{E}_m(\hat{\theta}) \geq t \mid \mathcal{F}\Big) \leq \exp(-2mt^2).$$

Taking $t = \sqrt{(2m)^{-1} \log(1/\delta)}$, we obtain, with conditional probability at least $1 - \delta$,

$$R_{\mathcal{F}}(\hat{\theta}) \leq \bar{E}_m(\hat{\theta}) + \sqrt{\frac{1}{2m} \log \frac{1}{\delta}}.$$

Since the conditional bound holds for every realization of $\mathcal{F}$, the same bound holds unconditionally. Dropping the explicit conditioning notation proves (12) with probability at least $1 - \delta$. □

F.3 COMBINED CERTIFICATE

Corollary F.3 (Budget-feasible certified deployment)**.** *Suppose the conditions of Proposition 4.1 hold, and suppose that for every deployment input the implementation returns a feasible budget-induced level. Define*

$$\widehat{R}_\delta(\hat{\theta}) = \min\left\{1,\ \bar{E}_m(\hat{\theta}) + \sqrt{\frac{1}{2m}\log\frac{1}{\delta}}\right\}.$$

Then, with probability at least $1-\delta$ *over the certification split,*

$$R(\hat{\theta}) \le \widehat{R}_\delta(\hat{\theta}).$$

Equivalently,

$$\mathbb{P}\big(E_{\hat{\theta}}(X,Y)=0\big) \ge 1 - \widehat{R}_\delta(\hat{\theta}).$$

In addition, for every input x *for which the implementation returns a feasible level,*

$$\mathcal{C}\Big(\mathcal{U}^\star_{\hat{\theta}}(x), x\Big) \le T.$$

Proof. The risk bound follows from Proposition 4.1 and the definition of $\widehat{R}_\delta(\hat{\theta})$. The success statement follows from $\mathbb{P}(E_{\hat{\theta}}(X,Y)=0) = 1 - R(\hat{\theta})$. The pointwise budget feasibility statement follows from the budget-induced construction of $\mathcal{U}^\star_{\hat{\theta}}$ in (6). □

F.4 COMPARISON WITH EXACT BINOMIAL BOUNDS

The high-probability certificate in Proposition 4.1 uses Hoeffding's inequality because it applies to general bounded deployment losses. In all experiments considered here, however, the deployment failure loss is binary. We therefore additionally compare the Hoeffding certificate with the one-sided Clopper–Pearson (CP) upper confidence bound.

Let K be the number of failures among m independent certification samples. Since $\bar{E}_m = K/m$, the clipped Hoeffding upper bound is $U_{\mathrm{H}} = \min\{1,\ K/m + \sqrt{\log(1/\delta)/(2m)}\}$.

For binary failures, the one-sided Clopper–Pearson upper bound is

$$U_{\mathrm{CP}} = \begin{cases} 1, & K = m, \\ F^{-1}_{\mathrm{Beta}(K+1,m-K)}(1-\delta), & K < m, \end{cases}$$

where $F^{-1}_{\mathrm{Beta}(a,b)}$ denotes the quantile function of a Beta distribution. We use $\delta = 0.05$ for both bounds.

Table 6: Comparison of one-sided 95% Clopper–Pearson and Hoeffding upper bounds for BCRC at $T = 5$. For continuous control, results are averaged over five seeds; for ImageNet, over nine backbones and five seeds; and for MMLU-Pro, over three LLMs and five seeds. Each bound is computed separately for each frozen deployed rule before aggregation. Mean ± standard deviation is descriptive and is not itself a new 95% confidence bound.

Setting	m	CP UCB (95%)	Hoeffding UCB (95%)	H–CP
Continuous synthetic	1500	0.5105 ± 0.0118	0.5205 ± 0.0118	0.0100
ImageNet	8000	0.0893 ± 0.0245	0.0977 ± 0.0239	0.0085
MMLU-Pro	2407	0.2037 ± 0.0429	0.2151 ± 0.0418	0.0114

Across the evaluated BCRC configurations, the exact binomial bound is consistently tighter than the Hoeffding bound, as expected from its use of the Bernoulli failure model. The difference is modest in the representative $T = 5$ comparison above, ranging from approximately 0.8 to 1.1 percentage points on average. We retain Hoeffding in the main theoretical development because it applies directly to general losses in $[0, 1]$, while CP is reported as a tighter binary-specific reference.

F.5 Certification Sample Size

The certification set is used only after the deployed rule has been fixed. Therefore, its sample size affects the tightness of the reported risk bound, but does not change the deployed rule or its pointwise budget feasibility. For the Hoeffding certificate, the correction term is $\epsilon_m = \sqrt{\log(1/\delta)/(2m)}$. Thus, to make this correction at most ϵ, it is sufficient to use $m \geq \log(1/\delta)/(2\epsilon^2)$.

At 95% confidence, corrections of at most 0.10, 0.05, and 0.025 require $m \geq 150$, 600, and 2397, respectively. Hence, there is no universal minimum certification-set size; the required m depends on the desired precision of the risk certificate. The exact binomial bound in Appendix F.4 tightens similarly as the certification sample size increases. In our continuous, ImageNet, and MMLU-Pro experiments, the certification sample sizes are 1500, 8000, and 2407, respectively.

F.6 Budget Selection in Practice

The deployment budget T can arise in two ways. First, in many applications it is imposed externally by the deployment system, such as a maximum shortlist length, review capacity, or an upper limit on a per-decision robust certificate. In this case, T is an operational constraint rather than a model-selection parameter, and BCRC directly learns and certifies the rule associated with that budget.

Second, when the budget is selected by the operator, one can specify a finite grid $\mathcal{T} = \{T_1, \ldots, T_K\}$ in advance and construct a certified budget–risk frontier. All candidate rules are fixed before the certification set is used. To allow T to be selected after inspecting this frontier, we compute each risk upper bound at confidence level $1 - \delta/K$. By a union bound, with probability at least $1 - \delta$, all K bounds then hold simultaneously; independence across budgets is not required.

Given a target deployment-risk level r_{target}, the operator may select

$$\widehat{T} = \min \left\{ T_k \in \mathcal{T} : \widehat{R}_{\delta/K}(T_k) \leq r_{\mathrm{target}} \right\}.$$

Thus, the selected budget is the smallest candidate budget whose certified risk satisfies the desired requirement. If no fixed risk target is available, the same frontier can instead be used to compare the reduction in certified risk against the additional computation, display space, or human-review effort required by a larger budget. An independent held-out selection split provides an alternative to simultaneous certification when desired.

G Experiment details and Additional results

Existing assets and licenses. We use existing datasets, pretrained models, and baseline methods only for research evaluation, and we do not redistribute these assets. The real-world portfolio experiment uses historical US stock market data following the setup of Chenreddy et al. (2022). ImageNet is used under the ImageNet Terms of Access. MMLU-Pro was downloaded from Hugging Face and is listed under the MIT License. The LLMs used in our experiments were also downloaded from Hugging Face: Qwen-7B is provided under the Tongyi Qianwen License Agreement, Qwen2.5-14B under the Apache-2.0 License, and Llama3.1-8B under the Llama 3.1 Community License. For the image experiments, we use standard ImageNet-pretrained classifiers, including ResNet, ResNeXt, DenseNet, VGG, Inception, and ShuffleNet variants, only for evaluation. We cite the original papers for all datasets, models, architectures, and baseline methods used in the experiments, and respect their corresponding licenses and terms of use.

Compute resources. All experiments were run on an x86-based machine equipped with a TianGai 150 accelerator. Each reported experimental run took approximately 3 hours. The total compute scales approximately linearly with the number of random seeds, model backbones, and budget configurations.

G.1 Experimental details

Common evaluation protocol. All experiments use disjoint data for fitting or model selection, post-hoc certification, and final empirical evaluation. Model parameters and hyperparameters are

fixed before the post-hoc certification split is used, and empirical metrics are reported on a held-out test split. In the experimental notation, D_{fit} is the training set $\mathcal{D}_{\mathrm{tr}}$ from Section 3.2, and D_{cert} is the independent sample used for post-hoc certification in Section 4. When a separate D_{cal} is present, it is used only to calibrate the forward baselines. Risk Upper Bound denotes the post-hoc high-probability certificate $\widehat{R}_\delta(\hat{\theta})$ from Proposition 4.1. We use $\delta = 0.05$. If the numerical backward search in (6) finds no feasible level, we return a deterministic budget-feasible fallback set fixed before certification.

Paired statistical comparisons. We assess matched-budget risk differences using a hierarchical paired bootstrap with 10,000 replicates on held-out test failures. Resampling preserves the experimental hierarchy and the pairing between BCRC and each baseline: seeds and paired test examples in continuous control; backbones, seeds, and paired test examples for ImageNet; and base models, seeds, and paired questions for MMLU-Pro. For comparisons aggregated across budgets, the budget conditions are retained as repeated conditions within the sampled experimental unit. We report 95% bootstrap confidence intervals and two-sided null-centered bootstrap p-values. Differences are defined as BCRC minus baseline, so negative values favor BCRC. The reported intervals and p-values are pairwise summaries for the pre-specified baseline comparisons rather than simultaneous confidence statements across baselines.

Evaluation metrics. In addition to the metrics defined in the main text, we report $\mathrm{Coverage} = n^{-1}\sum_i \mathbf{1}\{y_i \in \mathcal{U}_i\}$. For continuous robust control, $\mathrm{Robustness} = n^{-1}\sum_i \mathbf{1}\{\phi(y_i, z_i) \le r_i\} = 1 - \mathrm{Failure\ Risk}$. In the single-label discrete experiments, coverage likewise equals one minus the shortlist miss rate. For Tail ECE, let $d_i = 1 - \sum_{a\in\mathcal{U}_i} \pi(a \mid x_i)$ be the discarded predictive mass and $e_i = \mathbf{1}\{\mathcal{U}_i \cap \mathcal{G}(x_i, y_i) = \varnothing\}$. For a fixed partition of $[0, 1]$, let I_b contain the examples whose d_i lies in bin b. We compute

$$\mathrm{Tail\ ECE} = \sum_{b: I_b \neq \varnothing} \frac{|I_b|}{n} \left| \frac{1}{|I_b|} \sum_{i\in I_b} d_i - \frac{1}{|I_b|} \sum_{i\in I_b} e_i \right|. \tag{27}$$

Tail ECE is used as a validation diagnostic for MMLU-Pro rather than as a primary test metric. The same bin partition is used across methods within each experiment.

G.1.1 Experimental details of Continuous Robust Control

Continuous setup. For each seed, we use five disjoint splits, $\mathcal{D}_{\mathrm{train}}$, $\mathcal{D}_{\mathrm{val}}$, $\mathcal{D}_{\mathrm{cal}}$, $\mathcal{D}_{\mathrm{cert}}$, and $\mathcal{D}_{\mathrm{test}}$, each containing 1500 examples. The validation split is used for model and hyperparameter selection, the calibration split for the forward radii of CRO and E2E, the certification split for post-hoc certification, and the test split for final empirical evaluation. The base model is trained by Gaussian negative log-likelihood using Adam with learning rate 10^{-3}, weight decay 10^{-3}, batch size 32, and 250 epochs. The Gaussian models are fitted in cost coordinates $c = -Y$; reported losses in the paper are converted back through $\phi(Y, z) = c^\top z = -Y^\top z$. We initialize from the NLL-trained uncertainty model and train the budget-aware objective separately for each hard budget T. The BCRC hyperparameters are: 200 epochs, batch size 256, optimization settings selected using validation data, weight decay 10^{-4}, initial multiplier 1.0, maximum multiplier 50, and backward-search tolerance 5×10^{-4}. During training, we use an α-grid of size 81 and a simplex grid of size 301 for the soft decision relaxation, with $\tau_z = 0.05$. The budget and failure temperatures follow the 200-epoch linear schedules $\tau_b(e) = 0.6 + \frac{e-1}{199}(0.12 - 0.6)$ and $\tau_f(e) = 0.4 + \frac{e-1}{199}(0.10 - 0.4)$, respectively. The conditional Gaussian log-density supplies the score s_θ, so its superlevel sets are ellipsoids. After training, the hard budget-feasible level is reconstructed by bracketing on the α-grid followed by feasible-side bisection. The deployed decision is then obtained with the continuous-simplex convex solver, up to numerical tolerance. Writing $\langle\cdot\rangle$ for a minibatch average, $F = \langle \widetilde{E}_\theta \rangle$, and $D_T = \max\{|T|, 1\}$, training adds

$$\beta_F F + \beta_D \frac{\langle c^\top \widetilde{z} \rangle}{D_T} + \beta_N \,\mathrm{NLL} + \beta_S \left\langle \frac{[T - \widetilde{r}]_+}{D_T} \right\rangle + \beta_{\mathrm{fb}} \langle w_{\mathrm{fb}} \rangle, \qquad c = -Y,$$

to the core Lagrangian in (10). The three pre-specified validation candidates for $(\mathrm{LR}, \eta_\lambda, \beta_F, \beta_D, \beta_N, \beta_S, \beta_{\mathrm{fb}})$ are

$$(10^{-4}, .08, .5, .05, .01, .02, .20),$$
$$(2\times10^{-4}, .04, 1, .05, .01, .05, .30),$$
$$(3\times10^{-4}, .08, 1, .10, .02, .02, .50).$$

Selection uses only $\mathcal{D}_{\mathrm{val}}$; among the 20 final rules, the three candidates are selected 14, 6, and 0 times, respectively.

Budget Sensitivity. We next examine how the continuous BCRC rule changes as the hard budget threshold T is relaxed. In the continuous robust-control setting, T determines how much of the uncertainty set can be retained at deployment time. A smaller T forces more aggressive truncation, while a larger T allows $\mathcal{U}^\star_{\hat{\theta}}(x)$ to contain more outcome uncertainty.

Table 7: Budget sensitivity of BCRC in continuous robust control across five seeds. Larger T relaxes the hard deployment budget while preserving pointwise budget feasibility.

Metric	$T = 5$	$T = 8$	$T = 10$	$T = 12$
Coverage ↑	0.402 ± 0.015	0.569 ± 0.017	0.637 ± 0.021	0.715 ± 0.020
Avg. Certificate	4.883 ± 0.077	7.060 ± 0.376	8.978 ± 0.602	10.417 ± 0.519
Robustness ↑	0.511 ± 0.010	0.625 ± 0.013	0.692 ± 0.016	0.751 ± 0.020
Risk Upper Bound ↓	0.521 ± 0.012	0.399 ± 0.008	0.336 ± 0.008	0.280 ± 0.007
BVR ↓	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$	$\mathbf{0.000 \pm 0.000}$

Table 7 presents the corresponding budget–reliability trade-off. As T increases from 5 to 12, coverage improves from 40.2% to 71.5%, while robustness increases from 51.1% to 75.1% and budget violations remain zero. The average robust certificate increases from 4.883 to 10.417, as larger budgets allow the deployed rule to retain more outcome uncertainty. At the same time, the empirical failure risk and its post-hoc upper bound decrease as the budget is relaxed.

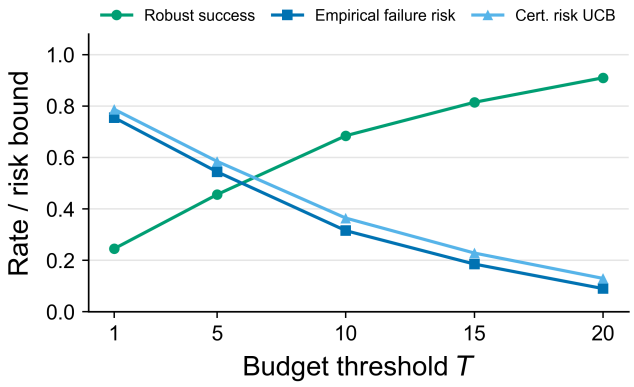


Figure 3: Budget-risk sensitivity of BCRC in continuous robust control. Relaxing T improves robust success while preserving pointwise budget feasibility.

Figure 3 further examines the budget-risk sensitivity of BCRC. As T increases, the deployed uncertainty set can retain more outcome uncertainty under the hard constraint. Consequently, failure risk decreases and robust success increases. The certified risk upper bound follows the same decreasing trend, suggesting that the post-hoc certificate reflects the empirical budget-risk frontier.

Real-world experiment. As an additional evaluation beyond the synthetic setting, we report results on a real-world US stock market dataset in Chenreddy et al. (2022). The dataset covers 64 US stocks across 8 sectors from January 1, 2012, to December 31, 2020. The data are computed based on the adjusted closing prices of consecutive trading days.

Table 8: Continuous robust-control results of US stock dataset under hard budget constraints.

Method	T	Coverage	Failure risk	Avg. Certificate	Decision Loss	BVR
CRC	0.5	0.000 ± 0.000	0.138 ± 0.027	0.951 ± 0.087	-0.121 ± 0.064	0.994 ± 0.006
CRO	0.5	0.899 ± 0.012	0.001 ± 0.001	9.025 ± 0.586	-0.120 ± 0.070	1.000 ± 0.000
E2E	0.5	0.903 ± 0.010	0.006 ± 0.003	5.420 ± 0.530	-0.120 ± 0.072	0.999 ± 0.001
BCRC	0.5	0.478 ± 0.069	0.279 ± 0.020	0.472 ± 0.017	-0.128 ± 0.070	$\mathbf{0.000 \pm 0.000}$
CRC	1	0.000 ± 0.000	0.138 ± 0.027	0.951 ± 0.087	-0.121 ± 0.064	0.836 ± 0.269
CRO	1	0.899 ± 0.012	0.001 ± 0.001	9.025 ± 0.586	-0.120 ± 0.070	1.000 ± 0.001
E2E	1	0.903 ± 0.010	0.006 ± 0.003	5.420 ± 0.530	-0.120 ± 0.072	0.998 ± 0.002
BCRC	1	0.691 ± 0.060	0.149 ± 0.013	0.983 ± 0.007	-0.119 ± 0.064	$\mathbf{0.000 \pm 0.000}$
CRC	1.5	0.000 ± 0.000	0.138 ± 0.027	0.951 ± 0.087	-0.121 ± 0.064	0.001 ± 0.002
CRO	1.5	0.899 ± 0.012	0.001 ± 0.001	9.025 ± 0.586	-0.120 ± 0.070	1.000 ± 0.001
E2E	1.5	0.903 ± 0.010	0.006 ± 0.003	5.420 ± 0.530	-0.120 ± 0.072	0.997 ± 0.003
BCRC	1.5	0.777 ± 0.028	0.081 ± 0.010	1.480 ± 0.004	-0.116 ± 0.065	$\mathbf{0.000 \pm 0.000}$

Table 8 indicates that CRO and E2E have budget violation rates close to one under all tested thresholds, making them infeasible under the hard budgets. CRC has lower failure risk under tighter budgets, but its outputs violate the prescribed budget in most cases. When the budget is sufficiently relaxed, CRC becomes nearly feasible, but BCRC still attains lower failure risk while maintaining zero BVR.

In contrast, BCRC has a dramatically improved risk frontier as the budget is relaxed. The average certificate remains close to the prescribed budget, indicating that BCRC effectively uses the available budget without violating it. Moreover, the decision loss remains comparable to the forward baselines, suggesting that the gain in deployability does not come at the cost of substantially worse downstream decisions in real-world data.

G.1.2 Experimental details of Discrete Finite-Candidate Shortlisting

In the discrete experiments, the failure indicator reduces to the label-missing event $E_{\hat{\theta}}(x_i, y_i) = \mathbf{1}\{y_i \notin \mathcal{U}_i\}$. Thus, the reported failure risk is the miss rate, $\mathrm{Risk} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{y_i \notin \mathcal{U}_i\}$. We also report the average set size $\mathrm{Size} = \frac{1}{n}\sum_{i=1}^{n} |\mathcal{U}_i|$, and the budget violation rate $\mathrm{BVR} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{|\mathcal{U}_i| > T\}$. So the deployed shortlist must contain at most T candidates for each input.

LLM shortlisting setup. For the LLM finite-candidate shortlisting experiments, we use MMLU-Pro, which provides up to ten candidate answers per question. For each question, the effective budget is $T_x = \min\{T, |\mathcal{Y}(x)|\}$. We use a five-way split $D_{\mathrm{fit}}, D_{\mathrm{val}}, D_{\mathrm{cal}}, D_{\mathrm{cert}}, D_{\mathrm{test}}$. The default split fractions are 20%, 15%, 25%, 20%, and 20%. For MMLU-Pro, this gives $m_{\mathrm{cert}} = 2407$ certification examples in the certification split. Forward conformal baselines are calibrated on D_{cal}, with RAPS and SAPS hyperparameters tuned using D_{val}. The reported Risk Upper Bound is computed on D_{cert} after the BCRC rule is fixed. Since all model selection and hyperparameter tuning are completed on D_{val}, no additional post-selection correction is needed for the reported certification bound. For each T, BCRC is fitted on D_{fit}, selected on D_{val}, certified on D_{cert}, and evaluated on D_{test}.

The inputs are frozen LLM scores. The 13 features summarize the frozen option scores and probabilities, within-question ranks and score gaps, cumulative probability and entropy, row-level score statistics, option position, and the number of valid options; feature standardization is fitted only on D_{fit}. The ranker is linear or has one GELU hidden layer of width 16 or 32, with an option bias and a rank-4 subject–option interaction. Each configuration is trained for 120 epochs using AdamW with batch size 256, weight decay 10^{-4}, and gradient clipping at 5. The multiplier uses $\lambda_0 = 1$, dual learning rate 0.20, and cap 10. The auxiliary loss contains $[g]_+^2$, 10^{-3} CE, and a 10^{-4}-weighted residual penalty. Learning rate, hidden width, residual cap, and temperature are selected from pre-specified configurations using only D_{val}. The sigmoid temperature t is selected from the pre-specified set $\{0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.50\}$ using D_{val}. The fit, validation, calibration, certification, and test splits contain 2409, 1804, 3007, 2407, and 2405 examples, respectively.

Budget-conditional joint training. To examine whether a separate model is required for each deployment budget, we additionally train a single budget-conditional BCRC score $s_\theta(x, a, T)$ jointly

over the pre-specified budget set $\mathcal{T} = \{1, 3, 5, 7, 9\}$ on Qwen-7B. The budget T is provided as an additional conditioning variable, while deployment still uses the exact Top-K rule for the requested budget. One checkpoint is selected using only fitting and validation data and is frozen before certification and testing.

Table 9: Budget-specific versus budget-conditional BCRC on Qwen-7B. Values are mean ± standard deviation across five split seeds. Both variants satisfy the pointwise hard budget for every evaluated budget.

T	Budget-specific risk	Budget-conditional risk
1	0.601 ± 0.010	$\mathbf{0.597 \pm 0.011}$
3	0.339 ± 0.010	$\mathbf{0.333 \pm 0.008}$
5	0.202 ± 0.010	$\mathbf{0.198 \pm 0.006}$
7	0.099 ± 0.005	$\mathbf{0.098 \pm 0.009}$
9	0.026 ± 0.003	$\mathbf{0.024 \pm 0.003}$

Budget Sensitivity. We examine how the discrete BCRC rule changes as the cardinality budget varies. We fix the base model to Qwen-7B and vary T, while keeping the scored cache, data split, and evaluation protocol fixed.

Table 10: Budget sensitivity of BCRC on Qwen-7B. Larger T reduces empirical failure risk and improves coverage, while preserving budget feasibility.

T	Failure risk ↓	Coverage ↑	Avg. Size	BVR ↓	Risk Upper Bound ↓
1	0.6011 ± 0.0099	0.3989 ± 0.0099	1.000 ± 0.000	$\mathbf{0.000 \pm 0.000}$	0.6355 ± 0.0066
3	0.3390 ± 0.0102	0.6610 ± 0.0102	3.000 ± 0.000	$\mathbf{0.000 \pm 0.000}$	0.3748 ± 0.0128
5	0.2017 ± 0.0097	0.7983 ± 0.0097	4.948 ± 0.003	$\mathbf{0.000 \pm 0.000}$	0.2295 ± 0.0084
7	0.0994 ± 0.0051	0.9006 ± 0.0051	6.834 ± 0.006	$\mathbf{0.000 \pm 0.000}$	0.1297 ± 0.0043
9	0.0264 ± 0.0028	0.9736 ± 0.0028	8.655 ± 0.014	$\mathbf{0.000 \pm 0.000}$	0.0563 ± 0.0037

Table 10 shows a clear budget–risk frontier. As T increases from 1 to 9, the empirical failure risk decreases from 0.6011 to 0.0264, while coverage increases from 0.3989 to 0.9736. BVR remains zero across all tested budgets. The average shortlist size can be slightly below T because some MMLU-Pro examples contain fewer than T available candidates, so the effective budget is $T_x = \min\{T, |\mathcal{Y}(x)|\}$. The post-hoc risk upper bound decreases consistently as the budget is relaxed. Figure 4 visualizes the

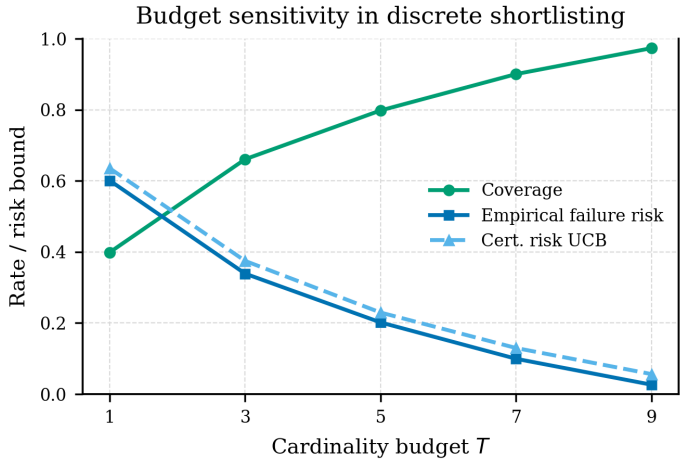


Figure 4: Budget-risk frontier for discrete BCRC on Qwen-7B.

budget–risk frontier in the discrete setting. Relaxing the deployment budget leads to lower empirical failure risk and higher success or coverage. The post-hoc certified risk upper bound tracks the same trend, suggesting that the post-hoc certificate reflects the empirical budget–risk trade-off.

G.2 Additional Continuous Experimental Results

We compare BCRC with CRO (Sun et al., 2023), CRC (Hu et al., 2026), and E2E (Chenreddy & Delage, 2024). All continuous methods use the same train, validation, calibration, certification, and test splits.

G.2.1 Baseline Methods

CRO, proposed by Sun et al. (2023), trains a predictive model to output a parametric uncertainty set. In our experiment, we use an ellipsoidal set in the experiment. The uncertainty set is then calibrated on a held-out calibration split to satisfy a target coverage level. After calibration, the calibrated uncertainty set is directly embedded into the downstream robust optimization problem. Thus it can solve the downstream robust decision problem.

E2E denotes the end-to-end conditional robust optimization baseline proposed by Chenreddy & Delage (2024) and Yeh et al. (2025). Unlike two-stage predict-then-optimize methods, E2E first outputs the parameters of an input-dependent uncertainty set. Second, this uncertainty set is passed into a differentiable robust optimization layer, which solves the downstream robust optimization problem and returns the decision and task loss. Finally, the task loss is backpropagated through the differentiable optimization layer to update the parameters of the predictive model. After the model is trained, nonconformity scores are computed on the calibration split and a conformal threshold is selected to satisfy the nominal coverage level. Therefore, E2E trains the uncertainty-set model directly through the downstream robust optimization objective, rather than fitting the uncertainty set and solving the decision problem as two independent stages.

CRC is proposed by Hu et al. (2026) as a decision-aware extension of CRO. It also learns a parameterized uncertainty set, embeds it into the robust optimization problem, and calibrates the induced certificate at a fixed nominal level α. This provides a conformal baseline whose calibration target is directly tied to the robustness instead of merely marginal coverage.

Matched-budget truncation. We implement the three matched-budget baselines in cost coordinates $c = -Y$. For $b \in \{\mathrm{CRC}, \mathrm{CRO}, \mathrm{E2E}\}$, the forward uncertainty set is

$$\mathcal{U}_b^{\mathrm{base}}(x) = \left\{\mu_b(x) + L_b(x)u : \|u\|_2 \le \rho_b^0\right\}.$$

For CRC, $\rho_{\mathrm{CRC}}^0 = \sqrt{-2\log(0.1)}$, without recalibrating the radius using outcomes from $\mathcal{D}_{\mathrm{cal}}$. For CRO and E2E, ρ_b^0 is the Mahalanobis conformal quantile computed on the independent $\mathcal{D}_{\mathrm{cal}}$ using the corresponding frozen (μ_b, L_b).

Let $\Delta_2 = \{z \in \mathbb{R}_+^2 : \mathbf{1}^\top z = 1\}$ and define

$$C_b(\rho, x) = \min_{z \in \Delta_2} \left[\mu_b(x)^\top z + \rho \|L_b(x)^\top z\|_2\right].$$

The hard budget requires $C_b(\rho, x) \le T$. Whenever $C_b(0, x) \le T$, we compute the feasible-side numerical approximation

$$\rho_b^T(x) \approx \sup\left\{\rho \in [0, \rho_b^0] : C_b(\rho, x) \le T\right\}$$

and deploy

$$\mathcal{U}_b^T(x) = \left\{\mu_b(x) + L_b(x)u : \|u\|_2 \le \rho_b^T(x)\right\}.$$

The center and factor L_b remain fixed, and only the scalar radius is reduced. We then recompute the robust decision $z_b^T(x)$ and certificate $r_b^T(x)$, rather than retaining the forward decision or clipping its certificate.

If $C_b(0, x) > T$, we use the predefined cost-space fallback $\mathcal{U}_{\mathrm{fb}}(T) = \{(T, T)\}$, $z_{\mathrm{fb}} = (1/2, 1/2)$, and $r_{\mathrm{fb}} = T$. Radius truncation is used whenever the radius-zero set is feasible; otherwise the predefined budget-feasible fallback is returned. The fallback need not be a subset of the original uncertainty set. For these inputs, we set $z_b^T = z_{\mathrm{fb}}$ and $r_b^T = r_{\mathrm{fb}}$.

The numerical search uses the 81-point α-grid for bracketing, followed by feasible-side bisection with α-tolerance 5×10^{-4} and at most 40 refinement steps. The final decision is computed with the continuous two-asset simplex convex solver. The resulting failure indicator is

$$E_b^T(x, y) = \mathbf{1}\left\{\phi(y, z_b^T(x)) > r_b^T(x)\right\}.$$

The truncation rule is fixed without tuning on $\mathcal{D}_{\text{cert}}$ or $\mathcal{D}_{\text{test}}$. It is independently certified on $\mathcal{D}_{\text{cert}}$, and its test failure is evaluated on $\mathcal{D}_{\text{test}}$; fallback cases are included in both.

Table 11: Overall paired failure-risk differences for the continuous matched-budget comparison.

Baseline	BCRC–baseline diff. (pp)	95% CI	p
CRC + truncation	-3.6233	$[-4.5067, -2.7933]$	$< 10^{-4}$
CRO + truncation	-1.8600	$[-2.1933, -1.5567]$	$< 10^{-4}$
E2E + truncation	-1.6167	$[-1.8333, -1.4167]$	$< 10^{-4}$

G.2.2 Full Continuous Robust-Control Results

Table 12 provides the full method-level results for the continuous robust-control experiment. They extend the compact summary in Table 1 by reporting coverage, average certificate, decision loss, robustness, and BVR for each method under each fixed budget. Coverage is reported for interpretability but is not the optimization target in this task. Under tight budgets, BCRC may exclude the realized outcome from the uncertainty set while still producing a valid robust-decision certificate for the downstream loss event. Therefore, the primary reliability metric in this experiment is the deployment failure risk $E_{\hat{\theta}}$, not marginal set coverage.

The results show a consistent pattern across all budgets. Forward baselines obtain high coverage or robustness, but their uncertainty sets are not constructed to satisfy the hard deployment budget pointwise and therefore have nonzero BVR. BCRC enforces the budget by construction and achieves zero BVR for all tested T. As T increases, BCRC retains larger uncertainty sets, leading to higher coverage and robustness while preserving strict feasibility.

Table 12: Continuous robust-control comparison under hard deployment budgets. Avg. Certificate denotes the average certificate value. Lower values are better except for Coverage and Robustness.

Method	T	Coverage	Avg. Certificate	Decision Loss	Robustness	BVR
CRO	5	0.900 ± 0.010	9.163 ± 0.254	7.075 ± 0.277	0.986 ± 0.002	0.620 ± 0.006
	8					0.447 ± 0.015
	10					0.376 ± 0.012
	12					0.310 ± 0.018
CRC	5	0.809 ± 0.031	8.560 ± 0.345	7.104 ± 0.280	0.934 ± 0.009	0.576 ± 0.012
	8					0.425 ± 0.015
	10					0.352 ± 0.017
	12					0.290 ± 0.023
E2E	5	0.905 ± 0.007	12.030 ± 0.833	7.014 ± 0.275	0.998 ± 0.001	0.821 ± 0.053
	8					0.530 ± 0.031
	10					0.449 ± 0.019
	12					0.386 ± 0.022
BCRC	5	0.402 ± 0.015	4.883 ± 0.077	8.277 ± 0.331	0.511 ± 0.010	$\mathbf{0.000 \pm 0.000}$
	8	0.569 ± 0.017	7.060 ± 0.376	8.440 ± 0.397	0.625 ± 0.013	$\mathbf{0.000 \pm 0.000}$
	10	0.637 ± 0.021	8.978 ± 0.602	7.842 ± 0.713	0.692 ± 0.016	$\mathbf{0.000 \pm 0.000}$
	12	0.715 ± 0.020	10.417 ± 0.519	7.772 ± 0.448	0.751 ± 0.020	$\mathbf{0.000 \pm 0.000}$

G.3 Additional Discrete Experimental Results

We compare BCRC with APS (Romano et al., 2020), RAPS (Angelopoulos et al., 2021), SAPS (Huang et al., 2024), and Fixed-α Top-K in the discrete setting.

G.3.1 Baseline Methods

APS is adaptive prediction sets for conformal classification proposed by Romano et al. (2020), using the cumulative probability mass required to include the true label as the conformity score, and then calibrates this score on a held-out calibration set. It returns all labels whose cumulative probability rank is below the calibrated threshold and provides conformal finite-sample coverage under exchangeability.

Angelopoulos et al. (2021) propose RAPS to regularize APS by penalizing low-probability tail labels in the sorted candidate list, so as to reduce overly large prediction sets while preserving conformal coverage. It often produces smaller and more stable sets than APS. However, RAPS calibrates a fixed nominal level rather than enforcing a pointwise deployment budget. As a result, its average set size may be smaller, but individual sets can still exceed the hard budget T, leading to nonzero BVR.

SAPS, proposed by Huang et al. (2024), is a label-ranking-based conformal classification baseline. It replaces the probability-mass score used by APS with a score based on the sorted rank of the candidate label, together with a stabilizing randomized or smoothed scoring scheme. This makes it less sensitive to poorly calibrated probability tails and can improve set-size efficiency in classification.

Fixed-α Top-K selects a single calibrated set size K_α such that the top-K prediction set attains the target coverage level $1-\alpha$ and returns the K highest-scoring candidates.

In the discrete label space, the budget directly restricts the prediction-set size. Therefore, the primary comparison here is not simply which method has the lowest failure risk, but which method actually satisfies the budget. The three tables share a consistent pattern: BCRC controls the BVR at 0 across all backbone networks and all budgets. This means that its returned sets are always deployable at the sample level, rather than merely close to the budget on average. In contrast, even when other methods achieve lower failure risk, they often do so by relying on larger sets. Therefore, they are not always being compared with BCRC under the same feasible problem.

G.3.2 Additional Image-Classification Results

Image-classification setup. For image classification, we use the ImageNet validation set and evaluate frozen ImageNet-pretrained classifiers from torchvision. All images are processed with the standard ImageNet evaluation preprocessing: resize to 256, center crop to 224×224, conversion to tensor, and normalization with the ImageNet mean and standard deviation. We use temperature-scaled softmax probabilities for the forward conformal baselines. The nominal miscoverage level for the forward conformal baselines is $\alpha = 0.1$.

BCRC takes the frozen feature vector and raw logits as input. Its residual ranker is LayerNorm–Linear$(d, 64)$–GELU–Dropout(0.05)–Linear$(64, 1000)$. It is trained for 35 epochs using AdamW with learning rate 3×10^{-4}, weight decay 10^{-4}, batch size 256, and multiplier settings $\lambda_0 = 1$, dual learning rate 0.05, and cap 50. With g denoting the minibatch constraint gap, the auxiliary loss is

$$[g]_+^2 + 0.10\,\mathrm{CE}(u, y) + 0.05 t_d^2\,\mathrm{KL}(\mathrm{softmax}(\ell/t_d) \,\|\, \mathrm{softmax}(u/t_d)) + 10^{-4}\|d_\theta\|_2^2, \qquad t_d = 2.$$

The fit, validation, calibration, certification, and test splits contain 25000, 5000, 5000, 8000, and 7000 images, respectively.

Tables 13–16 report the forward image-classification results under budgets $T \in \{1, 2, 3, 5\}$. These results illustrate how the original forward conformal rules trade failure risk against pointwise budget feasibility as the hard cardinality constraint is varied. The corresponding risk–BVR trade-off over $T \in \{1, 2, 3, 5, 10\}$ is visualized below, followed by the full matched-budget comparison.

Across these strict-budget settings, the forward conformal baselines retain relatively low failure risk but do not control set size pointwise. RAPS, SAPS and APS exhibit nonzero BVR on many backbone–budget settings. In contrast, BCRC satisfies the prescribed cardinality budget by construction.

Under the relaxed budget $T = 5$, BCRC remains strictly feasible and its failure risk drops sharply relative to the $T = 1$ regime. For example, the risk of BCRC decreases from 0.207 to 0.055 on ResNeXt101, from 0.229 to 0.062 on DenseNet161, and from 0.214 to 0.058 on ResNet152. Moreover, BCRC outperforms the corresponding Fixed-α Top-K baseline on ResNeXt101, DenseNet161,

Table 13: Image classification results under hard budget constraint $T = 1$.

Model	Method	Size	Failure risk	BVR	Model	Method	Size	Failure risk	BVR
	Fixed-α Top-K	6.40 ± 0.55	0.095 ± 0.008	1.000 ± 0.000		Fixed-α Top-K	3.80 ± 0.45	0.081 ± 0.010	1.000 ± 0.000
	APS	16.64 ± 0.90	0.076 ± 0.003	0.571 ± 0.005		APS	13.47 ± 1.25	0.065 ± 0.004	0.463 ± 0.008
ResNet18	RAPS	4.75 ± 0.90	0.095 ± 0.008	0.732 ± 0.076	DenseNet161	RAPS	2.37 ± 0.16	0.094 ± 0.009	0.630 ± 0.041
	SAPS	4.76 ± 0.81	0.095 ± 0.007	0.785 ± 0.147		SAPS	2.62 ± 0.40	0.087 ± 0.012	0.525 ± 0.026
	BCRC	1.00 ± 0.00	0.303 ± 0.004	$\mathbf{0.000 \pm 0.000}$		BCRC	1.00 ± 0.00	0.229 ± 0.004	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	4.00 ± 0.00	0.085 ± 0.002	1.000 ± 0.000		Fixed-α Top-K	5.40 ± 0.55	0.095 ± 0.007	1.000 ± 0.000
	APS	12.89 ± 0.31	0.067 ± 0.002	0.469 ± 0.007		APS	14.60 ± 0.79	0.073 ± 0.005	0.545 ± 0.009
ResNet50	RAPS	2.61 ± 0.07	0.092 ± 0.004	0.601 ± 0.025	VGG16	RAPS	3.74 ± 0.28	0.095 ± 0.006	0.755 ± 0.141
	SAPS	2.52 ± 0.33	0.096 ± 0.009	0.620 ± 0.078		SAPS	3.87 ± 0.30	0.094 ± 0.004	0.706 ± 0.165
	BCRC	1.00 ± 0.00	0.238 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	1.00 ± 0.00	0.290 ± 0.002	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.094 ± 0.009	1.000 ± 0.000		Fixed-α Top-K	6.80 ± 0.45	0.097 ± 0.005	1.000 ± 0.000
	APS	11.53 ± 0.40	0.065 ± 0.003	0.450 ± 0.005		APS	85.40 ± 3.41	0.077 ± 0.002	0.503 ± 0.005
ResNet101	RAPS	2.30 ± 0.12	0.097 ± 0.005	0.583 ± 0.075	Inception	RAPS	6.26 ± 1.97	0.098 ± 0.009	0.767 ± 0.180
	SAPS	2.61 ± 0.40	0.086 ± 0.013	0.525 ± 0.055		SAPS	5.26 ± 0.75	0.097 ± 0.005	0.754 ± 0.192
	BCRC	1.00 ± 0.00	0.224 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	1.00 ± 0.00	0.300 ± 0.005	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.086 ± 0.008	1.000 ± 0.000		Fixed-α Top-K	7.40 ± 0.55	0.095 ± 0.006	1.000 ± 0.000
	APS	10.73 ± 0.34	0.064 ± 0.002	0.431 ± 0.006		APS	33.91 ± 2.18	0.071 ± 0.003	0.543 ± 0.009
ResNet152	RAPS	2.18 ± 0.12	0.095 ± 0.004	0.508 ± 0.053	ShuffleNet	RAPS	6.10 ± 1.44	0.095 ± 0.009	0.726 ± 0.156
	SAPS	2.20 ± 0.29	0.091 ± 0.008	0.504 ± 0.057		SAPS	5.00 ± 0.36	0.099 ± 0.004	0.966 ± 0.054
	BCRC	1.00 ± 0.00	0.214 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	1.00 ± 0.00	0.306 ± 0.003	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.00 ± 0.00	0.083 ± 0.002	1.000 ± 0.000					
	APS	19.81 ± 1.26	0.062 ± 0.005	0.399 ± 0.009					
ResNeXt101	RAPS	2.23 ± 0.11	0.090 ± 0.007	0.475 ± 0.044					
	SAPS	1.96 ± 0.03	0.095 ± 0.004	0.455 ± 0.017					
	BCRC	1.00 ± 0.00	0.207 ± 0.002	$\mathbf{0.000 \pm 0.000}$					

Table 14: Image classification results under hard budget constraint $T = 2$.

Model	Method	Size	Failure risk	BVR	Model	Method	Size	Failure risk	BVR
	Fixed-α Top-K	6.40 ± 0.55	0.095 ± 0.008	1.000 ± 0.000		Fixed-α Top-K	3.80 ± 0.45	0.081 ± 0.010	1.000 ± 0.000
	APS	16.64 ± 0.90	0.076 ± 0.003	0.443 ± 0.007		APS	13.47 ± 1.25	0.065 ± 0.004	0.329 ± 0.011
ResNet18	RAPS	4.75 ± 0.90	0.095 ± 0.008	0.623 ± 0.092	DenseNet161	RAPS	2.37 ± 0.16	0.094 ± 0.009	0.361 ± 0.096
	SAPS	4.76 ± 0.81	0.095 ± 0.007	0.666 ± 0.188		SAPS	2.62 ± 0.40	0.087 ± 0.012	0.379 ± 0.037
	BCRC	2.00 ± 0.00	0.197 ± 0.004	$\mathbf{0.000 \pm 0.000}$		BCRC	2.00 ± 0.00	0.128 ± 0.004	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	4.00 ± 0.00	0.085 ± 0.002	1.000 ± 0.000		Fixed-α Top-K	5.40 ± 0.55	0.095 ± 0.007	1.000 ± 0.000
	APS	12.89 ± 0.31	0.067 ± 0.002	0.334 ± 0.002		APS	14.60 ± 0.79	0.073 ± 0.005	0.414 ± 0.008
ResNet50	RAPS	2.61 ± 0.07	0.092 ± 0.004	0.446 ± 0.070	VGG16	RAPS	3.74 ± 0.28	0.095 ± 0.006	0.543 ± 0.053
	SAPS	2.52 ± 0.33	0.096 ± 0.009	0.385 ± 0.029		SAPS	3.87 ± 0.30	0.094 ± 0.004	0.619 ± 0.211
	BCRC	2.00 ± 0.00	0.139 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	2.00 ± 0.00	0.181 ± 0.002	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.094 ± 0.009	1.000 ± 0.000		Fixed-α Top-K	6.80 ± 0.45	0.097 ± 0.005	1.000 ± 0.000
	APS	11.53 ± 0.40	0.065 ± 0.003	0.315 ± 0.005		APS	85.40 ± 3.41	0.077 ± 0.002	0.402 ± 0.006
ResNet101	RAPS	2.30 ± 0.12	0.097 ± 0.005	0.284 ± 0.010	Inception-v3	RAPS	6.26 ± 1.97	0.098 ± 0.009	0.700 ± 0.232
	SAPS	2.61 ± 0.40	0.086 ± 0.013	0.369 ± 0.035		SAPS	5.26 ± 0.75	0.097 ± 0.005	0.554 ± 0.047
	BCRC	2.00 ± 0.00	0.126 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	2.00 ± 0.00	0.192 ± 0.004	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.086 ± 0.008	1.000 ± 0.000		Fixed-α Top-K	7.40 ± 0.55	0.095 ± 0.006	1.000 ± 0.000
	APS	10.73 ± 0.34	0.064 ± 0.002	0.294 ± 0.004		APS	33.91 ± 2.18	0.071 ± 0.003	0.446 ± 0.008
ResNet152	RAPS	2.18 ± 0.12	0.095 ± 0.004	0.255 ± 0.012	ShuffleNet-V2	RAPS	6.10 ± 1.44	0.095 ± 0.009	0.650 ± 0.200
	SAPS	2.20 ± 0.29	0.091 ± 0.008	0.318 ± 0.030		SAPS	5.00 ± 0.36	0.099 ± 0.004	0.744 ± 0.161
	BCRC	2.00 ± 0.00	0.119 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	2.00 ± 0.00	0.200 ± 0.003	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.00 ± 0.00	0.083 ± 0.002	1.000 ± 0.000					
	APS	19.81 ± 1.26	0.062 ± 0.005	0.278 ± 0.009					
ResNeXt101	RAPS	2.23 ± 0.11	0.090 ± 0.007	0.291 ± 0.076					
	SAPS	1.96 ± 0.03	0.095 ± 0.004	0.272 ± 0.008					
	BCRC	2.00 ± 0.00	0.112 ± 0.003	$\mathbf{0.000 \pm 0.000}$					

Table 15: Image classification results under hard budget constraint $T = 3$.

Model	Method	Size	Failure risk	BVR	Model	Method	Size	Failure risk	BVR
	Fixed-α Top-K	6.40 ± 0.55	0.095 ± 0.008	1.000 ± 0.000		Fixed-α Top-K	3.80 ± 0.45	0.081 ± 0.010	0.800 ± 0.447
	APS	16.64 ± 0.90	0.076 ± 0.003	0.379 ± 0.009		APS	13.47 ± 1.25	0.065 ± 0.004	0.271 ± 0.013
ResNet18	RAPS	4.75 ± 0.90	0.095 ± 0.008	0.500 ± 0.057	DenseNet161	RAPS	2.37 ± 0.16	0.094 ± 0.009	0.184 ± 0.035
	SAPS	4.76 ± 0.81	0.095 ± 0.007	0.572 ± 0.159		SAPS	2.62 ± 0.40	0.087 ± 0.012	0.277 ± 0.070
	BCRC	3.00 ± 0.00	0.151 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	3.00 ± 0.00	0.092 ± 0.003	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	4.00 ± 0.00	0.085 ± 0.002	1.000 ± 0.000		Fixed-α Top-K	5.40 ± 0.55	0.095 ± 0.007	1.000 ± 0.000
	APS	12.89 ± 0.31	0.067 ± 0.002	0.278 ± 0.003		APS	14.60 ± 0.79	0.073 ± 0.005	0.351 ± 0.009
ResNet50	RAPS	2.61 ± 0.07	0.092 ± 0.004	0.236 ± 0.015	VGG16	RAPS	3.74 ± 0.28	0.095 ± 0.006	0.388 ± 0.048
	SAPS	2.52 ± 0.33	0.096 ± 0.009	0.257 ± 0.055		SAPS	3.87 ± 0.30	0.094 ± 0.004	0.477 ± 0.068
	BCRC	3.00 ± 0.00	0.102 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	3.00 ± 0.00	0.140 ± 0.002	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.094 ± 0.009	0.200 ± 0.447		Fixed-α Top-K	6.80 ± 0.45	0.097 ± 0.005	1.000 ± 0.000
	APS	11.53 ± 0.40	0.065 ± 0.003	0.258 ± 0.004		APS	85.40 ± 3.41	0.077 ± 0.002	0.365 ± 0.006
ResNet101	RAPS	2.30 ± 0.12	0.097 ± 0.005	0.167 ± 0.024	Inception	RAPS	6.26 ± 1.97	0.098 ± 0.009	0.456 ± 0.082
	SAPS	2.61 ± 0.40	0.086 ± 0.013	0.271 ± 0.072		SAPS	5.26 ± 0.75	0.097 ± 0.005	0.468 ± 0.014
	BCRC	3.00 ± 0.00	0.093 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	3.00 ± 0.00	0.149 ± 0.002	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.20 ± 0.45	0.086 ± 0.008	0.200 ± 0.447		Fixed-α Top-K	7.40 ± 0.55	0.095 ± 0.006	1.000 ± 0.000
	APS	10.73 ± 0.34	0.064 ± 0.002	0.241 ± 0.005		APS	33.91 ± 2.18	0.071 ± 0.003	0.404 ± 0.009
ResNet152	RAPS	2.18 ± 0.12	0.095 ± 0.004	0.155 ± 0.023	ShuffleNet	RAPS	6.10 ± 1.44	0.095 ± 0.009	0.608 ± 0.226
	SAPS	2.20 ± 0.29	0.091 ± 0.008	0.197 ± 0.054		SAPS	5.00 ± 0.36	0.099 ± 0.004	0.619 ± 0.213
	BCRC	3.00 ± 0.00	0.088 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	3.00 ± 0.00	0.155 ± 0.002	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	3.00 ± 0.00	0.083 ± 0.002	0.000 ± 0.000					
	APS	19.81 ± 1.26	0.062 ± 0.005	0.235 ± 0.011					
ResNeXt101	RAPS	2.23 ± 0.11	0.090 ± 0.007	0.167 ± 0.022					
	SAPS	1.96 ± 0.03	0.095 ± 0.004	0.153 ± 0.004					
	BCRC	3.00 ± 0.00	0.081 ± 0.001	$\mathbf{0.000 \pm 0.000}$					

ResNet152, ResNet101, and ResNet50. Thus, once the budget is moderately relaxed, the budget-first construction can be both feasible and risk-competitive.

Table 16: Image classification results under hard budget constraint $T = 5$.

Model	Method	Size	Failure risk	BVR	Model	Method	Size	Failure risk	BVR
ResNet18	Fixed-α Top-K	6.40 ± 0.55	0.095 ± 0.008	1.000 ± 0.000	DenseNet161	Fixed-α Top-K	3.80 ± 0.45	0.081 ± 0.010	0.000 ± 0.000
	APS	16.64 ± 0.90	0.076 ± 0.003	0.311 ± 0.008		APS	13.47 ± 1.25	0.065 ± 0.004	0.218 ± 0.011
	RAPS	4.75 ± 0.90	0.095 ± 0.008	0.303 ± 0.048		RAPS	2.37 ± 0.16	0.094 ± 0.009	0.056 ± 0.013
	SAPS	4.76 ± 0.81	0.095 ± 0.007	0.368 ± 0.045		SAPS	2.62 ± 0.40	0.087 ± 0.012	0.115 ± 0.095
	BCRC	5.00 ± 0.00	0.108 ± 0.004	$\mathbf{0.000 \pm 0.000}$		BCRC	5.00 ± 0.00	0.062 ± 0.002	$\mathbf{0.000 \pm 0.000}$
ResNet50	Fixed-α Top-K	4.00 ± 0.00	0.085 ± 0.002	0.000 ± 0.000	VGG16	Fixed-α Top-K	5.40 ± 0.55	0.095 ± 0.007	0.400 ± 0.548
	APS	12.89 ± 0.31	0.067 ± 0.002	0.225 ± 0.005		APS	14.60 ± 0.79	0.073 ± 0.005	0.284 ± 0.010
	RAPS	2.61 ± 0.07	0.092 ± 0.004	0.088 ± 0.012		RAPS	3.74 ± 0.28	0.095 ± 0.006	0.217 ± 0.046
	SAPS	2.52 ± 0.33	0.096 ± 0.009	0.058 ± 0.082		SAPS	3.87 ± 0.30	0.094 ± 0.004	0.287 ± 0.016
	BCRC	5.00 ± 0.00	0.070 ± 0.003	$\mathbf{0.000 \pm 0.000}$		BCRC	5.00 ± 0.00	0.099 ± 0.002	$\mathbf{0.000 \pm 0.000}$
ResNet101	Fixed-α Top-K	3.20 ± 0.45	0.094 ± 0.009	0.000 ± 0.000	Inception	Fixed-α Top-K	6.80 ± 0.45	0.097 ± 0.005	1.000 ± 0.000
	APS	11.53 ± 0.40	0.065 ± 0.003	0.207 ± 0.004		APS	85.40 ± 3.41	0.077 ± 0.002	0.335 ± 0.006
	RAPS	2.30 ± 0.12	0.097 ± 0.005	0.066 ± 0.028		RAPS	6.26 ± 1.97	0.098 ± 0.009	0.293 ± 0.039
	SAPS	2.61 ± 0.40	0.086 ± 0.013	0.118 ± 0.096		SAPS	5.26 ± 0.75	0.097 ± 0.005	0.364 ± 0.042
	BCRC	5.00 ± 0.00	0.063 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	5.00 ± 0.00	0.110 ± 0.001	$\mathbf{0.000 \pm 0.000}$
ResNet152	Fixed-α Top-K	3.20 ± 0.45	0.086 ± 0.008	0.000 ± 0.000	ShuffleNet	Fixed-α Top-K	7.40 ± 0.55	0.095 ± 0.006	1.000 ± 0.000
	APS	10.73 ± 0.34	0.064 ± 0.002	0.191 ± 0.006		APS	33.91 ± 2.18	0.071 ± 0.003	0.363 ± 0.009
	RAPS	2.18 ± 0.12	0.095 ± 0.004	0.065 ± 0.024		RAPS	6.10 ± 1.44	0.095 ± 0.009	0.363 ± 0.054
	SAPS	2.20 ± 0.29	0.091 ± 0.008	0.039 ± 0.069		SAPS	5.00 ± 0.36	0.099 ± 0.004	0.386 ± 0.021
	BCRC	5.00 ± 0.00	0.058 ± 0.002	$\mathbf{0.000 \pm 0.000}$		BCRC	5.00 ± 0.00	0.114 ± 0.002	$\mathbf{0.000 \pm 0.000}$
ResNeXt101	Fixed-α Top-K	3.00 ± 0.00	0.083 ± 0.002	0.000 ± 0.000					
	APS	19.81 ± 1.26	0.062 ± 0.005	0.198 ± 0.010					
	RAPS	2.23 ± 0.11	0.090 ± 0.007	0.072 ± 0.017					
	SAPS	1.96 ± 0.03	0.095 ± 0.004	0.011 ± 0.002					
	BCRC	5.00 ± 0.00	0.055 ± 0.002	$\mathbf{0.000 \pm 0.000}$					

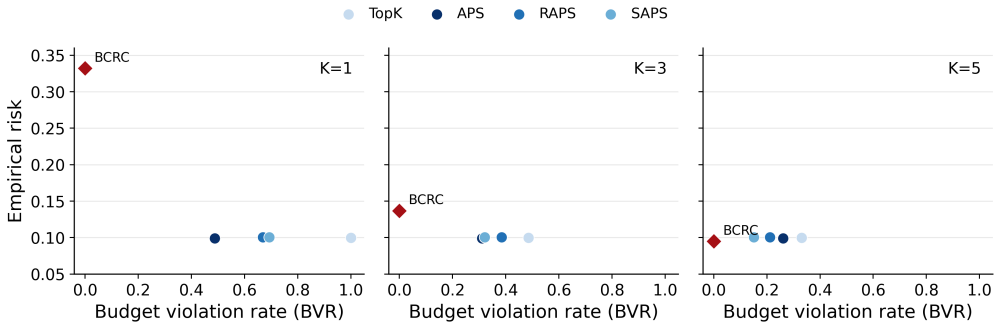


Figure 5: Risk–budget-violation trade-off in image classification over $T \in \{1, 2, 3, 5, 10\}$. The vertical axis shows failure risk and the horizontal axis shows budget violation rate. BCRC remains on the zero-BVR feasible axis for every budget, while the forward baselines may violate the pointwise cardinality constraint.

Figure 5 visualizes the same trade-off. In image classification, the forward baselines can achieve low empirical risk while violating the hard budget on a nonzero fraction of test examples. BCRC remains on the zero-BVR feasible axis for every tested budget, and its failure risk decreases as the budget is relaxed. This behavior complements the matched-budget comparisons below, where feasibility is equalized across methods.

Table 17: Paired failure-risk differences for the ImageNet matched-budget comparison at $T = 2$.

Baseline	BCRC–baseline diff. (pp)	95% CI	p
Direct Top-K	-0.3587	$[-0.5118, -0.1870]$	3.0×10^{-4}
Trunc. APS	-2.2397	$[-2.5311, -1.9260]$	$< 10^{-4}$
Trunc. RAPS	-1.1517	$[-1.4829, -0.8086]$	$< 10^{-4}$
Trunc. SAPS	-1.0762	$[-1.4283, -0.7076]$	$< 10^{-4}$

Differences are BCRC minus baseline; negative values favor BCRC.

Additional matched-budget result at $T = 5$. At $T = 5$, BCRC achieves lower mean failure risk than Direct Top-K and the truncated and re-certified baselines on all nine backbones. Direct

Table 18: Matched-budget image classification results under hard budget constraint $T = 5$. All deployed rules satisfy the same pointwise cardinality budget.

Model	Method	Size	Failure risk	Model	Method	Size	Failure risk
ResNet18	Direct Top-K	5.000 ± 0.000	0.1116 ± 0.0037	DenseNet161	Direct Top-K	5.000 ± 0.000	0.0666 ± 0.0025
	Trunc. APS	2.732 ± 0.028	0.1448 ± 0.0040		Trunc. APS	2.302 ± 0.044	0.1042 ± 0.0033
	Trunc. RAPS	3.292 ± 0.190	0.1235 ± 0.0061		Trunc. RAPS	2.279 ± 0.129	0.0974 ± 0.0081
	Trunc. SAPS	3.475 ± 0.512	0.1217 ± 0.0068		Trunc. SAPS	2.367 ± 0.178	0.0937 ± 0.0059
	BCRC	5.000 ± 0.000	**0.1082 ± 0.0044**		BCRC	5.000 ± 0.000	**0.0618 ± 0.0021**
ResNet50	Direct Top-K	5.000 ± 0.000	0.0723 ± 0.0018	VGG16	Direct Top-K	5.000 ± 0.000	0.0990 ± 0.0022
	Trunc. APS	2.328 ± 0.014	0.1091 ± 0.0022		Trunc. APS	2.623 ± 0.035	0.1313 ± 0.0034
	Trunc. RAPS	2.425 ± 0.106	0.0979 ± 0.0054		Trunc. RAPS	2.985 ± 0.155	0.1141 ± 0.0044
	Trunc. SAPS	2.404 ± 0.151	0.0992 ± 0.0061		Trunc. SAPS	3.181 ± 0.460	0.1119 ± 0.0070
	BCRC	5.000 ± 0.000	**0.0703 ± 0.0026**		BCRC	5.000 ± 0.000	**0.0988 ± 0.0025**
ResNet101	Direct Top-K	5.000 ± 0.000	0.0673 ± 0.0018	Inception-v3	Direct Top-K	5.000 ± 0.000	0.1171 ± 0.0026
	Trunc. APS	2.250 ± 0.016	0.1038 ± 0.0034		Trunc. APS	2.617 ± 0.021	0.1607 ± 0.0024
	Trunc. RAPS	2.140 ± 0.046	0.1020 ± 0.0047		Trunc. RAPS	3.272 ± 0.482	0.1348 ± 0.0107
	Trunc. SAPS	2.352 ± 0.177	0.0940 ± 0.0071		Trunc. SAPS	3.189 ± 0.209	0.1337 ± 0.0062
	BCRC	5.000 ± 0.000	**0.0633 ± 0.0024**		BCRC	5.000 ± 0.000	**0.1102 ± 0.0013**
ResNet152	Direct Top-K	5.000 ± 0.000	0.0602 ± 0.0014	ShuffleNet-V2	Direct Top-K	5.000 ± 0.000	0.1195 ± 0.0020
	Trunc. APS	2.177 ± 0.019	0.0983 ± 0.0022		Trunc. APS	2.772 ± 0.032	0.1534 ± 0.0024
	Trunc. RAPS	2.018 ± 0.043	0.0996 ± 0.0046		Trunc. RAPS	3.447 ± 0.571	0.1336 ± 0.0075
	Trunc. SAPS	2.119 ± 0.147	0.0927 ± 0.0048		Trunc. SAPS	3.804 ± 0.463	0.1248 ± 0.0044
	BCRC	5.000 ± 0.000	**0.0579 ± 0.0019**		BCRC	5.000 ± 0.000	**0.1142 ± 0.0022**
ResNeXt101	Direct Top-K	5.000 ± 0.000	0.0563 ± 0.0017				
	Trunc. APS	2.124 ± 0.040	0.0979 ± 0.0048				
	Trunc. RAPS	2.042 ± 0.122	0.0955 ± 0.0077				
	Trunc. SAPS	1.947 ± 0.029	0.0953 ± 0.0036				
	BCRC	5.000 ± 0.000	**0.0547 ± 0.0017**				

Top-K uses the full cardinality budget with the base-model scores, whereas the truncated conformal baselines often return smaller sets than the allowed budget.

Full matched-budget sweep. Following the detailed $T = 5$ comparison above, we report matched-budget results across the full budget grid $T \in \{1, 2, 3, 5, 10\}$. All deployed rules satisfy the same pointwise cardinality budget.

Table 19: Full matched-budget ImageNet sweep averaged over nine backbones. Each entry reports failure risk / average set size. All methods satisfy the same pointwise hard budget.

T	BCRC	Direct Top-K	Trunc. APS	Trunc. RAPS	Trunc. SAPS
1	0.257 / 1.000	0.263 / 1.000	0.263 / 1.000	0.263 / 1.000	0.263 / 1.000
2	0.155 / 2.000	0.158 / 2.000	0.177 / 1.486	0.166 / 1.642	0.166 / 1.649
3	0.117 / 3.000	0.120 / 3.000	0.148 / 1.848	0.134 / 2.103	0.133 / 2.127
5	0.082 / 5.000	0.086 / 5.000	0.123 / 2.436	0.111 / 2.656	0.107 / 2.760
10	0.051 / 10.000	0.053 / 10.000	0.099 / 3.603	0.098 / 3.192	0.094 / 3.318

Across the 45 backbone–budget settings, BCRC has lower mean failure risk than Top-K in $42/45$ settings and than truncated APS, RAPS, and SAPS in $44/45$, $44/45$, and $44/45$ settings, respectively. Averaged across all settings, the corresponding mean risk reductions are 0.364, 2.978, 2.222, and 2.029 percentage points. For BCRC versus Direct Top-K, the overall paired difference is -0.3640 pp (95% hierarchical paired-bootstrap CI $[-0.4840, -0.2260]$, $p < 10^{-4}$); BCRC has the lower five-seed mean failure risk in $42/45$ backbone–budget settings. The backbone–budget settings share backbones, nested budgets, and split structure, so the setting-level counts describe the evaluated grid rather than 45 independent tasks. The size columns also show that the truncated conformal baselines increasingly under-use the available budget as T grows, whereas BCRC and Top-K use the full cardinality budget.

Direct-risk training reference. To isolate the effect of the BCRC training criterion from learned ranking itself, we additionally train a Direct-risk learned Top-T reference with the same ranker architecture, initialization, data splits, optimizer, shared auxiliary terms, 35-epoch training budget, soft Top-T operator, and exact hard Top-T deployment. The Direct-risk reference replaces the BCRC-specific primary criterion with direct minimization of the soft miss $M = \mathrm{mean}(1 - w_y)$.

Across the 45 backbone–budget settings, the mean BCRC-minus-Direct-risk difference is -0.004698 pp (95% hierarchical paired-bootstrap CI $[-0.021652, 0.012763]$, two-sided $p = 0.596$); BCRC has lower mean risk in 26 settings and Direct-risk in 19, with no ties. Direct-risk uses the common method-independent checkpoint selector based on validation exact hard Top-T risk, soft miss, and then earlier epoch. The historical BCRC selector used validation exact hard Top-T risk, soft miss, soft discarded mass, and then earlier epoch; removing the discarded-mass tie-breaker changes none of the 225 selected BCRC checkpoints. The two training objectives are closely matched on the evaluated ImageNet grid, with no statistically supported difference under this paired analysis.

G.3.3 Additional LLM Results

Tables 20–22 provide the full budget sweep for the LLM finite-candidate shortlisting experiment. We report results for Qwen-7B, Qwen2.5-14B, and Llama3.1-8B under budgets $T \in \{1, 3, 5, 7, 9\}$. These tables extend the main-text $T = 5$ comparison by showing how each method behaves as the shortlist budget changes. Size denotes the average number of returned answer options, failure risk is the probability that the returned shortlist contains no correct answer, and BVR is the fraction of examples whose shortlist exceeds the prescribed budget.

Table 20: Results of different predictors for Qwen-7B under hard budget constraints.

T	Method	Size	Failure risk	BVR	T	Method	Size	Failure risk	BVR
	Fixed-α Top-K	7.574 ± 0.416	0.0715 ± 0.0219	1.000 ± 0.000		Fixed-α Top-K	7.574 ± 0.416	0.0715 ± 0.0219	0.740 ± 0.413
	APS	7.819 ± 0.092	0.0480 ± 0.0053	0.992 ± 0.001		APS	7.819 ± 0.092	0.0480 ± 0.0053	0.666 ± 0.022
1	RAPS	7.741 ± 0.141	0.0570 ± 0.0081	0.998 ± 0.004	7	RAPS	7.741 ± 0.141	0.0570 ± 0.0081	0.720 ± 0.121
	SAPS	6.845 ± 0.135	0.1020 ± 0.0068	0.976 ± 0.054		SAPS	6.845 ± 0.135	0.1020 ± 0.0068	0.373 ± 0.187
	BCRC	1.000 ± 0.000	0.6011 ± 0.0099	$\mathbf{0.000 \pm 0.000}$		BCRC	6.834 ± 0.006	0.0994 ± 0.0051	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	7.574 ± 0.416	0.0715 ± 0.0219	0.998 ± 0.001		Fixed-α Top-K	7.574 ± 0.416	0.0715 ± 0.0219	0.000 ± 0.000
	APS	7.819 ± 0.092	0.0480 ± 0.0053	0.943 ± 0.004		APS	7.819 ± 0.092	0.0480 ± 0.0053	0.261 ± 0.017
3	RAPS	7.741 ± 0.141	0.0570 ± 0.0081	0.988 ± 0.023	9	RAPS	7.741 ± 0.141	0.0570 ± 0.0081	0.064 ± 0.114
	SAPS	6.845 ± 0.135	0.1020 ± 0.0068	0.888 ± 0.151		SAPS	6.845 ± 0.135	0.1020 ± 0.0068	0.219 ± 0.240
	BCRC	3.000 ± 0.000	0.3390 ± 0.0102	$\mathbf{0.000 \pm 0.000}$		BCRC	8.655 ± 0.014	0.0264 ± 0.0028	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	7.574 ± 0.416	0.0715 ± 0.0219	0.947 ± 0.002					
	APS	7.819 ± 0.092	0.0480 ± 0.0053	0.831 ± 0.010					
5	RAPS	7.741 ± 0.141	0.0570 ± 0.0081	0.923 ± 0.052					
	SAPS	6.845 ± 0.135	0.1020 ± 0.0068	0.754 ± 0.178					
	BCRC	4.948 ± 0.003	0.2017 ± 0.0097	$\mathbf{0.000 \pm 0.000}$					

Table 21: Results of different predictors for Qwen2.5-14B under hard budget constraints.

T	Method	Size	Failure risk	BVR	T	Method	Size	Failure risk	BVR
	Fixed-α Top-K	6.459 ± 0.514	0.0755 ± 0.0197	1.000 ± 0.000		Fixed-α Top-K	6.459 ± 0.514	0.0755 ± 0.0197	0.000 ± 0.000
	APS	7.898 ± 0.081	0.0104 ± 0.0019	0.931 ± 0.011		APS	7.898 ± 0.081	0.0104 ± 0.0019	0.707 ± 0.009
1	RAPS	6.757 ± 0.188	0.0612 ± 0.0066	1.000 ± 0.000	7	RAPS	6.757 ± 0.188	0.0612 ± 0.0066	0.034 ± 0.044
	SAPS	5.535 ± 0.101	0.0975 ± 0.0063	0.964 ± 0.080		SAPS	5.535 ± 0.101	0.0975 ± 0.0063	0.272 ± 0.136
	BCRC	1.000 ± 0.000	0.4929 ± 0.0117	$\mathbf{0.000 \pm 0.000}$		BCRC	6.834 ± 0.006	0.0596 ± 0.0026	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	6.459 ± 0.514	0.0755 ± 0.0197	0.998 ± 0.001		Fixed-α Top-K	6.459 ± 0.514	0.0755 ± 0.0197	0.000 ± 0.000
	APS	7.898 ± 0.081	0.0104 ± 0.0019	0.853 ± 0.008		APS	7.898 ± 0.081	0.0104 ± 0.0019	0.566 ± 0.017
3	RAPS	6.757 ± 0.188	0.0612 ± 0.0066	0.998 ± 0.001	9	RAPS	6.757 ± 0.188	0.0612 ± 0.0066	0.000 ± 0.000
	SAPS	5.535 ± 0.101	0.0975 ± 0.0063	0.708 ± 0.268		SAPS	5.535 ± 0.101	0.0975 ± 0.0063	0.207 ± 0.163
	BCRC	3.000 ± 0.000	0.2431 ± 0.0103	$\mathbf{0.000 \pm 0.000}$		BCRC	8.655 ± 0.014	0.0132 ± 0.0023	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	6.459 ± 0.514	0.0755 ± 0.0197	0.947 ± 0.002					
	APS	7.898 ± 0.081	0.0104 ± 0.0019	0.765 ± 0.008					
5	RAPS	6.757 ± 0.188	0.0612 ± 0.0066	0.947 ± 0.002					
	SAPS	5.535 ± 0.101	0.0975 ± 0.0063	0.350 ± 0.083					
	BCRC	4.948 ± 0.003	0.1313 ± 0.0091	$\mathbf{0.000 \pm 0.000}$					

Full matched-budget comparison. To complement the representative $T = 5$ comparison in Table 4, we report matched-budget results across all three base LLMs and the full budget grid $T \in \{1, 3, 5, 7, 9\}$. All deployed rules satisfy the same pointwise cardinality budget. Each entry in Table 24 reports failure risk / average shortlist size.

Table 22: Results of different predictors for Llama3.1-8B under hard budget constraints.

T	Method	Size	Failure risk	BVR	T	Method	Size	Failure risk	BVR
	Fixed-α Top-K	7.758 ± 0.009	0.0711 ± 0.0058	1.000 ± 0.000		Fixed-α Top-K	7.758 ± 0.009	0.0711 ± 0.0058	0.924 ± 0.003
	APS	8.534 ± 0.078	0.0129 ± 0.0032	0.955 ± 0.006		APS	8.534 ± 0.078	0.0129 ± 0.0032	0.794 ± 0.010
1	RAPS	7.522 ± 0.094	0.0529 ± 0.0092	0.998 ± 0.004	7	RAPS	7.522 ± 0.094	0.0529 ± 0.0092	0.573 ± 0.040
	SAPS	6.554 ± 0.157	0.1004 ± 0.0106	1.000 ± 0.000		SAPS	6.554 ± 0.157	0.1004 ± 0.0106	0.390 ± 0.098
	BCRC	1.000 ± 0.000	0.6299 ± 0.0068	$\mathbf{0.000 \pm 0.000}$		BCRC	6.834 ± 0.006	0.1143 ± 0.0074	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	7.758 ± 0.009	0.0711 ± 0.0058	0.998 ± 0.001		Fixed-α Top-K	7.758 ± 0.009	0.0711 ± 0.0058	0.000 ± 0.000
	APS	8.534 ± 0.078	0.0129 ± 0.0032	0.920 ± 0.007		APS	8.534 ± 0.078	0.0129 ± 0.0032	0.637 ± 0.012
3	RAPS	7.522 ± 0.094	0.0529 ± 0.0092	0.973 ± 0.057	9	RAPS	7.522 ± 0.094	0.0529 ± 0.0092	0.167 ± 0.105
	SAPS	6.554 ± 0.157	0.1004 ± 0.0106	0.903 ± 0.131		SAPS	6.554 ± 0.157	0.1004 ± 0.0106	0.075 ± 0.103
	BCRC	3.000 ± 0.000	0.3756 ± 0.0077	$\mathbf{0.000 \pm 0.000}$		BCRC	8.655 ± 0.014	0.0279 ± 0.0023	$\mathbf{0.000 \pm 0.000}$
	Fixed-α Top-K	7.758 ± 0.009	0.0711 ± 0.0058	0.947 ± 0.002					
	APS	8.534 ± 0.078	0.0129 ± 0.0032	0.851 ± 0.010					
5	RAPS	7.522 ± 0.094	0.0529 ± 0.0092	0.803 ± 0.083					
	SAPS	6.554 ± 0.157	0.1004 ± 0.0106	0.710 ± 0.137					
	BCRC	4.948 ± 0.003	0.2234 ± 0.0067	$\mathbf{0.000 \pm 0.000}$					

Table 23: Paired failure-risk differences for the MMLU-Pro matched-budget comparison at $T = 5$.

Baseline	BCRC–baseline diff. (pp)	95% CI	p
Direct Top-K	-0.6209	$[-1.2557, -0.1358]$	0.0291
Trunc. APS	-0.6653	$[-1.3223, -0.1802]$	0.0250
Trunc. RAPS	-0.6209	$[-1.2529, -0.1414]$	0.0291
Trunc. SAPS	-0.6376	$[-1.2724, -0.1719]$	0.0258

Table 24: Full matched-budget MMLU-Pro sweep over three base LLMs. Each entry reports failure risk / average shortlist size. All deployed rules satisfy the same pointwise hard budget.

Model	T	BCRC	Direct Top-K	Trunc. APS	Trunc. RAPS	Trunc. SAPS
Qwen-7B	1	**0.6011** / 1.000	0.6216 / 1.000	0.6216 / 1.000	0.6216 / 1.000	0.6216 / 1.000
	3	**0.3390** / 3.000	0.3537 / 3.000	0.3537 / 3.000	0.3537 / 3.000	0.3537 / 3.000
	5	**0.2017** / 4.948	0.2141 / 4.948	0.2147 / 4.940	0.2141 / 4.948	0.2141 / 4.948
	7	**0.0994** / 6.834	0.1039 / 6.834	0.1045 / 6.823	0.1039 / 6.834	0.1039 / 6.834
	9	**0.0264** / 8.655	0.0267 / 8.655	0.0281 / 8.629	0.0267 / 8.655	0.0273 / 8.618
Qwen2.5-14B	1	0.4929 / 1.000	**0.4877** / 1.000	**0.4877** / 1.000	**0.4877** / 1.000	**0.4877** / 1.000
	3	**0.2431** / 3.000	0.2449 / 3.000	0.2449 / 3.000	0.2449 / 3.000	0.2449 / 3.000
	5	**0.1313** / 4.948	0.1357 / 4.948	0.1358 / 4.943	0.1357 / 4.948	0.1357 / 4.948
	7	**0.0596** / 6.834	0.0621 / 6.834	0.0623 / 6.827	0.0621 / 6.834	0.0621 / 6.834
	9	**0.0132** / 8.655	0.0135 / 8.655	0.0138 / 8.634	0.0135 / 8.655	0.0135 / 8.655
Llama3.1-8B	1	0.6299 / 1.000	**0.6277** / 1.000	**0.6277** / 1.000	**0.6277** / 1.000	**0.6277** / 1.000
	3	0.3756 / 3.000	**0.3721** / 3.000	**0.3721** / 3.000	**0.3721** / 3.000	0.3725 / 2.969
	5	**0.2234** / 4.948	0.2253 / 4.948	0.2259 / 4.944	0.2253 / 4.948	0.2258 / 4.916
	7	**0.1143** / 6.834	0.1157 / 6.834	0.1163 / 6.826	0.1157 / 6.834	0.1159 / 6.811
	9	**0.0279** / 8.655	0.0284 / 8.655	0.0292 / 8.630	0.0284 / 8.655	0.0309 / 8.406

Soft-budget extension. BCRC focuses on strict pointwise hard budgets, but some applications may allow occasional violations of a nominal operating budget. We consider an illustrative extension on Qwen-7B with nominal budget $T = 3$. We use the discarded predictive mass outside the Top-3 shortlist as an unlabeled difficulty score. For each pre-specified escalation fraction, a threshold is fixed on the validation set: queries above the threshold may return Top-4, while the remaining queries return Top-3. The escalation rule is fixed before certification and test evaluation.

Table 25: Illustrative soft-budget extension on MMLU-Pro with Qwen-7B and nominal budget $T = 3$. A pre-specified fraction of difficult queries may use one additional candidate. Results are from a single split and are intended only to illustrate the risk–budget trade-off.

Escalation fraction	Nominal BVR	Avg. Size	Failure risk
0.00	0.0000	3.000	0.3538
0.01	0.0087	3.009	0.3526
0.05	0.0349	3.035	0.3501
0.10	0.0886	3.089	0.3455
0.20	0.1896	3.190	0.3318

Allowing a small fraction of difficult queries to use one additional candidate reduces failure risk as nominal BVR and average shortlist size increase. This illustrates that, when occasional nominal-budget violations are acceptable, additional capacity can be exchanged for lower failure risk. Our main experiments instead focus on strict pointwise hard-budget deployment.