Proxy-Only NAS is Information-Theoretically Impossible: Minimal Calibration Conditions for Provable Architecture Selection

Liz Lemma Future Detective

January 20, 2026

Abstract

Neural Architecture Search (NAS) relies heavily on cheap proxies—weight-sharing supernets, early-stopped training, or proxy tasks—to avoid the prohibitive cost of training each architecture from scratch, as emphasized by the survey source material (efficient NAS, DARTS variants, random search with weight-sharing, performance predictors). Yet the field lacks clarity on what can be guaranteed when proxies are biased or misrank architectures. We propose a clean oracle model separating a cheap proxy signal P from the expensive true evaluation F. Our first result is an impossibility theorem: without any structural linkage between P and F, no proxy-only algorithm can guarantee a nontrivial approximation to the optimal architecture. We then introduce a minimal, testable proxy-to-true linkage condition—a bounded-distortion calibration assumption stating that $F(a)$ is close to an unknown transformation of $P(a)$ within a known function class. Under this condition we give a simple calibration-based algorithm with tight sample complexity: $\Theta(1/\varepsilon^2)$ true evaluations suffice (and are necessary) to select an ε-optimal architecture up to the irreducible distortion η. Finally, we outline diagnostics that empirically test whether a given NAS proxy satisfies the calibration condition on held-out architectures, offering a practical checklist for when proxies such as weight-sharing or early stopping can be trusted.

Table of Contents

3. 3. Formal Model: architecture set, proxy oracle, true evaluation oracle with noise and budget; approximation notions (multiplicative/additive); discussion of how common NAS proxies instantiate P.

4. 4. Impossibility of Proxy-Only NAS: indistinguishability construction; tightness of the bound; implications for DARTS/one-shot as purely proxy-driven ranking methods.

5. 5. Minimal Proxy-to-True Linkage Conditions: define bounded-distortion calibration; alternative candidate conditions (monotone calibration, Lipschitz, scaling-law parametric); which are testable and why.

6. 6. Algorithm: Calibrate-Then-Select. Learn calibration map from few true evaluations; use proxy to screen and then validate; optional top-k verification for robustness.

7. 7. Upper Bounds: sample complexity and guarantees under bounded distortion; finite-class and capacity-based bounds; robustness to noise and misspecification.

8. 8. Lower Bounds: matching $\Omega(1/\varepsilon^2)$ true-evaluation requirement via reduction from coin-bias (best-arm) testing; dependence on function class complexity.

9. 9. Proxy Diagnostics (Practical Add-on): held-out calibration tests, residual bounds, conformal intervals, and how to decide budget allocation between proxy and true evals (recommended but optional experiments).

10. 10. Discussion and Future Work: extending to Pareto objectives and hardware-aware costs; adaptive choice of function class; implications for benchmarks and reproducibility.

1 Introduction

Neural architecture search (NAS) is commonly posed as the problem of selecting, from a finite candidate set $\mathcal{A}$ with $|\mathcal{A}| = n$, an architecture $a \in \mathcal{A}$ maximizing an unknown performance objective $F(a) \in [0,1]$ (e.g., test accuracy after full training). The defining difficulty is that evaluating $F(a)$ is expensive: a single query typically requires a full training run, and the resulting observation is noisy due to stochastic optimization, data ordering, and evaluation variance. We therefore model each expensive evaluation as an oracle call returning $\widetilde{F}(a) = F(a) + \xi$, where ξ is mean-zero and sub-Gaussian. Under a budget constraint of at most B expensive evaluations, the goal is to output $\hat{a}$ with $F(\hat{a})$ close to $\mathrm{OPT} := \max_{a \in \mathcal{A}} F(a)$ with high probability.

To reduce the number of expensive evaluations, practical NAS pipelines use *proxies*—cheap signals $P(a) \in \mathcal{Y}$ intended to correlate with $F(a)$ while being much cheaper to compute. Typical examples include weight-sharing scores from one-shot models, early-stopping validation accuracy, training-free heuristics, or learned predictors. In abstraction, we grant the algorithm oracle access to a proxy oracle O_P returning $P(a)$ on query a, and we treat O_P queries as essentially free relative to O_F queries. This modeling choice isolates the fundamental question: what can be guaranteed from proxy information, and how many expensive evaluations are unavoidable?

Guarantees matter for two reasons. First, proxy-based methods are known to succeed or fail abruptly as the search space, training protocol, and proxy construction change. Empirically, one often observes ranking reversals: architectures that score highly under a proxy may train poorly when fully evaluated, and conversely strong architectures may be suppressed by proxy bias. Without a formal linkage between P and F, an algorithm that is “good” on one benchmark may be provably incapable of providing any uniform performance guarantee across problem instances. Second, NAS is frequently deployed under strict budgets, and the relevant question is not only whether a proxy correlates with F on average, but whether one can certify that a pipeline will return a near-optimal architecture with probability at least $1 - \delta$ while using at most B expensive evaluations.

Our first contribution is to formalize a minimal oracle model capturing the ubiquitous proxy–truth dichotomy. The algorithm may adaptively query O_P and O_F, but its dominant cost is the number of O_F calls. This allows us to separate two roles played by proxies in practice: (i) *screening*, where one uses P to restrict attention to a small candidate set, and (ii) *estimation*, where one uses expensive evaluations to resolve remaining uncertainty. The model is deliberately simple—finite $\mathcal{A}$, bounded objective values, and sub-Gaussian evaluation noise—so that any limitations we prove are not artifacts of pathological function classes or continuous domains.

Our second contribution is a worst-case impossibility statement for proxy-only NAS. If an algorithm has access only to O_P, then for any such algorithm we can construct instances in which the proxy conveys no information about which architecture is optimal. In this regime, even allowing randomization and adaptive proxy queries, no method can guarantee a nontrivial approximation to OPT uniformly over all (F, P). This observation is not a criticism of proxies per se; rather, it identifies the exact point at which additional assumptions are logically necessary. In particular, any meaningful performance guarantee must posit some relationship between P and F that rules out indistinguishable instances.

Our third contribution is to articulate a linkage assumption that is both weak enough to cover diverse proxy mechanisms and structured enough to yield tight sample complexity bounds. We assume there exists an unknown calibration function g in a known function class $\mathcal{G}$ such that $g(P(a))$ predicts $F(a)$ up to a bounded distortion η, i.e.,

$$|F(a) - g(P(a))| \leq \eta \qquad \text{for all } a \in \mathcal{A}.$$

This condition separates two sources of error: a *modeling* error η reflecting intrinsic proxy bias, and a *statistical* error arising from learning g from finitely many expensive evaluations. The parameter η is not assumed known a priori; it is a property of the given proxy and search space, and it can be empirically assessed via residuals on held-out calibration data. The function class $\mathcal{G}$ encodes any structural knowledge we are willing to assert (e.g., monotonicity or Lipschitzness), and its complexity $\mathrm{comp}(\mathcal{G})$ governs how many calibration evaluations are required for uniform generalization.

Our fourth contribution is an algorithmic template, *calibrate then select*, that explicitly mirrors common practice while admitting end-to-end guarantees. We use a subset of the expensive evaluation budget to collect calibration data $(P(a_i), \widetilde{F}(a_i))$ and fit $\hat{g} \in \mathcal{G}$. We then score architectures by $\hat{g}(P(a))$ using cheap proxy queries, retain a top-k candidate set, and spend the remaining budget validating candidates through additional calls to O_F. The analysis yields an additive guarantee of the form

$$F(\hat{a}) \;\geq\; \mathrm{OPT} - \varepsilon - 2\eta$$

with probability at least $1 - \delta$, where the expensive evaluation complexity decomposes into a calibration term scaling as $\Theta((\mathrm{comp}(\mathcal{G}) + \log(1/\delta))/\varepsilon^2)$ and a validation term scaling as $O(k \log(k/\delta)/\varepsilon^2)$. This decomposition makes explicit a tradeoff often left implicit in practice: increasing the screening aggressiveness (smaller k) reduces validation cost but increases the risk that calibration error causes the optimum to be screened out.

Finally, we show that the $\Theta(1/\varepsilon^2)$ dependence on the desired accuracy is information-theoretically necessary: even when $\eta = 0$ and the proxy is perfectly calibratable within $\mathcal{G}$, one cannot in general beat the canonical rate dictated by hypothesis testing and best-arm identification. Thus, under bounded distortion, calibration can improve constant factors and reduce dependence on n via screening, but it cannot circumvent the fundamental sampling barrier imposed by noisy expensive evaluations. In this sense, our results delineate what proxies can and cannot buy: they can reduce the *search* burden by guiding where to spend expensive evaluations, but they cannot eliminate the *estimation* burden required to certify near-optimality.

2 Related Work

We group prior work by the role played by inexpensive signals in reducing expensive evaluations, and we highlight where existing analyses leave gaps that our oracle formalization and tight bounds are designed to fill.

Weight-sharing and one-shot NAS. A major line of work seeks to amortize training cost by constructing a *one-shot* supernet whose shared parameters are used to score sub-architectures, thereby inducing a cheap proxy for the fully trained objective. This idea appears in a wide range of methods, including differentiable NAS and its variants (e.g., DARTS and follow-ups), as well as sampling-based one-shot procedures. In differentiable approaches (DARTS, DARTS+, and subsequent stabilizations such as Fair-DARTS), the search objective is relaxed so that architecture parameters and shared weights are optimized jointly; the relaxed solution is then discretized to produce a final architecture. In the language of our model, the supernet-derived validation accuracy (or a related score) is a proxy $P(a)$ that can be queried for many a at low marginal cost. Empirical work has documented that weight-sharing proxies can be informative in some regimes but can also exhibit significant bias and instability, including sensitivity to optimization hyperparameters, discretization effects, and mismatches between proxy ranking and the ranking under full training. These observations motivate treating P as an arbitrary signal that may or may not reliably preserve the ordering induced by F, rather than as a faithful estimator of F.

Surrogate predictors and performance modeling. Another prominent approach trains a learned predictor that maps an architecture encoding to a predicted performance, using a dataset of previously evaluated architectures. This includes classical surrogate modeling (e.g., Gaussian processes and random forests in Bayesian optimization for structured spaces) as well as modern neural predictors. Such predictors may be used to propose candidates directly, to rank a large set and then validate a shortlist, or to guide acquisition functions in sequential design. Conceptually, a learned predictor is itself a proxy oracle O_P once trained. However, the predictor is typically learned from noisy, budget-limited data, and its generalization depends on the hypothesis class and the sampling scheme. Our abstraction separates these issues: we allow *any* proxy oracle, but we require expensive evaluations to learn a *calibration map* from proxy values to the true scale, and we quantify how many such evaluations are necessary as a function of the complexity of the calibration class. This differs from work that treats the predictor as the final decision rule without explicit end-to-end guarantees relative to OPT.

Multi-fidelity HPO and early-stopping proxies. Multi-fidelity methods reduce cost by evaluating configurations at lower fidelities, such as fewer epochs, smaller models, reduced data, or coarser resolutions, and adaptively allocating more budget to promising candidates. Representative algorithms include successive halving and Hyperband, as well as Bayesian optimization variants that incorporate fidelity as an input. These methods can be viewed as producing a proxy $P(a)$ derived from partial training information, often with a well-defined computational cost that is lower than full training. In practice, the critical question is how reliably low-fidelity performance predicts full-fidelity performance, and under what conditions aggressive early-stopping can discard the eventual best configuration. Our framework does not model fidelity cost explicitly; instead, it isolates the dominant cost—full evaluations through O_F—and treats proxies as inexpensive. This allows us to state worst-case limits (proxy-only impossibility) and then recover positive results under a minimal linkage (bounded-distortion calibration) that captures when low-fidelity signals can be trusted up to a distortion level η.

Learning-curve extrapolation. A closely related literature attempts to predict final performance from early learning curves, using parametric curve families, Bayesian models, or neural sequence predictors. These methods again provide a cheap or moderately cheap proxy derived from partial observations, sometimes with uncertainty estimates used to terminate unpromising runs. From our perspective, learning-curve extrapolation is one instantiation of a proxy oracle: $P(a)$ may be a vector of early metrics, and g is the extrapolation rule mapping that proxy to a predicted final score. The bounded-distortion view clarifies what must be true for such extrapolations to support guarantees: there must exist a calibration map within a known class that uniformly approximates the truth up to η across the architectures under consideration.

Known proxy pathologies and ranking unreliability. A recurring empirical finding in NAS is that proxies can induce substantial ranking error: high proxy scores may correspond to architectures that underperform when trained independently, and proxy rankings can change across seeds, datasets, or training protocols. In weight-sharing, this is often attributed to interference among sub-architectures, optimization mismatch between supernet and stand-alone training, and implicit regularization effects. In early-stopping, it arises from non-uniform convergence rates and regime changes late in training. Predictor-based proxies can fail due to covariate shift between the training set of architectures and the queried region, or due to model misspecification. These pathologies align with the logical content of our impossibility result: without an explicit assumption linking P to F, it is possible to construct instances where proxies provide no usable information about the optimizer, and thus no algorithm can guarantee even a weak approximation uniformly over instances.

Positioning and contribution relative to prior analyses. Prior work often provides (i) empirical comparisons of proxies, (ii) asymptotic convergence statements under smoothness or realizability assumptions in continuous relaxations, or (iii) regret-type guarantees in Bayesian optimization settings under strong priors and kernel assumptions. Our objective is different: we seek finite-sample, instance-uniform guarantees for a finite candidate set under a budget of noisy expensive evaluations, while allowing arbitrary proxy mechanisms. The key conceptual move is to treat proxies as *screening devices* whose utility must be justified by a testable linkage to the truth. The bounded-distortion calibration condition serves as such a linkage: it is weak enough to encompass weight-sharing, early-stopping, and predictor-based proxies (by placing the burden on the existence of an appropriate g), yet structured enough to yield sharp rates governed by the complexity of $\mathcal{G}$. Under this condition, we obtain end-to-end guarantees with explicit dependence on ε, δ, and $\mathrm{comp}(\mathcal{G})$, and we complement them with matching lower bounds showing that $\Theta(1/\varepsilon^2)$ expensive evaluations are unavoidable even when proxies are perfectly calibratable. In this sense, our results formalize what is implicit across much of the NAS literature: proxies can reduce *where* we spend expensive evaluations, but they cannot remove the fundamental sampling requirements imposed by noisy true evaluation.

3 Formal Model

We formalize neural architecture search (NAS) as an optimization problem over a *finite* candidate set with two sources of information: an inexpensive proxy signal and an expensive noisy evaluation of the true objective. The finiteness assumption is not merely for convenience; it isolates the role of statistical noise and information constraints from issues of continuous optimization, and it captures the common experimental regime in which one searches within a discretized cell space or a benchmarked search space.

Architectures and true objective. Let $\mathcal{A}$ denote a finite set of architectures with $|\mathcal{A}| = n \geq 2$. The (unknown) true objective is a function

$$F : \mathcal{A} \to [0, 1],$$

where $F(a)$ represents the performance of architecture a under a fixed training-and-evaluation protocol (e.g., validation accuracy after full training). We denote the optimal value by

$$\mathrm{OPT} := \max_{a \in \mathcal{A}} F(a).$$

The scaling to $[0, 1]$ is without loss of generality and simplifies concentration statements.

True-evaluation oracle and noise model. Architectures are not observed through F directly; instead, an expensive evaluation produces a noisy observation. Formally, we assume oracle access to O_F such that a query at a returns an independent sample

$$\widetilde{F}(a) \; = \; F(a) + \xi,$$

where the noise ξ is mean-zero and σ-sub-Gaussian (uniformly over a). This captures the dominant randomness in training runs (initialization, minibatch order, stochastic regularization, hardware nondeterminism), as well as measurement noise in the evaluation procedure. If one averages r independent runs, the effective noise parameter scales as $\sigma/\sqrt{r}$; we freely allow such replication, noting that it consumes r units of expensive budget.

Proxy oracle. In addition, we assume access to a cheap proxy oracle O_P which, on query $a \in \mathcal{A}$, returns a proxy value

$$P(a) \in \mathcal{Y},$$

for some proxy range $\mathcal{Y}$ (e.g., $\mathcal{Y} = [0, 1]$ for accuracies, or $\mathbb{R}$ for scores). We treat O_P as deterministic in this model: once the proxy mechanism is fixed (including any training of a supernet or a predictor), the value $P(a)$ is fixed. This abstraction separates the statistical difficulty arising from noisy *true* evaluations from the algorithmic role played by proxies; extensions to noisy proxies can be handled by introducing an additional noise term in O_P and adjusting concentration arguments, but the impossibility and lower-bound phenomena we study are already present in the deterministic-proxy setting.

Algorithms and resource constraints. An algorithm Alg may be randomized and adaptive. It interacts with O_P and O_F sequentially: at each round it may query either oracle on an architecture of its choice, based on all past observations and internal randomness. The output is an architecture $\hat{a} \in \mathcal{A}$.

The dominant cost is the number of calls to O_F. We therefore impose a budget B on O_F queries (counting repetitions separately). Proxy queries to O_P are assumed free or separately budgeted; in particular, Alg may query O_P for all architectures in $\mathcal{A}$ if desired. Computationally, we assume that processing proxy values is feasible in time polynomial in the number of proxy queries and in n (e.g., sorting scores, fitting a low-complexity calibration map), whereas O_F queries represent expensive training runs.

Approximation notions. We evaluate Alg through its achieved true performance $F(\hat{a})$ relative to OPT under a failure probability parameter $\delta \in (0, 1)$. The primary notion used in our upper bounds is an *additive* approximation guarantee: for $\varepsilon > 0$, we say that Alg is (ε, δ)-accurate if

$$\Pr\bigl(F(\hat{a}) \geq \mathrm{OPT} - \varepsilon\bigr) \;\geq\; 1 - \delta.$$

We also discuss a *multiplicative* approximation factor $\alpha \in (0, 1]$, requiring

$$\mathbb{E}\bigl[F(\hat{a})\bigr] \;\geq\; \alpha\, \mathrm{OPT},$$

or the corresponding high-probability variant. In our setting, additive guarantees are the more meaningful baseline because OPT may be arbitrarily small (e.g., for difficult tasks or adversarial instances), making multiplicative statements vacuous unless one further assumes $\mathrm{OPT} \geq \tau > 0$. Our impossibility result is stated multiplicatively to emphasize that even a weak constant-factor approximation cannot be achieved in the absence of assumptions linking P to F.

How common NAS signals instantiate the proxy. The proxy mapping P is intended to encompass the inexpensive signals commonly used to reduce the number of full training runs:

- *Weight-sharing / one-shot scores:* after training a supernet, one evaluates a sub-architecture a using inherited weights, producing a cheap validation accuracy or a related score. This defines $P(a)$ as the supernet-derived metric.

- *Early-stopping and multi-fidelity metrics:* one trains a for a small number of epochs, on a subset of data, or at reduced resolution, and uses the resulting validation accuracy or loss as $P(a)$.

- *Surrogate predictors:* one trains a regression model on previously evaluated architectures, and then uses the predictor output as $P(a)$ for new architectures.

In each case, our model regards $P(a)$ as an arbitrary real-valued signal that may or may not preserve the ordering induced by F. The point of the subsequent sections is to make precise what can be concluded from proxy access alone (nothing nontrivial in the worst case), and what can be concluded once we posit a minimal, testable relationship between P and F and allow a limited number of noisy true evaluations to learn that relationship.

4 Impossibility of Proxy-Only NAS

We now isolate a basic information-theoretic obstruction: without *any* assumption linking the proxy P to the true objective F, oracle access to O_P alone is insufficient to guarantee nontrivial optimization performance. The obstruction is not computational; it holds even for unbounded computation and fully adaptive, randomized procedures. The crux is an indistinguishability construction in which two different objectives induce *identical* proxy observations.

Proxy-only algorithms. Fix any (possibly randomized and adaptive) algorithm Alg that is allowed to query O_P arbitrarily many times, and then outputs an architecture $\hat{a} \in \mathcal{A}$. Since O_P is deterministic in our model, and Alg has no access to O_F, the entire transcript of interaction with O_P (and hence the output distribution of $\hat{a}$) is a deterministic function of the proxy mapping P and the internal randomness of Alg. In particular, if two problem instances share the same proxy mapping P, then Alg induces the same distribution over outputs on both instances.

Indistinguishability construction. We prove Theorem 1 by exhibiting two instances that share the same proxy but have different maximizers of F. Let $a^{(0)}, a^{(1)} \in \mathcal{A}$ be two distinct architectures. Define a proxy mapping that carries no information,

$$P(a) \equiv 0 \qquad \text{for all } a \in \mathcal{A},$$

so that *every* proxy query returns the same value. Next define two objective functions $F_0, F_1 : \mathcal{A} \to [0,1]$ by

$$F_0(a^{(0)}) = 1, \quad F_0(a^{(1)}) = 0, \quad F_0(a) = 0 \;\; \forall a \notin \{a^{(0)}, a^{(1)}\},$$

and symmetrically,

$$F_1(a^{(0)}) = 0, \quad F_1(a^{(1)}) = 1, \quad F_1(a) = 0 \;\; \forall a \notin \{a^{(0)}, a^{(1)}\}.$$

Both instances have $\mathrm{OPT} = 1$. However, the identity of the unique optimizer is swapped between F_0 and F_1 while the proxy mapping remains identical.

Let q denote the output distribution of Alg on this proxy mapping P; that is, $q(a) = \Pr(\hat{a} = a)$, where the probability is over the internal randomness of Alg (there is no other randomness in the proxy-only setting). Since P is the same under both instances, q is the same under F_0 and F_1. The expected true performance under each instance is therefore

$$\mathbb{E}[F_0(\hat{a})] = q(a^{(0)}), \qquad \mathbb{E}[F_1(\hat{a})] = q(a^{(1)}).$$

Averaging over the two instances yields

$$\frac{1}{2}\Big(\mathbb{E}[F_0(\hat{a})] + \mathbb{E}[F_1(\hat{a})]\Big) = \frac{q(a^{(0)}) + q(a^{(1)})}{2} \le \frac{1}{2},$$

since $q(a^{(0)}) + q(a^{(1)}) \le 1$. Consequently, at least one of the two instances must satisfy $\mathbb{E}[F_j(\hat{a})] \le 1/2$, proving that no proxy-only algorithm can guarantee $\mathbb{E}[F(\hat{a})] \ge \alpha\,\mathrm{OPT}$ uniformly over instances for any $\alpha > 1/2$.

No nontrivial additive approximation without linkage. The same indistinguishability phenomenon rules out additive guarantees. Indeed, fix any $\varepsilon < 1$. In the above construction, exactly one architecture attains value 1 and all others attain value 0. Unless Alg outputs the optimizer with probability at least $1 - \delta$, it fails the (ε, δ)-accuracy requirement $\Pr(F(\hat{a}) \ge 1 - \varepsilon) \ge 1 - \delta$. But because the proxy transcript is identical under F_0 and F_1, any event of the form "$\hat{a} = a^{(0)}$" has the same probability under both instances, and therefore Alg cannot simultaneously assign high probability to the optimizer in both cases. Thus, without assumptions linking P to F, even extremely weak high-probability guarantees are impossible.

Tightness of the $1/2$ barrier as a guessing limit. The constant $1/2$ should be understood as the unavoidable loss incurred when the proxy collapses two competing hypotheses about which architecture is optimal. In the two-instance construction, the best one can do (given only P) is essentially to guess between $a^{(0)}$ and $a^{(1)}$; any strategy induces probabilities $q(a^{(0)}), q(a^{(1)})$, and one of the two worlds penalizes the less-likely guess. This is the standard form of a lower bound by indistinguishability: whenever an information source makes two instances observationally identical, no algorithm using only that source can reliably choose the correct maximizer across both instances.

Implications for proxy-driven ranking methods. Many NAS procedures are, at their core, *proxy-driven rankers*: they compute a score $P(a)$ (e.g., a weight-sharing validation accuracy, an early-stop metric, or a predictor output), then return an architecture that is approximately maximal under this score. If such a procedure performs *no* expensive true evaluations during search, then it falls within the proxy-only model and inherits the above impossibility: there exist search spaces and training protocols for which the proxy ranking is completely uninformative about the true ranking, and the selected architecture can be arbitrarily suboptimal in expectation and with nontrivial probability.

This observation applies directly to one-shot and differentiable methods (e.g., DARTS-style relaxations) when interpreted as mechanisms that produce a final discrete architecture purely by optimizing a proxy objective induced by shared weights or continuous relaxation. Our lower bound does not assert that these methods fail on typical benchmarks; rather, it formalizes a worst-case limitation: absent explicit conditions ensuring that the proxy preserves enough information about F, *no* proxy-only procedure can provide guarantees. In particular, empirical success of proxy-based NAS must rely on additional structure (sometimes implicit) connecting P and F, or on the use of at least some expensive evaluations to correct proxy errors.

Motivation for linkage assumptions. The conclusion is that proxies are not, by themselves, a source of optimization guarantees; they must be justified by a verifiable relationship to the true objective. This motivates our next step: we will posit minimal proxy-to-true linkage conditions under which a small number of noisy true evaluations suffices to *calibrate* the proxy and recover provable near-optimal selection.

5 Minimal Proxy-to-True Linkage Conditions

Theorem 1 shows that, in the absence of any relationship between the cheap signal P and the true objective F, proxy access alone cannot support nontrivial guarantees. We therefore require an explicit *linkage condition.* Our goal is to articulate a condition that is (i) weak enough to plausibly hold for realistic proxies, (ii) strong enough to enable provable near-optimal selection with a small number of expensive evaluations, and (iii) at least partially *testable* from finite calibration data.

Bounded-distortion calibration as a minimal linkage. We adopt a calibration viewpoint: the proxy $P(a)$ need not be on the same scale as $F(a)$, but it should be possible to map proxy values to predicted true values via an unknown function g, up to a bounded residual. Formally, we require that F is well-approximated by a calibrated proxy.

Definition 5.1 (Bounded-distortion calibration)**.** Fix a known function class $\mathcal{G}$ of maps $g : \mathcal{Y} \to [0, 1]$ and a distortion level $\eta \geq 0$. We say that a pair (F, P) satisfies $(\mathcal{G}, \eta)$-bounded distortion if there exists an (unknown) $g \in \mathcal{G}$ such that for all architectures $a \in \mathcal{A}$,

$$\big|F(a) - g(P(a))\big| \leq \eta.$$

This condition separates the proxy-to-true relationship into two components. First, $\mathcal{G}$ encodes *structural beliefs* about how proxy values translate to true performance (e.g., monotonicity, smoothness, or parametric form). Second, η measures the *irreducible mismatch* between the best such calibration and the true objective. The definition is deliberately agnostic about the origin of the proxy: P may arise from weight-sharing, early stopping, learned predictors, or any other cheap heuristic. If η is small, then P is informative after calibration; if η is large, then no method can hope to recover OPT closely from P alone, and our final bounds will explicitly reflect this.

Two aspects merit emphasis. (a) The calibration function g is *unknown*, so we must learn it from a small set of expensive evaluations; thus $\mathcal{G}$ cannot be arbitrarily rich if we want generalization from few samples. (b) The condition is *uniform over $a \in \mathcal{A}$*; this is the form most convenient for worst-case optimization guarantees. In applications one may also consider distributional variants (e.g., the inequality holding for $a \sim D$), but we focus on the uniform statement to keep the downstream selection argument simple.

Why a function class is necessary. If we allowed $\mathcal{G}$ to contain all functions $\mathcal{Y} \to [0,1]$, then the condition would become vacuous with $\eta = 0$ whenever P is injective (one could fit g to interpolate F). Such a linkage would not be learnable from limited calibration data: empirical fitting could overfit proxy values without controlling prediction error on unseen architectures. The role of $\mathrm{comp}(\mathcal{G})$ is precisely to quantify how many true evaluations are required to identify a calibrator with small uniform error, which then translates into preservation of near-optimal candidates during proxy screening.

Structured alternatives for $\mathcal{G}$. Definition 5.1 is a wrapper: different proxy mechanisms motivate different choices of $\mathcal{G}$. We record several common candidates.

(i) Monotone calibration. A basic assumption is that larger proxy values should not indicate systematically worse true performance after calibration. This corresponds to taking $\mathcal{G}$ to be the class of nondecreasing functions on $\mathcal{Y}$ (possibly also bounded in $[0,1]$). Monotonicity is particularly appropriate when $P(a)$ is itself an accuracy-like quantity (early-stop validation accuracy, predictor score trained to be order-preserving, etc.). It also matches the practical use of isotonic regression, which yields an ERM procedure with favorable finite-sample behavior.

(ii) Lipschitz (or smooth) calibration. When we believe that small changes in the proxy cannot correspond to arbitrarily large changes in F, we may restrict to L-Lipschitz functions:

$$|g(p) - g(p')| \le L|p - p'| \qquad \forall p, p' \in \mathcal{Y}.$$

Such a condition can be interpreted as a stability requirement for the proxy: it rules out adversarial "spikes" in the proxy-to-true map that would make calibration sample-inefficient. Lipschitzness also helps justify that screening by $\hat{g}(P(a))$ is robust to small proxy noise or discretization of $\mathcal{Y}$.

(iii) Low-complexity parametric calibration ("scaling laws"). In some regimes, proxies arise from partial training curves, dataset-size extrapolation, or compute-allocation rules. In such settings one often posits a parametric relationship g_θ (e.g., a linear map, logistic link, or a power-law/exponential saturation curve) and sets $\mathcal{G} = \{g_\theta : \theta \in \Theta\}$ for a low-dimensional parameter space. The benefit is that $\mathrm{comp}(\mathcal{G})$ can scale with $\dim(\theta)$ rather than with a nonparametric complexity, yielding sharper calibration sample requirements when the model is well-specified. The drawback is potential misspecification, which is then absorbed into η.

Testability and diagnostics. A linkage condition is only useful if we can obtain some evidence that it holds. While the universal quantifier over $a \in \mathcal{A}$ prevents perfect verification from finite data, bounded-distortion calibration admits practical diagnostics.

Given a calibration set $S = \{a_1, \ldots, a_m\}$ with proxy–truth pairs $(P(a_i), \widetilde{F}(a_i))$, we can fit $\hat{g} \in \mathcal{G}$ and examine residuals

$$R_i := \bigl|\widetilde{F}(a_i) - \hat{g}(P(a_i))\bigr|.$$

Because $\widetilde{F}(a) = F(a) + \xi$ with mean-zero sub-Gaussian ξ, the residuals combine both distortion and evaluation noise. Nonetheless, standard concentration arguments allow us to translate empirical residual quantiles on a held-out test set into bounds on the *population* distortion on the proxy distribution (after correcting for noise), thereby producing an operational estimate of whether η is plausibly small. Concretely, if even the median (or a high quantile) of R_i remains large relative to the target accuracy, then no algorithm can reasonably expect to achieve an ε-level guarantee via that proxy without substantially increasing true-evaluation budget.

This is in contrast with weaker, harder-to-test notions such as "P is correlated with F" or "the top-k under P contains the optimum", which can hold or fail in ways that are difficult to detect from limited calibration data and do not yield stable uniform guarantees. By phrasing the linkage as approximation by a low-complexity calibrator plus a distortion term, we obtain a condition that (i) aligns with how proxies are used (ranking after calibration), (ii) supports uniform convergence analyses through $\mathrm{comp}(\mathcal{G})$, and (iii) admits concrete residual-based diagnostics.

In the next section we exploit this linkage: we learn $\hat{g}$ from few expensive evaluations, use $\hat{g} \circ P$ to screen the search space, and then spend the remaining budget to validate only a small candidate set.

Algorithmic template. We now describe a simple procedure, *Calibrate-Then-Select*, that uses a small number of expensive evaluations to learn a calibration map and then leverages the proxy to reduce the search to a small candidate set. The algorithm has two logically distinct phases: (i) *calibration*, where we estimate a proxy-to-true map $\hat{g} \in \mathcal{G}$ from a limited set of architectures with true evaluations, and (ii) *selection*, where we use $\hat{g} \circ P$ to screen architectures and then spend the remaining expensive budget only on a small retained set.

Budgeting and the role of repeated evaluations. We treat the number of calls to O_F as the dominant cost. In the calibration phase, we choose a calibration set size m and (optionally) a replication factor r and form averaged observations

$$t_i := \frac{1}{r}\sum_{j=1}^{r} \widetilde{F}^{(j)}(a_i),$$

so that t_i is a mean-zero sub-Gaussian perturbation of $F(a_i)$ with variance proxy scaling as σ^2/r. This cleanly separates two sources of error in later bounds: statistical error in learning g (controlled by m and $\mathrm{comp}(\mathcal{G})$) and evaluation noise (controlled by r). In practice we may take $r = 1$ and absorb the noise level into constants, but we keep r explicit since it is sometimes preferable to average a small number of runs per calibration point rather than increase m.

Calibration set construction. The distribution D used to pick $S = \{a_1, \ldots, a_m\}$ is a design choice. Uniform sampling over $\mathcal{A}$ is conceptually simplest, but it can be statistically wasteful if proxy values are highly imbalanced. A more robust approach is to first compute proxy values for many architectures (possibly all of $\mathcal{A}$), partition architectures into quantile bins according to $P(a)$, and then sample roughly uniformly across bins. This stratification ensures that $\hat{g}$ is trained on a proxy range relevant to screening, and it mitigates the failure mode in which the calibration data cover only a narrow region of $\mathcal{Y}$. Because proxy queries are cheap, we may regard such stratification as essentially free.

Fitting the calibrator. Given calibration data $\{(p_i, t_i)\}_{i=1}^{m}$ with $p_i = P(a_i)$, we fit $\hat{g} \in \mathcal{G}$ by empirical risk minimization with squared loss (or absolute loss),

$$\hat{g} \in \arg\min_{g \in \mathcal{G}} \sum_{i=1}^{m} \big(t_i - g(p_i)\big)^2.$$

When $\mathcal{G}$ is the class of nondecreasing functions, isotonic regression yields an exact ERM solution in $O(m \log m)$ time. For parametric classes, standard regression suffices. We emphasize that we do not require $\hat{g}$ to be correct pointwise on all of $\mathcal{Y}$; rather, we require that $\hat{g}(P(a))$ approximate $g(P(a))$ on the proxy values actually realized by architectures under consideration, which is precisely what uniform convergence over $\mathcal{G}$ (in an appropriate metric) will provide in the next section.

Proxy-based scoring and screening. After fitting $\hat{g}$, we compute for each $a \in \mathcal{A}$ the proxy $p(a) = P(a)$ and a predicted true score

$$s(a) \;:=\; \hat{g}\big(p(a)\big).$$

We then form a candidate set C consisting of the top-k architectures under $s(a)$ (breaking ties arbitrarily). Screening is the step that converts cheap proxy access into a reduction of the search space. The intended effect is that if $\hat{g}$ is accurate enough and the distortion is small enough, then C contains at least one near-optimal architecture (and, under a margin condition, contains the optimal one). Importantly, screening makes no additional calls to O_F; its cost is dominated by $O(n)$ proxy queries and a sort/selection step.

Validation as best-arm identification on the candidate set. The final step is to identify the best architecture within C using the remaining expensive budget. The simplest validator allocates an equal number of evaluations to each $a \in C$ and outputs the empirical maximizer. This yields a direct concentration analysis with total sample complexity scaling as $O(k/\varepsilon^2)$ for additive error ε at fixed confidence. When k is moderately large or gaps are heterogeneous, we may instead run a standard adaptive routine (successive halving, LUCB, or any best-arm identification algorithm) on the arms indexed by C. Such adaptive validation can reduce constant factors and often improves empirical efficiency while retaining worst-case guarantees of the same order.

Optional top-k verification for robustness. A practical concern is that screening may be brittle if $\hat{g}$ extrapolates poorly outside the proxy range covered in calibration or if the proxy has heavy-tailed failure modes concentrated in a small region. To address this, we may incorporate a lightweight verification subroutine: before committing the full validation budget, we evaluate a small number of architectures near the screening threshold (e.g., ranks $k, k+1, \ldots, k+u$ under $s(a)$) with a few true-evaluation calls each. If these spot checks reveal systematic underestimation beyond what calibration noise predicts, we can enlarge k, resample additional calibration points targeted to the suspicious proxy region, or revert to a more conservative validator. This verification does not change the asymptotic sample complexity in our guarantees, but it makes the method less sensitive to finite-sample pathologies.

Parameter selection and interface with the analysis. The remaining design degrees of freedom are m, k, and the split of the O_F budget between calibration and validation. In the next section we will set m so that $\hat{g}$ achieves a uniform (or distribution-dependent) approximation error of order ε over the relevant proxy values, as dictated by $\mathrm{comp}(\mathcal{G})$ and δ. We then choose k to balance two competing effects: larger k decreases the risk that screening discards near-optimal architectures, but increases the validation cost. The resulting guarantee will explicitly decompose into (i) calibration error, (ii) distortion, and (iii) validation error, thereby clarifying how proxy quality and function-class complexity translate into expensive-evaluation requirements.

6 Upper bounds under bounded distortion

Setup and error decomposition. We analyze CALIBRATE-THEN-SELECT under the $(\mathcal{G}, \eta)$-bounded distortion condition (Definition 2): there exists an unknown $g \in \mathcal{G}$ such that for all $a \in \mathcal{A}$,

$$|F(a) - g(P(a))| \leq \eta.$$

The analysis proceeds by isolating three contributions to suboptimality: (i) *calibration estimation error* $\Delta_{\mathrm{cal}} := \sup_{a \in \mathcal{A}} |\hat{g}(P(a)) - g(P(a))|$ (or a distribution-restricted variant), (ii) *proxy distortion* η, and (iii) *validation error* from noisy comparisons within the screened candidate set C.

Controlling noise in calibration via replication. Recall that we form, for each calibration architecture a_i, an averaged response

$$t_i = \frac{1}{r} \sum_{j=1}^{r} \widetilde{F}^{(j)}(a_i) = F(a_i) + \zeta_i,$$

where ζ_i is mean-zero and $\sigma/\sqrt{r}$-sub-Gaussian. Thus all calibration generalization bounds that scale as $1/\sqrt{m}$ in the noiseless case lift directly to a dependence of order $1/\sqrt{mr}$ in the number of *oracle calls* to O_F used for calibration. In particular, taking $r = 1$ is always valid, while taking $r > 1$ improves constants when evaluation noise dominates.

Finite-class calibration bounds (union bound). When $\mathcal{G}$ is finite, we can bound calibration error with a direct uniform concentration argument. Let D denote the sampling distribution used to pick calibration architectures $a_i \sim D$, and let $p_i = P(a_i)$. Consider squared-loss ERM

$$\hat{g} \in \arg\min_{g \in \mathcal{G}} \sum_{i=1}^{m} (t_i - g(p_i))^2.$$

Under bounded outputs (we may clip t_i and $g(p)$ to $[0,1]$ without worsening the guarantee), standard sub-Gaussian concentration and a union bound yield that with probability at least $1-\delta$,

$$\sup_{g\in\mathcal{G}}\Big|\,\mathbb{E}_{a\sim D}\big[(t-g(P(a)))^2\big]-\frac{1}{m}\sum_{i=1}^{m}(t_i-g(p_i))^2\Big| \;\le\; O\left(\sqrt{\frac{\log|\mathcal{G}|+\log(1/\delta)}{m}}\right),$$

with the hidden constant scaling linearly in $\sigma/\sqrt{r}$ when we express the bound in terms of oracle calls. Consequently, choosing

$$m=\Theta\left(\frac{\log|\mathcal{G}|+\log(1/\delta)}{\varepsilon^2}\right) \quad\text{and}\quad r=\Theta(1)$$

suffices to ensure that the learned $\hat{g}$ predicts $g(P(a))$ to accuracy $O(\varepsilon)$ on proxy values realized under D; when we require a uniform bound over all $a\in\mathcal{A}$, we may take D uniform over $\mathcal{A}$ (or use stratified designs that effectively approximate this uniform control on the screened region).

Capacity-based bounds (pseudo-dimension, coverings). For infinite classes, we replace $\log|\mathcal{G}|$ by a suitable complexity term. Concretely, suppose $\mathcal{G}$ admits a uniform convergence guarantee of the form

$$\sup_{g\in\mathcal{G}}\Big|\,\mathbb{E}\ell(g)-\frac{1}{m}\sum_{i=1}^{m}\ell_i(g)\Big| \;\le\; O\left(\sqrt{\frac{\mathrm{comp}(\mathcal{G})+\log(1/\delta)}{m}}\right),$$

for an appropriate loss ℓ (e.g., squared loss) and sample-dependent losses ℓ_i. One may instantiate $\mathrm{comp}(\mathcal{G})$ as a pseudo-dimension term (for parametric regression), a VC-subgraph term, or a covering-number/Rademacher complexity bound (for Lipschitz or shape-constrained classes). For example, for isotonic regression on a totally ordered proxy space with bounded range, one may take $\mathrm{comp}(\mathcal{G})=O(1)$ up to log factors, whereas for a d-dimensional parametric family one typically has $\mathrm{comp}(\mathcal{G})=\tilde{O}(d)$. In all cases, the calibration sample size choice

$$m=\Theta\left(\frac{\mathrm{comp}(\mathcal{G})+\log(1/\delta)}{\varepsilon^2}\right)$$

drives $\Delta_{\mathrm{cal}}=O(\varepsilon)$ on the proxy region covered by the calibration design.

From calibration error to screening correctness. Assume we have obtained an event $\mathcal{E}_{\mathrm{cal}}$ on which

$$|\hat{g}(P(a))-g(P(a))|\le\Delta_{\mathrm{cal}} \qquad \text{for all } a\in\mathcal{A} \text{ (or for all } a \text{ considered in screening).}$$

Let $a^\star\in\arg\max_a F(a)$, and let $\bar{a}\in\arg\max_a s(a)$ be the maximizer of the predicted score $s(a)=\hat{g}(P(a))$ (note $\bar{a}\in C$ for any $k\ge 1$). On $\mathcal{E}_{\mathrm{cal}}$ we have

$$\hat{g}(P(\bar{a}))\ge\hat{g}(P(a^\star)) \;\Rightarrow\; g(P(\bar{a}))\ge g(P(a^\star))-2\Delta_{\mathrm{cal}}.$$

Using bounded distortion twice yields

$$F(\bar{a}) \geq g(P(\bar{a})) - \eta \geq g(P(a^\star)) - 2\Delta_{\text{cal}} - \eta \geq F(a^\star) - 2\Delta_{\text{cal}} - 2\eta.$$

Thus screening alone ensures that the *best predicted* architecture is already near-optimal up to $2\Delta_{\text{cal}} + 2\eta$. Introducing $k > 1$ does not change this worst-case bound, but it enables cheaper or more reliable downstream validation (e.g., when we wish to hedge against finite-sample miscalibration outside the calibration support).

Validation bounds on the candidate set. Condition on any fixed candidate set C with $|C| = k$. If we allocate T independent calls to O_F per candidate and output the empirical maximizer, Hoeffding-type bounds for sub-Gaussian noise imply that with probability at least $1 - \delta$,

$$F(\hat{a}) \geq \max_{a \in C} F(a) - \varepsilon_{\text{val}} \qquad \text{provided} \qquad T = \Theta\left(\frac{\sigma^2 \log(k/\delta)}{\varepsilon_{\text{val}}^2}\right).$$

Equivalently, the *total* validation budget scales as $kT = O(k\sigma^2 \log(k/\delta)/\varepsilon_{\text{val}}^2)$. More adaptive validators (successive halving, LUCB) can improve constants and exploit gaps, but the worst-case dependence on $k/\varepsilon_{\text{val}}^2$ is unavoidable.

Putting the pieces together. Combining the screening inequality with validation, and choosing (for instance) $\Delta_{\text{cal}} \leq \varepsilon/2$ and $\varepsilon_{\text{val}} \leq \varepsilon/2$, we obtain the guarantee summarized in Theorem 3: with probability at least $1 - \delta$,

$$F(\hat{a}) \geq \text{OPT} - \varepsilon - 2\eta,$$

using

$$m = \Theta\left(\frac{\text{comp}(\mathcal{G}) + \log(1/\delta)}{\varepsilon^2}\right) \quad \text{calibration points and} \quad O\left(\frac{k \log(k/\delta)}{\varepsilon^2}\right) \quad \text{validation evaluations,}$$

up to factors depending on σ^2 and the replication choice r.

Robustness to misspecification and heavier tails. If the bounded distortion condition holds only approximately for the chosen class $\mathcal{G}$, the same argument yields a graceful degradation. Define the approximation error

$$\eta^\star := \inf_{g \in \mathcal{G}} \sup_{a \in \mathcal{A}} |F(a) - g(P(a))|.$$

Then, under identical calibration and validation conditions, we obtain

$$F(\hat{a}) \geq \text{OPT} - \varepsilon - 2\eta^\star$$

(with $\eta^\star$ replacing η), since all steps use only triangle inequalities around $g(P(a))$. Finally, while our standing assumption is sub-Gaussian evaluation noise, the procedure can be made robust to heavier-tailed ξ by replacing simple averaging by median-of-means (both in calibration and validation), at the cost of additional logarithmic factors; the structural dependence on $\mathrm{comp}(\mathcal{G})$, k, and $1/\varepsilon^2$ remains the same.

7 Lower bounds: necessity of $\Omega(1/\varepsilon^2)$ true evaluations

Why lower bounds are unavoidable in our oracle model. The bounded-distortion condition postulates the existence of a calibration map $g \in \mathcal{G}$ for which $F(a) = g(P(a)) \pm \eta$. This linkage is sufficient to make proxy-guided search meaningful, but it does not eliminate the need to *sample* O_F: both the identity of the maximizer and the unknown calibration map must still be inferred from noisy observations. We formalize this by reductions to classical hypothesis testing (coin-bias) and best-arm identification, which together yield tight worst-case dependences on ε, δ, and (when g is unknown) the complexity of $\mathcal{G}$.

A two-point coin-bias reduction (Theorem 4). We first isolate the intrinsic $\Omega(\log(1/\delta)/\varepsilon^2)$ dependence even in the simplest case $\eta = 0$. Fix two architectures $a_0, a_1 \in \mathcal{A}$ and set the proxy to be *uninformative* among them, e.g.,

$$P(a_0) = P(a_1) = p, \qquad P(a) = p_0 \text{ for } a \notin \{a_0, a_1\},$$

so that proxy information cannot distinguish a_0 from a_1. Let $\mathcal{G}$ contain at least two hypotheses g_+ and g_- satisfying $g_+(p) = \frac{1}{2} + \varepsilon$ and $g_-(p) = \frac{1}{2} - \varepsilon$ (and arbitrary values elsewhere in $[0, 1]$). Consider two instances indexed by $\theta \in \{+, -\}$ defined by

$$F_\theta(a_0) = g_\theta(P(a_0)), \qquad F_\theta(a_1) = g_{-\theta}(P(a_1)),$$

and $F_\theta(a) = 0$ for all other a. Then (F_θ, P) satisfies $(\mathcal{G}, 0)$-bounded distortion (indeed equality holds with $g = g_\theta$). Moreover, under $\theta = +$ the optimal architecture is a_0 and under $\theta = -$ it is a_1, and the optimality gap is exactly 2ε.

Now choose the evaluation oracle to return Bernoulli rewards $\widetilde{F}(a) \sim \mathrm{Bernoulli}(F_\theta(a))$, which is 1/2-sub-Gaussian and lies in $[0, 1]$. Any algorithm that outputs an ε-optimal architecture with probability $\geq 1 - \delta$ must, in particular, identify the correct maximizer among $\{a_0, a_1\}$ with error probability $\leq \delta$. But distinguishing the two hypotheses $\theta \in \{+, -\}$ reduces to distinguishing a coin of bias $\frac{1}{2} + \varepsilon$ from a coin of bias $\frac{1}{2} - \varepsilon$ given adaptively chosen samples from (at most) these two coins. Standard Le Cam/Bretagnolle–Huber inequalities imply that if the total number of samples from O_F is T, then the sum of type-I and type-II errors is bounded below by a constant unless

$$T = \Omega\left(\frac{\log(1/\delta)}{\varepsilon^2}\right),$$

since the KL divergence per sample between Bernoulli($\frac{1}{2}+\varepsilon$) and Bernoulli($\frac{1}{2}-\varepsilon$) is $\Theta(\varepsilon^2)$. This establishes Theorem 4: even with *perfect* bounded distortion ($\eta = 0$), one cannot beat the parametric ε^{-2} rate in the number of expensive evaluations.

Best-arm identification lower bound once screening has occurred. The same phenomenon persists when the proxy has already reduced the search space to a candidate set C of size k. Fix any $k \geq 2$ and consider instances where C contains one arm with mean $\frac{1}{2} + \varepsilon$ and $k - 1$ arms with mean $\frac{1}{2}$, with all proxies equal (or, equivalently, with g known and P perfectly informative about membership in C but not about the best element). Any algorithm that returns an ε-optimal element of C with probability $\geq 1 - \delta$ must, in the worst case, spend

$$\Omega\left(\frac{k\log(1/\delta)}{\varepsilon^2}\right)$$

total samples across the k arms; this follows from standard bandit lower bounds for fixed-confidence best-arm identification by embedding k-ary hypothesis testing instances with pairwise KL of order ε^2 and applying either Fano's inequality or change-of-measure arguments. Consequently, the validation phase dependence on k/ε^2 cannot be improved in the worst case by any proxy mechanism: at best, the proxy can reduce k.

Lower bounds reflecting complexity of $\mathcal{G}$ (calibration is information-theoretically costly). The previous constructions used only two candidate calibrators. More generally, when g is unknown and must be learned from calibration data, the number of *distinct plausible* calibration functions drives an additional information requirement. To make this precise in the simplest setting, suppose $\mathcal{G}$ is finite with $|\mathcal{G}| = M$ and contains a subset $\{g_1, \ldots, g_M\}$ such that for each $\ell \neq \ell'$ there exists a proxy value p (realized by some architecture) with

$$|g_\ell(p) - g_{\ell'}(p)| \geq 2\varepsilon.$$

We may construct M instances (F_ℓ, P) with $\eta = 0$ and $F_\ell(a) = g_\ell(P(a))$ such that (i) the proxy values are identical across instances, and (ii) the identity of the near-optimal architecture depends on ℓ. Any algorithm that achieves additive error ε must then effectively identify ℓ up to a small ambiguity class. By Fano's inequality, if the mutual information between the instance index

ℓ and the full transcript is $o(\log M)$, then the probability of recovering ℓ (hence of selecting an ε-optimal architecture) is bounded away from 1. Since each expensive oracle call contributes at most $O(\varepsilon^2)$ KL information when functions differ by $O(\varepsilon)$ on the queried proxy values, one obtains a lower bound of the form

$$\Omega\left(\frac{\log M + \log(1/\delta)}{\varepsilon^2}\right)$$

true evaluations in regimes where the algorithm must discriminate among M calibrators. This matches, up to constants and standard log factors, the $\mathrm{comp}(\mathcal{G})/\varepsilon^2$ calibration term in Theorem 3 when $\mathrm{comp}(\mathcal{G}) = \log|\mathcal{G}|$.

Interpretation. Taken together, these reductions show that our upper bounds are tight in the minimax sense: (i) even in the most favorable calibration regime ($\eta = 0$), ε^{-2} expensive samples are necessary to resolve noise; (ii) even with perfect screening to k candidates, $\Omega(k/\varepsilon^2)$ validation cost is unavoidable; and (iii) when g must be learned, the complexity of $\mathcal{G}$ necessarily enters the expensive sample count through a term of order $\mathrm{comp}(\mathcal{G})/\varepsilon^2$. Thus, proxy signals can reduce *search* but not the fundamental *statistical* cost of overcoming evaluation noise and calibrator uncertainty.

8 Proxy diagnostics (practical add-on)

Our guarantees hinge on a linkage between P and F—captured abstractly by the existence of $g \in \mathcal{G}$ with $|F(a) - g(P(a))| \leq \eta$—and on the ability of a finite calibration sample to learn a usable approximation $\hat{g}$. In practice, before spending a large expensive-evaluation budget on proxy-guided screening and validation, we should empirically *test* whether the proxy is behaving in a manner consistent with small distortion on the region of interest (typically the high-proxy tail), and we should propagate this diagnostic into conservative screening rules and budget allocation.

Held-out calibration tests and residual summaries. We recommend splitting the expensive calibration set into a fitting subset S_{fit} and a held-out subset S_{test}, with $\hat{g}$ trained only on S_{fit}. For each $a \in S_{\text{test}}$, let $p(a) = P(a)$ and let $\bar{t}(a)$ be the average of r independent calls to $\mathsf{O}_F(a)$, so that $\bar{t}(a) = F(a) + \bar{\xi}(a)$ with $\bar{\xi}(a)$ mean-zero and $\sigma/\sqrt{r}$-sub-Gaussian. Define residuals

$$e(a) \;=\; \left|\bar{t}(a) - \hat{g}(p(a))\right|.$$

We then report (i) the empirical mean and quantiles of $\{e(a) : a \in S_{\text{test}}\}$, (ii) the same quantities restricted to architectures whose proxies fall in the top q-quantile of $p(a)$ (to focus on the screening-relevant region), and (iii) a binned reliability plot over proxy bins, e.g. intervals of proxy values, showing $\hat{g}(p)$ versus the empirical mean of $\bar{t}$ in each bin. The latter often reveals regime changes (e.g. proxy saturating at high scores) that are invisible to global metrics.

While $e(a)$ is not exactly $|F(a)-\hat{g}(P(a))|$ because of noise, it is an upper envelope of that quantity up to the additive fluctuation $|\bar{\xi}(a)|$. Using sub-Gaussian concentration and a union bound over $|S_{\text{test}}|$, with probability at least $1-\delta$ we have

$$\max_{a\in S_{\text{test}}} |\bar{\xi}(a)| \;\le\; \frac{\sigma}{\sqrt{r}}\sqrt{2\log\frac{2|S_{\text{test}}|}{\delta}}.$$

Consequently,

$$\max_{a\in S_{\text{test}}} \big|F(a)-\hat{g}(P(a))\big| \;\le\; \max_{a\in S_{\text{test}}} e(a) \;+\; \frac{\sigma}{\sqrt{r}}\sqrt{2\log\frac{2|S_{\text{test}}|}{\delta}}.$$

This bound is conservative (it targets the maximum). In settings where we only need distributional control (e.g. on the proxy-induced distribution over architectures), a quantile-based diagnostic is typically more stable.

Quantile residual bounds and an empirical distortion estimate. Let $\hat{q}_\alpha$ denote the empirical α-quantile of $\{e(a) : a \in S_{\text{test}}\}$ for some $\alpha \in (0,1)$ (e.g. $\alpha = 0.9$ or 0.95). Interpreting $\hat{q}_\alpha$ as an estimate of the (α)-quantile of $|\bar{t}(a) - \hat{g}(P(a))|$ under the design distribution over S_{test}, we may form an empirical distortion certificate of the form

$$\hat{\eta}_\alpha \;=\; \hat{q}_\alpha \;+\; c\cdot\frac{\sigma}{\sqrt{r}}\sqrt{\log\frac{1}{\delta}},$$

for an absolute constant c, which aligns with the logic of Proposition 5: the first term captures systematic miscalibration and proxy mismatch, while the second accounts for evaluation noise and finite-sample uncertainty in the quantile. Operationally, if $\hat{\eta}_\alpha$ is large compared to the target ε (or large compared to the expected spread among top candidates), then a proxy-guided pipeline should be treated as unreliable without either improving the proxy, enlarging S_{fit}, or increasing r to reduce noise.

Conformal prediction intervals for screening with uncertainty. Residual summaries can be turned into *prediction intervals* for $F(a)$ given only $P(a)$. A simple split-conformal construction is as follows. Fit $\hat{g}$ on S_{fit}, compute conformity scores on S_{test},

$$u(a) = \big|\bar{t}(a) - \hat{g}(P(a))\big|,$$

and let $\hat{q}$ be the $(1-\alpha)$ conformal quantile (the usual finite-sample corrected quantile). For a new architecture a, output the interval

$$I(a) \;=\; \Big[\hat{g}(P(a)) - \hat{q},\; \hat{g}(P(a)) + \hat{q}\Big] \cap [0,1].$$

Under exchangeability of the calibration/test designs, we obtain the marginal coverage guarantee $\Pr\{F(a) \in I(a)\} \geq 1 - \alpha$ (with the understanding that $\bar{t}$ introduces additional noise, which can be mitigated by increasing r or by using noise-aware conformity scores). Such intervals support a conservative screening rule: retain all a whose *upper* endpoint is within τ of the best *lower* endpoint, i.e.,

$$\hat{g}(P(a)) + \hat{q} \;\geq\; \max_{a'} \big(\hat{g}(P(a')) - \hat{q}\big) - \tau,$$

so that we only discard architectures that are very unlikely to be competitive given the observed calibration uncertainty. This turns the proxy stage into an approximate "safe set" computation, with k emerging from the data rather than being fixed a priori.

Budget allocation between calibration and validation. Given an expensive-evaluation budget B, we allocate it across (i) calibration (learning $\hat{g}$) and (ii) validation (best-arm identification within candidates). The upper bound in Theorem 3 suggests a decomposition

$$B \;\approx\; mr + B_{\mathrm{val}}, \qquad m = \Theta\!\left(\frac{\mathrm{comp}(\mathcal{G}) + \log(1/\delta)}{\varepsilon^2}\right), \qquad B_{\mathrm{val}} = \Theta\!\left(\frac{k \log(k/\delta)}{\varepsilon^2}\right),$$

where r is the number of repeats per calibrated architecture and k is the post-screening candidate count. Diagnostics inform this allocation in two ways. First, if $\hat{\eta}_\alpha$ is large, increasing B_{val} alone is ineffective: we must either enlarge m (reducing estimation error of $\hat{g}$) or change $\mathcal{G}$ / the proxy. Second, if $\hat{\eta}_\alpha$ is small but noise dominates (large $\sigma/\sqrt{r}$), then we should increase r in both calibration and validation to sharpen estimates. A practical heuristic is to (a) start with a pilot calibration (small m) at moderate r, (b) compute $\hat{\eta}_\alpha$ and conformal widths, (c) set screening to achieve a manageable k while keeping intervals conservative, and (d) spend the remaining budget on validation using successive halving or fixed-confidence best-arm identification within the retained set.

Recommended experimental reporting. When proxies are evaluated in benchmarks or empirical NAS studies, we recommend reporting: (i) the held-out residual distribution (overall and in the high-proxy region), (ii) conformal interval widths and empirical coverage (when additional held-out true evaluations are available), (iii) sensitivity of the final selected $F(\hat{a})$ to the screening threshold (or to k), and (iv) a budget breakdown $(m, r, k, B_{\mathrm{val}})$. These diagnostics make explicit whether proxy guidance is acting through genuine calibration (small residuals) or merely through accidental correlation, and they render the proxy–truth linkage empirically falsifiable within the same oracle model.

9 Discussion and future work

We conclude by outlining several natural extensions of the proxy–truth framework, and by highlighting implications for empirical NAS methodology.

Beyond single objectives: Pareto and constrained optimization. Many NAS applications are inherently multi-objective: we care not only about predictive performance but also about latency, energy, memory footprint, or other deployment constraints. A direct extension models a vector-valued truth

$$F(a) \in [0,1]^d$$

(e.g. $d = 2$ for accuracy and normalized latency), and asks for an ε-approximate Pareto set, or for a point satisfying a constraint such as $F_{\text{lat}}(a) \leq \tau$ while maximizing $F_{\text{acc}}(a)$. Proxies likewise become vector-valued or partially observed, with potentially different noise profiles and distortions across coordinates. One principled approach is to posit coordinate-wise bounded distortion: there exist calibration maps $g_j \in \mathcal{G}_j$ such that

$$\left|F_j(a) - g_j(P_j(a))\right| \leq \eta_j, \qquad j = 1, \ldots, d,$$

and then use a calibrated proxy *surrogate* for each objective. Screening can proceed by (i) scalarization (e.g. weighted sums or Chebyshev norms), or (ii) conservatively retaining any architecture whose calibrated intervals intersect a target Pareto region. The latter naturally leverages uncertainty quantification (e.g. conformal bands) to avoid discarding Pareto-relevant points.

Theoretical guarantees in this setting require care: in multi-objective optimization, small additive errors can change dominance relations near the frontier. Nonetheless, we expect analogues of Theorem 3 in which k scales with the complexity of the frontier (or with a discretization of the scalarization weights) and the calibration sample complexity depends on $\sum_j \mathrm{comp}(\mathcal{G}_j)$. A particularly appealing special case is *hardware-aware* NAS where the cost objective (latency/energy) can often be measured nearly deterministically and cheaply, whereas accuracy is expensive and noisy. Then one can treat the cost as an additional proxy feature and calibrate accuracy conditional on cost strata, thereby sharpening the effective distortion in the relevant deployment slice.

Hardware-aware costs and nonstationary evaluation. Even in single-objective settings, hardware effects introduce a mismatch between the proxy environment and the deployment environment: the same architecture can exhibit different latency or throughput across devices, compiler stacks, or batch sizes. This is naturally captured as distribution shift in either P or F (or both). One direction is to incorporate *context* x (device, compiler, batch size) and model $F(a, x)$ with proxies $P(a, x)$, with a calibration class

$\mathcal{G}$ mapping (P, x) to predicted F. Another is to treat hardware variation as an additional noise term in $\widetilde{F}$, which inflates σ and correspondingly increases the necessary number of repeats r for stable conclusions. Both perspectives suggest that benchmark reports should specify not only mean performance but also the evaluation context and the variance across contexts, as these directly affect the feasibility of any $(\mathcal{G}, \eta)$-style linkage.

Adaptive choice of the calibration class. Our formal results assume a known function class $\mathcal{G}$ containing a suitable calibrator g. In practice, however, we rarely know whether g is well-approximated by an isotonic map, a Lipschitz function, a small neural predictor, or a richer nonparametric class. This motivates *data-dependent* selection of $\mathcal{G}$ under an expensive-evaluation budget.

A standard route is to consider a nested family $\mathcal{G}_1 \subset \mathcal{G}_2 \subset \cdots$ with increasing expressive power and increasing complexity penalty. Using the calibration sample, we can choose an index $\hat{j}$ by a held-out risk estimate plus a complexity term (structural risk minimization), or by a PAC-Bayes style bound. Conceptually, this trades approximation error (distortion not captured by $\mathcal{G}_j$) against estimation error (the uniform convergence term governed by $\mathrm{comp}(\mathcal{G}_j)$). An open theoretical question is to design an end-to-end algorithm that allocates budget adaptively between (i) expanding the class to reduce bias and (ii) collecting more calibration points to reduce variance, while preserving a guarantee of the form

$$F(\hat{a}) \geq \mathrm{OPT} - \varepsilon - \mathrm{poly}(\eta, \text{estimation error})$$

with the best achievable tradeoff among the candidate classes. We also expect gains from *active* calibration designs that oversample architectures in proxy regions where model disagreement across classes is high, rather than sampling uniformly.

Implications for benchmarks and reproducibility. The proxy-only impossibility (Theorem 1) underscores a methodological point: reporting a proxy–truth correlation on a fixed benchmark does not by itself justify proxy-only selection, because selection requires *calibrated* ranking accuracy in the tail, not global correlation. Conversely, our bounded-distortion viewpoint suggests concrete changes to benchmark design and reporting standards:

- *Publish calibration splits.* Benchmarks should provide standardized protocols for splitting architectures into calibration, screening, and validation subsets, so that proxy calibration and its uncertainty can be audited.

- *Report tail diagnostics.* Residual distributions should be reported not only globally but also restricted to the high-proxy region that drives selection. This reduces the risk of proxies that "look good on average" but fail near the optimum.

- *Quantify evaluation noise.* Benchmarks should specify variance across seeds and training nondeterminism; without this, the effective σ is unknown and sample-complexity comparisons are not meaningful.
- *Avoid inadvertent leakage.* If proxies are trained on labels derived from the same benchmark evaluations, the calibration problem becomes ill-posed and can lead to overly optimistic conclusions about generalization.

More broadly, the calibration-first perspective encourages separating questions of *proxy quality* (small η on the relevant region) from questions of *optimization strategy* (how to spend B once a reliable proxy exists). This separation can improve reproducibility: two studies that use different search heuristics but share the same calibration diagnostics and noise estimates can be compared on an equal footing, since their effective information budget is commensurable.

Open directions. Several theoretical directions remain. First, bounded distortion is an *absolute* condition; it would be useful to formalize *local* or *tail* distortion conditions that only need to hold near optimal architectures, aligning more tightly with screening. Second, in many pipelines the proxy itself is learned adaptively (e.g. predictors trained online); analyzing joint learning of P and g under a single expensive-evaluation budget is largely open. Finally, the interplay between screening and validation suggests refined lower bounds that depend on the *empirical* candidate set size k and on proxy-induced margins, rather than worst-case k and worst-case gaps. Establishing such instance-dependent guarantees would move the theory closer to the behavior observed in practical NAS systems.