EFFICIENT SEQUENTIAL CALIBRATION WITH $O(T^{2/3-\epsilon})$ ERROR BOUND

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the online binary sequential calibration problem. A recent breakthrough by Dagan et al. (2025) overcomes the classical $T^{2/3}$ barrier for calibration error. Building on this result, we present a computationally efficient randomized forecaster that achieves an expected calibration error $O(T^{2/3-\varepsilon})$ for some constant $\varepsilon > 0$.

Our forecaster combines the SPR-CALIBRATION procedure (Dagan et al., 2025) with an outer Blackwell-style correction layer. The SPR-CALIBRATION procedure controls calibration with respect to a surrogate sequence of conditional-mean estimates, while the correction layer controls the additional error incurred when these surrogates are used to approximate the true outcomes. The analysis decomposes the total calibration error into the surrogate calibration error and the residual discrepancy between the surrogate sequence and the true outcomes. The former is bounded by the SPR-CALIBRATION guarantee in Dagan et al. (2025), and the latter is controlled using a quadratic potential argument together with the sparsity of the SPR-CALIBRATION forecaster.

1 INTRODUCTION

Calibration captures a basic reliability requirement for probabilistic predictions: when a forecaster repeatedly assigns probability p to an event, the event should occur on roughly a p-fraction of those rounds. This notion is important because probability forecasts are often used directly in downstream decisions, such as risk assessment, medical prediction, weather forecasting, and machine-learning systems that report confidence scores (Dawid, 1982; Guo et al., 2017; Kuleshov et al., 2018; Hébert-Johnson et al., 2018). In these applications, the numerical value of a forecast matters, not only the ranking of alternatives: a prediction of 0.8 should be interpretable as an event that happens about 80% of the time. Sequential calibration studies how to achieve this reliability guarantee when predictions are made online and the data may arrive adaptively or non-stationarily.

In this work, we consider the online sequential calibration problem initiated by Foster & Vohra (1998). At each round t, a forecaster announces a probability $p_t \in [0, 1]$ for a possibly adversarial binary outcome $y_t \in \{0, 1\}$. The standard L_1-calibration error after T rounds is

$$\mathrm{CalErr}_T = \sum_{p \in \mathcal{P}_T} \left| \sum_{t \leqslant T : p_t = p} (p - y_t) \right|,$$

where $\mathcal{P}_T$ is the set of distinct predictions used by the forecaster. Classical algorithms achieve calibration error of order $T^{2/3}$. One way to obtain this rate is through a discretization and Blackwell-approachability argument (Foster, 1999): the forecaster restricts predictions to a finite grid of size m, controls the calibration residuals over this grid using an approachability strategy, and balances the approachability term $O(\sqrt{mT})$ with the

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.

discretization term $O(T/m)$. Optimizing over m gives $m \asymp T^{1/3}$, and hence calibration error $O(T^{2/3})$. This approach is conceptually clean and computationally efficient, but it also highlights the longstanding $O(T^{2/3})$ barrier.

A recent breakthrough by Dagan et al. (2025) showed that the classical $O(T^{2/3})$ upper bound is not intrinsic. They introduced an algorithm, which we refer to as SPR-CALIBRATION, that achieves a calibration error bound of order $O(T^{2/3-\varepsilon})$ for some constant $\varepsilon > 0$, thereby breaking the $T^{2/3}$ barrier for the first time.

Despite this breakthrough, obtaining an efficient $O(T^{2/3-\varepsilon})$ guarantee in the sequential setting is not immediate. The SPR-CALIBRATION algorithm (Dagan et al., 2025) is analyzed through a minimax reduction that allows the proof to work in a full-information model, where the forecaster is given the conditional mean $e_t = \mathbb{E}[y_t \mid \mathcal{F}_{t-1}]$ at each round. This idea goes back to the minimax proof of calibration (Hart, 2025): if the forecaster knew the adversary's mixed strategy, then it could simply predict the induced conditional probability of the next outcome; the minimax theorem then converts this observation into the existence of a randomized forecasting strategy that is calibrated against every adversary. However, this minimax transformation is primarily an existence argument rather than a computationally tractable algorithm.

It then naturally raises the question:

Is there a computationally efficient algorithm with $O(T^{2/3-\varepsilon})$ calibration error?

In this work, we answer this problem affirmatively.

Theorem 1. *There exists a forecaster (see Algorithm 2), such that for any (possibly adaptive) sequence $\{y_t\}_{t=1}^T \in \{0,1\}^T$, the expected calibration error $\mathbb{E}[\mathrm{CalErr}_T]$ is bounded by $O(\log_2(T) \cdot T^{2/3-\varepsilon_{\mathrm{AB}}/18})$, where $\varepsilon_{\mathrm{AB}} > 0$ is the same as the constant ε in Theorem 1.3 of Dagan et al. (2025). Moreover, the computation cost of the algorithm is $O(T^{7/2}\log_2(T))$.*

To the best of our knowledge, this is the first computationally efficient algorithm to achieve an L_1 calibration error of $O(T^{2/3-\epsilon})$. Moreover, any improvement in the value of the underlying SPR game would directly yield a sharper calibration bound while preserving computational efficiency.

Our algorithm is based a natural combination of the SPR-CALIBRATION procedure of Dagan et al. (2025) with an outer Blackwell-style correction layer. At each round t, the algorithm first forms a surrogate estimate $\widetilde{e}_t$ of the conditional mean and then passes this value to the calibration procedure of Dagan et al. (2025). Since calibration is ultimately measured against the realized outcome y_t, using the surrogate $\widetilde{e}_t$ introduces an additional source of error. To control this error, the algorithm augments the underlying calibration procedure with a Blackwell-style correction layer. The analysis separates the total calibration error into two components: the calibration error with respect to the surrogate sequence $\{\widetilde{e}_t\}_{t=1}^T$, and the discrepancy between the surrogate means and the realized outcomes. The first component is controlled directly by the guarantee of Dagan et al. (2025); the second is bounded using a quadratic potential argument together with the sparsity property of SPR-CALIBRATION.

We further extend the Blackwell wrapper to accommodate delayed outcomes, yielding a computationally efficient algorithm with an L_1 calibration error bound of $O\left(\log^2(2T)\, T^{\frac{2}{3}-\frac{\epsilon_{\mathrm{AB}}}{18}}\,(1+d)^{\frac{1}{3}+\frac{\epsilon_{\mathrm{AB}}}{18}}\right)$ (see Theorem 2), where d denotes the average delay. This extension highlights the flexibility of our framework and its potential applicability to other related settings.

1.1 RELATED WORKS

Calibration and adversarial forecasting. Calibration has a long history as a criterion for evaluating probabilistic forecasts. Dawid (1982) emphasized calibration as a basic consistency requirement for subjective

probabilities, and Foster & Vohra (1998) initiated the adversarial sequential calibration problem studied in this paper. Their work showed that randomized forecasters can be calibrated against arbitrary binary outcome sequences and gave the classical $O(T^{2/3})$ calibration-error guarantee. Several subsequent works gave alternative proofs and perspectives on calibration, including the myopic minimax construction (Fudenberg & Levine, 1999), the Blackwell-approachability proof (Foster, 1999), the minimax proof (Hart, 2025), and the geometric approachability proof (Mannor & Stoltz, 2010).

Approachability, regret, and calibration. Blackwell's approachability theorem (Blackwell, 1956) is one of the central tools behind adversarial calibration. In the approachability formulation, calibration residuals are treated as coordinates of a vector-valued payoff, and the forecaster chooses predictions so that the cumulative payoff approaches an appropriate target set. This connection was made explicit by Foster (1999) and was further developed through geometric and online-learning viewpoints (Mannor & Stoltz, 2010; Abernethy et al., 2011). Calibration is also closely related to regret minimization: Foster & Vohra (1999) connected calibration and internal regret, and calibrated learning rules are known to lead to correlated equilibrium in repeated games (Foster & Vohra, 1997). The broader connections between prediction of individual sequences, regret, and game-theoretic learning are surveyed in Cesa-Bianchi & Lugosi (2006).

Rates for sequential calibration. The optimal rate of L_1-calibration error has been a central question in sequential calibration. The classical upper bound is $O(T^{2/3})$, obtained by balancing the discretization error of a finite grid with an approachability-type residual term (Foster & Vohra, 1998; Abernethy et al., 2011). For many years, the only general lower bound was the trivial $\Omega(\sqrt{T})$ bound obtained from independent fair coin flips. Qiao & Valiant (2021) gave the first super-$\sqrt{T}$ lower bound, proving an $\Omega(T^{0.528})$ lower bound via the sign-preservation game. Dagan et al. (2025) recently broke the $T^{2/3}$ upper-bound barrier by introducing *sign preservation with reuse* (SPR), proving that improved SPR strategies yield calibration error $O(T^{2/3-\varepsilon})$ for some constant $\varepsilon > 0$. They also improved the lower bound to $\Omega(T^{0.54389})$, leaving a gap between the known upper and lower exponents.

2 PRELIMINARIES

We follow the sequential calibration setup of Dagan et al. (2025). Fix a time horizon T. At each round $t \in [T]$, the forecaster outputs a probability prediction $p_t \in [0, 1]$, and the environment outputs an outcome $y_t \in \{0, 1\}$. The prediction p_t is interpreted as the forecaster's announced probability that $y_t = 1$.

The adversary may be adaptive, in the sense that its choice at time t may depend on the past history $H_t := \{(p_s, y_s) : s < t\}$, but it does not observe the current prediction p_t before choosing y_t. Let $\mathcal{P}_t := \{p_s : s \leqslant t\}$ denote the set of distinct probability values actually output up to time t. Define $E_t(p) = \sum_{s \leqslant t: p_s = p} (p - y_s)$. The cumulative L_1-calibration error at time t is

$$\mathrm{CalErr}_t := \sum_{p \in \mathcal{P}_t} |E_t(p)| = \sum_{p \in \mathcal{P}_t} \left| \sum_{s \leqslant t: p_s = p} (p - y_s) \right|.$$

When the forecaster is randomized, all quantities above are random variables. We measure performance by expected calibration error $\mathbb{E}[\mathrm{CalErr}_T]$, where the expectation is over the forecaster's internal randomness.

Notations. For an integer N, we use $[N]$ to denote the set $\{1, 2, \ldots, N\}$. For any $1 \leqslant t \leqslant T + 1$, we use $\mathcal{F}_{t-1}$ to denote the event field before the t-th round. Without loss of generality, we assume T is a power of 2[1]

[1]For a general horizon T, we can use its binary expansion to partition the horizon into consecutive segments whose lengths are powers of two, and run the algorithm separately on each segment.

Algorithm 1 L_1-Calibration Protocol

Input: Number of rounds T.
for $t = 1, 2, \ldots, T$ **do**
  Option: Receive the conditional mean $e_t \in [0, 1]$, such that $\mathbb{E}[y_t|\mathcal{F}_{t-1}] = e_t$
  Set the history $\mathcal{H}_t \leftarrow \{p_s, y_s\}_{s=1}^{t-1}$ $\left(\mathcal{H}_t \leftarrow \{p_s, y_s\}_{s=1}^{t-1} \cup \{e_s\}_{s=1}^{t} \text{ if } \textbf{Option} \text{ is chosen}\right)$
  Generate a prediction $p_t \in [0, 1]$ using some algorithm $\mathcal{A}$ and history $\mathcal{H}_t$
  Receive the outcome $y_t \in \{0, 1\}$.
end for
Let $\mathcal{P}_T \leftarrow \{p_1, \ldots, p_T\}$ be the set of distinct predictions.
Return: The L_1 calibration error $\mathrm{CalErr}_T = \sum_{p \in \mathcal{P}_T} \left| \sum_{t=1}^{T} \mathbf{1}\{p_t = p\}(y_t - p_t) \right|$.

Knowledge of conditional mean. The full protocol of the calibration problem is presented in Algorithm 1. In the original setting, the learner does not observe the conditional mean $e_t = \mathbb{E}[y_t \mid \mathcal{F}_{t-1}]$. By Yao's minimax principle (Yao, 1977), to establish an upper bound on the expected calibration error, it suffices to construct, for every fixed mixed strategy of the adversary, a learner that achieves this bound with knowledge of that strategy. In this auxiliary setting, the conditional mean e_t is therefore known to the learner, as in Hart (2025). However, this minimax argument establishes the existence of a suitable learner without directly providing a computationally efficient algorithm.

3 Main Algorithm

We present our main algorithm in Algorithm 2. The algorithm maintains an instance of SPR-Calibration (Dagan et al., 2025). The main difficulty in applying SPR-Calibration is that the conditional means $\{e_t\}_{t=1}^{T}$ are not available to the forecaster. To address this, we use a Blackwell-style correction layer to construct a surrogate estimate $\widetilde{e}_t$ of e_t at each round $t \in [T]$. This surrogate is then passed as the input to the SPR-Calibration instance.

3.1 The SPR-Calibration Algorithm

Our algorithm uses a deterministic implementation of SPR-Calibration as a black-box subroutine. In the mean-aware setting, at each round t, this subroutine takes the past predictions $\{p_s\}_{s=1}^{t-1}$ and the conditional means $\{e_s\}_{s=1}^{t}$ as input and returns a prediction p_t.[2] To formalize this interface, let S^{SPR} denote the internal state of a SPR-Calibration instance (see Definition 2 in Appendix A). We use the following two functions:

$$\mathrm{Predict}_{\text{SPR}}(S^{\text{SPR}}, t, z_t) \qquad \text{and} \qquad \mathrm{Update}_{\text{SPR}}(S^{\text{SPR}}, t, z_t).$$

Both functions take the current internal state S^{SPR}, the round index t, and a supplied mean $z_t \in [0, 1]$ as input. The first returns the prediction p_t, while the second returns the updated internal state. Thus, given an initial state S_0^{SPR} and a sequence $\{z_t\}_{t=1}^{T}$, the subroutine predicts $\{p_t\}_{t=1}^{T}$ according to the recursion

$$\begin{aligned} p_t &= \mathrm{Predict}_{\text{SPR}}(S_t^{\text{SPR}}, t, z_t), \\ S_{t+1}^{\text{SPR}} &= \mathrm{Update}_{\text{SPR}}(S_t^{\text{SPR}}, t, z_t). \end{aligned} \tag{1}$$

Due to space constraints, we defer the details of this deterministic implementation of SPR-Calibration to Algorithm 3 in Appendix A, where we also introduce the relevant algorithmic notations.

[2] The algorithm of Dagan et al. (2025) does not require the realized outcome y_t to compute p_t.

Building on the theoretical results of Dagan et al. (2025), we obtain two key guarantees. Recall that $\mathcal{P}_T = \{p_1, \ldots, p_T\}$ denotes the set of distinct predictions. With an appropriate choice of parameters, there exists a constant $\kappa > 0$ such that

$$\sum_{p \in \mathcal{P}_T} \left| \sum_{t:p_t = p} (z_t - p_t) \right| = O(T^{2/3-\kappa}), \tag{2}$$

$$|\mathcal{P}_T| = O(T^{1/3-\kappa}). \tag{3}$$

These two properties provide the slack needed for our Blackwell-type correction. The first directly bounds the calibration error with respect to the supplied sequence $\{z_t\}_{t=1}^T$. The second limits the number of distinct predictions, allowing the correction to control the additional error incurred when replacing the supplied means z_t with the realized outcomes y_t. Together with the correction analysis, these guarantees yield the final $O(T^{2/3-\kappa})$ calibration error bound, possibly after decreasing the constant $\kappa > 0$. Formal statements of the theoretical guarantees of SPR-CALIBRATION are provided in Appendix B.

3.2 THE BLACKWELL-STYLE CORRECTION

Let $\mathcal{X} = \{0, \eta, 2\eta, \ldots, N\eta\}$ be a grid with mesh width $\eta = T^{-1/2}$ and $N = \lfloor 1/\eta \rfloor$. At each round t, let S_t^{SPR} denote the internal state of the SPR-CALIBRATION instance used by Algorithm 2 before round t. Recall that $\mathcal{P}_t = \{p_s : s \leqslant t\}$. For each $x \in \mathcal{X}$ and $p \in \mathcal{P}_{t-1}$, we define

$$D_t(x) := \mathrm{Predict}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, x) \tag{4}$$

$$R_{p,t-1} := \sum_{s<t:p_s=p} (y_s - \tilde{e}_s) \tag{5}$$

Moreover, for each $x \in \mathcal{X}$, define $\widehat{R}_{x,t-1} := R_{D_t(x),t-1}$. In particular, if $D_t(x) \notin \mathcal{P}_{t-1}$, we use the default initialization $\widehat{R}_{x,t-1} = R_{D_t(x),t-1} = 0$.

Given the quantities $\widehat{R}_{x,t-1}$ defined above, we solve the following Blackwell-style minimax program to obtain an optimal distribution $\mu_t \in \Delta(\mathcal{X})$, where $\Delta(\mathcal{X})$ denotes the set of probability distributions over $\mathcal{X}$.

$$\beta_t = \min_{\mu \in \Delta(\mathcal{X})} \max_{y \in \{0,1\}} \sum_{x \in \mathcal{X}} \mu(x) \widehat{R}_{x,t-1}(y - x). \tag{6}$$

Then the algorithm samples the surrogate mean $\widetilde{e}_t \sim \mu_t$, and runs the deterministic SPR-CALIBRATION steps:

$$p_t = D_t(\widetilde{e}_t) = \mathrm{Predict}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, \widetilde{e}_t), \qquad S_{t+1}^{\mathrm{SPR}} = \mathrm{Update}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, \widetilde{e}_t).$$

After outputting the forecast p_t and observing the true outcome y_t, we update $\mathcal{P}_t$ and the residuals $R_{p,t}$ for all $p \in \mathcal{P}_t$.

4 ANALYSIS

In this section, we present the proof of Theorem 1. The proof consists of two parts: bounding the calibration error and the computational cost.

Algorithm 2 Blackwell-Wrapped SPR-CALIBRATION

Require: Horizon T, grid $\mathcal{X}$.

Initialize the internal state S_1^{SPR} of SPR-CALIBRATION
Initialize the set of actually predicted values $\mathcal{P}_0 \leftarrow \varnothing$
for $t = 1, 2, \ldots, T$ **do**
  Define $D_t(x)$ following equation 4 for all $x \in \mathcal{X}$;
  For all $x \in \mathcal{X}$, set $\widehat{R}_{x,t-1} = R_{D_t(x),t-1}$.
  Choose $\mu_t \in \Delta(\mathcal{X})$ by solving:

$$\min_{\mu \in \Delta(\mathcal{X})} \max_{y \in \{0,1\}} \sum_{x \in \mathcal{X}} \mu(x) \widehat{R}_{x,t-1}(y - x) \tag{7}$$

  Sample an internal surrogate mean $\widetilde{e}_t \sim \mu_t$.
  Feed $\widetilde{e}_t$ to SPR-CALIBRATION and update SPR-CALIBRATION:

$$(p_t, S_{t+1}^{\mathrm{SPR}}) \leftarrow \left(\mathrm{Predict}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, \widetilde{e}_t), \mathrm{Update}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, \widetilde{e}_t)\right). \tag{8}$$

  if $p_t \notin \mathcal{P}_{t-1}$ **then**
    $\mathcal{P}_t \leftarrow \mathcal{P}_{t-1} \cup \{p_t\}$
    $R_{p_t,t-1} \leftarrow 0$
  else
    $\mathcal{P}_t \leftarrow \mathcal{P}_{t-1}$
  end if
  Observe the outcome $y_t \in \{0, 1\}$.
  Update the external Blackwell residual: $R_{p_t,t} \leftarrow R_{p_t,t-1} + y_t - \widetilde{e}_t$.
  For all $p \in \mathcal{P}_t \backslash \{p_t\}$, set $R_{p,t} \leftarrow R_{p,t-1}$.
end for

Let $\tau = \log_2 T$, and choose the scale parameter $h \leqslant \tau$ of SPR-CALIBRATION later. Recall $\mathcal{P}_t = \{p_s : s \leqslant t\}$. For $t \in [T]$ and $p \in \mathcal{P}_t$, recall the outer residual $R_{p,t} = \sum_{s \leqslant t: p_s = p}(y_s - \widetilde{e}_s)$ if $p \in \mathcal{P}_t$. Define $R_{p,t} = 0$ if $p \notin \mathcal{P}_t$.

For every $p \in \mathcal{P}_T$, $\sum_{t \in [T]: p_t = p}(y_t - p) = \sum_{t \in [T]: p_t = p}(\widetilde{e}_t - p) + \sum_{t \in [T]: p_t = p}(y_t - \widetilde{e}_t)$. Therefore,

$$\mathbb{E}[\mathrm{CalErr}_T] \leqslant \underbrace{\mathbb{E}\left[\sum_{p \in \mathcal{P}_T} \left| \sum_{t \in [T]: p_t = p} (\widetilde{e}_t - p) \right| \right]}_{\text{SPR-CALIBRATION bias term}} + \underbrace{\mathbb{E}\left[\sum_{p \in \mathcal{P}_T} |R_{p,T}|\right]}_{\text{Blackwell residual term}}. \tag{9}$$

Bound of the SPR-CALIBRATION bias term. The bound of the SPR-CALIBRATION bias directly follows the result in Dagan et al. (2025). Let $\gamma = \frac{1-\varepsilon_{\mathrm{AB}}}{2-\varepsilon_{\mathrm{AB}}}$. By the property in equation 2 (see formal statement in Lemma 5, Appendix B), we have that

$$\mathbb{E}\left[\sum_{p \in \mathcal{P}_T} \left| \sum_{t \in [T]: p_t = p} (\widetilde{e}_t - p) \right| \right] \leqslant O\left(\sum_{i=1}^{\tau-h} \sum_{j=i+1}^{i+h} 2^{j-i} 2^{\gamma(i+\tau-j)} + \sum_{i \leqslant \tau-h} \sum_{j=i+1}^{i+h} \min\{2^i, 2^{\tau-h-i}\} \right). \tag{10}$$

Bound of the Blackwell residual term. By the property in equation 3 (see formal statement in Lemma 3, Appendix B), we have that $|\mathcal{P}_T| \leqslant O\big(2^{(\tau-h)/2}\big)$. Then by Cauchy's inequality, we have that

$$\mathbb{E}\left[\sum_{p\in\mathcal{P}_T}|R_{p,T}|\right] \leqslant O\left(\sqrt{2^{(\tau-h)/2}\mathbb{E}\left[\sum_{p\in\mathcal{P}_T}R_{p,T}^2\right]}\right) \leqslant O\left(\sqrt{\mathbb{E}[\Phi_T]}\cdot 2^{(\tau-h)/4}\right), \tag{11}$$

where we define $\Phi_t = \sum_{p\in\mathcal{P}_t} R_{p,t}^2$ for $t \in [T]$ and $\Phi_0 = 0$. The next step is to bound the total expected energy $\mathbb{E}[\Phi_T]$.

Lemma 1. $\mathbb{E}[\Phi_T] = O\big((\sqrt{T}+\eta T)^2\big)$. *If* $\eta = O(T^{-1/2})$, *then* $\mathbb{E}[\Phi_T] = O(T)$.

Proof. Recall that $\mathcal{X}$ is the grid with interval length η. At round t, the wrapper chooses a distribution $\mu_t \in \Delta(\mathcal{X})$ by solving the finite minimax problem equation 6 as

$$\beta_t = \min_{\mu\in\Delta(\mathcal{X})}\max_{y\in\{0,1\}}\sum_{x\in\mathcal{X}}\mu(x)\widehat{R}_{x,t-1}(y-x).$$

By Von Neumann's minimax theorem

$$\begin{aligned}\beta_t &= \max_{q\in\Delta(\{0,1\})}\min_{x\in\mathcal{X}}\mathbb{E}_{y\sim q}\left[\widehat{R}_{x,t-1}(y-x)\right] = \max_{q\in\Delta(\{0,1\})}\min_{x\in\mathcal{X}}\widehat{R}_{x,t-1}(\mathbb{E}_{y\sim q}[y]-x)\\ &= \max_{y\in[0,1]}\min_{x\in\mathcal{X}}\widehat{R}_{x,t-1}(y-x).\end{aligned} \tag{12}$$

Fix any $y \in [0,1]$, and let $x^*(y) \in \arg\min_{x\in\mathcal{X}}|y-x|$. Then $|y-x^*(y)| \leqslant \eta$. Therefore,

$$\min_{x\in\mathcal{X}}\widehat{R}_{x,t-1}(y-x) \leqslant \left|\widehat{R}_{x^*(y),t-1}\right|\eta.$$

Since $\left|\widehat{R}_{x^*(y),t-1}\right| \leqslant \sqrt{\Phi_{t-1}}$, we have $\min_{x\in\mathcal{X}}\widehat{R}_{x,t-1}(y-x) \leqslant \widehat{R}_{x^*(y),t-1}(y-x) \leqslant \eta\sqrt{\Phi_{t-1}}$. Taking the maximum over $y \in [0,1]$ gives $\beta_t \leqslant \eta\sqrt{\Phi_{t-1}}$. Since μ_t is optimal for the finite minimax problem, for both $y \in \{0,1\}$, $\mathbb{E}_{x\sim\mu_t}\left[\widehat{R}_{x,t-1}(y-x)\right] \leqslant \beta_t \leqslant \eta\sqrt{\Phi_{t-1}}$. Recalling that $\widehat{R}_{x,t-1} = R_{D_t(x),t-1}$ for all $x \in \mathcal{X}$, we always have

$$\mathbb{E}_{x\sim\mu_t}\left[R_{D_t(x),t-1}(y_t-x)\right] \leqslant \beta_t \leqslant \eta\sqrt{\Phi_{t-1}}. \tag{13}$$

Let $\mathcal{F}_{t-1}$ contain the history and the forecaster's internal state before round t. The adversary chooses y_t, possibly at random, without observing the forecaster's fresh random draw. Set $\mathcal{G}_t = \mathcal{F}_{t-1} \vee \sigma(y_t)$. Conditional on $\mathcal{G}_t$, the surrogate $\widetilde{e}_t$ has distribution μ_t. We then have that

$$\mathbb{E}[\Phi_t - \Phi_{t-1} \mid G_t] = \mathbb{E}_{x\sim\mu_t}[2R_{D_t(x),t-1}(y_t-x) + (y_t-x)^2 \mid \mathcal{G}_t] \leqslant 2\eta\sqrt{\Phi_{t-1}} + 1,$$

which implies

$$\mathbb{E}[\Phi_t] - \mathbb{E}[\Phi_{t-1}] \leqslant 2\eta\mathbb{E}[\sqrt{\Phi_{t-1}}] + 1.$$

Let $Z_t = \mathbb{E}[\Phi_t]$ for $t \geqslant 0$. By Jensen's inequality, $\mathbb{E}[\sqrt{\Phi_{t-1}}] \leqslant \sqrt{Z_{t-1}}$. Therefore,

$$Z_t \leqslant Z_{t-1} + 1 + 2\eta\sqrt{Z_{t-1}}.$$

Summing over t yields $Z_{t'} \leqslant t' + 2\eta\sum_{t=0}^{t'-1}\sqrt{Z_t}$ for any $t' \in [T]$. Let $M_T = \max_{0\leqslant t\leqslant T}\sqrt{Z_t}$. Then $M_T^2 \leqslant T + 2\eta T M_T$. Solving this quadratic inequality gives

$$M_T \leqslant \eta T + \sqrt{\eta^2T^2 + T} \leqslant 2\eta T + \sqrt{T}.$$

Hence

$$Z_T \leqslant M_T^2 \leqslant O\Big((\sqrt{T} + \eta T)^2\Big).$$

If $\eta \leqslant T^{-1/2}$, then $\sqrt{T} + \eta T = O(\sqrt{T})$, and therefore $\mathbb{E}[\Phi_T] = O(T)$. □

By Lemma 1 and equation 11, we have that $\mathbb{E}\left[\sum_{p\in\mathcal{P}_T} |R_{p,T}|\right] \leqslant O(\sqrt{T} \cdot 2^{(\tau-h)/4})$.

Putting all together. Recall that $\eta = T^{-1/2}$. Let $\varepsilon_{\mathrm{AB}}$ be defined in Lemma 4 and recall $\gamma = \frac{1-\varepsilon_{\mathrm{AB}}}{2-\varepsilon_{\mathrm{AB}}}$. We then have that

$$\begin{aligned}
\mathbb{E}[\mathrm{CalErr}_T] &\leqslant O\left(\sum_{i=1}^{\tau-h}\sum_{j=i+1}^{i+h} 2^{j-i}2^{\gamma(i+\tau-j)} + \sum_{i=1}^{\tau-h}\sum_{j=i+1}^{i+h} \min\{2^i, 2^{\tau-h-i}\}\right) + O\left(\sqrt{T}\cdot 2^{(\tau-h)/4}\right) \\
&\leqslant O\left(\sum_{i=1}^{\tau-h} 2^{h(1-\gamma)+\gamma\tau} + \log_2(T)\cdot\sum_{i\leqslant\tau-h}\min\{2^i, 2^{\tau-h-i}\} + \sqrt{T}\cdot 2^{(\tau-h)/4}\right) \\
&\leqslant O\left(\log_2(T)\cdot 2^{h+\gamma(\tau-h)} + \log_2(T)\cdot 2^{(\tau-h)/2} + \sqrt{T}\cdot 2^{(\tau-h)/4}\right) \qquad (14) \\
&\leqslant O\Big(\log_2(T)\cdot 2^{h+\gamma(\tau-h)} + \log_2(T)\cdot 2^{3\tau/4-h/4}\Big). \qquad (15)
\end{aligned}$$

Here, equation 14 follows from the inequality $j - i \leqslant h$ and the fact that $\sum_{i=1}^{\tau-h}\min\{2^i, 2^{\tau-h-i}\} = O(2^{(\tau-h)/2})$, while equation 15 follows from $(\tau-h)/2 \leqslant 3\tau/4 - h/4$. By choosing $h = \left\lceil \frac{3-4\gamma}{5-4\gamma}\tau \right\rceil = \left\lceil \frac{2+\varepsilon_{\mathrm{AB}}}{6-\varepsilon_{\mathrm{AB}}}\tau \right\rceil$, we have that $\mathbb{E}[\mathrm{CalErr}_T] \leqslant O\left(\log_2(T)\cdot T^{\frac{2}{3}-\frac{\varepsilon_{\mathrm{AB}}}{18}}\right)$.

Finally, since SPR-CALIBRATION admits a polynomial-time implementation, Algorithm 2 also runs in polynomial time. Specifically, its overall running time is $O\big(T^{7/2}\log T\big)$, as established in Lemma 6 of Appendix C.

5 CALIBRATION WITH DELAYED FEEDBACK

Delayed feedback arises naturally in sequential prediction, for example when advertising conversions are observed only some time after an ad impression (Chapelle, 2014). The need to continue making predictions while earlier outcomes remain unavailable motivates the study of online learning with delayed feedback (Joulani et al., 2013; Quanrud & Khashabi, 2015).

We extend the preceding setting by allowing each outcome y_t to be revealed, together with its round index, at the end of round $t + d_t$, where $d_t \geqslant 0$ is an integer delay. Thus, $d_t = 0$ recovers the immediate-feedback setting. The delays need not be known in advance, and outcomes may arrive out of order or remain unobserved at the end of the horizon.

Let m_t denote the number of pending outcomes when the prediction at round t is made, and define the cumulative delay $\mathcal{D}_T$ by

$$\begin{aligned}
m_t &:= \big|\{s < t : s + d_s \geqslant t\}\big|, \\
\mathcal{D}_T &:= \sum_{t=1}^{T} m_t = \sum_{s=1}^{T} \min\{d_s, T - s\}. \qquad (16)
\end{aligned}$$

Each pending outcome contributes one unit to $\mathcal{D}_T$ for every subsequent prediction made before its revelation, with waiting time truncated at the horizon. Our objective remains to control the expected calibration error over all T predictions and their corresponding outcomes, including outcomes not yet revealed by time T.

Under this setting, we have the following results for L_1 calibration error.

Theorem 2. *Fix a horizon $T \geqslant 2$. There exists a polynomial-time delay-adaptive Blackwell-wrapped* SPR-CALIBRATION *forecaster that satisfies*

$$\mathbb{E}[\mathrm{CalErr}_T] \leqslant O\Big(\log^2(2T)\, T^{\frac{2}{3}-\frac{\varepsilon_{\mathrm{AB}}}{18}}\, (1+d)^{\frac{1}{3}+\frac{\varepsilon_{\mathrm{AB}}}{18-3\varepsilon_{\mathrm{AB}}}}\Big), \tag{17}$$

where $\varepsilon_{\mathrm{AB}} > 0$ is the same constant as in the main theorem and $d = \mathbb{E}[\mathcal{D}_T]/T$.

Remark 1. The forecaster is given T, but requires no advance knowledge of the individual delays or any upper bound on their cumulative budget. The expectation is over the forecaster's internal randomness and any randomness in the environment and feedback process.

The proof builds on a natural combination of the Blackwell correction layer and SPR-CALIBRATION. To accommodate unknown delays, we initialize multiple instances of SPR-CALIBRATION, each equipped with its own Blackwell-style correction layer. We obtain the overall calibration error bound by summing the per-instance bounds controlled in terms of the cumulative delay. Due to space constraints, we defer the algorithm and its analysis to Appendix D.

6 DISCUSSION

In this paper, we develop an efficient algorithm that achieves $O(T^{2/3-\varepsilon})$ calibration error. Our algorithm is based on a simple combination of the SPR-CALIBRATION algorithm in Dagan et al. (2025) and a Blackwell-approachability correction argument. The resulting improvement exponent ε is inherited from the SPR guarantee in Dagan et al. (2025); in particular, any improvement in the value of the underlying SPR game would translate directly into a stronger efficient calibration bound. We also extend the framework to accommodate delayed feedback, demonstrating its flexibility and potential applicability to other related settings.

AI USE STATEMENT

AI tools (GPT-5.6 and GPT-6) were used to refine the writing, assist with the literature review, and help check the proofs. The authors developed the algorithm and theoretical analysis independently, without AI assistance. The authors take full responsibility for the content of this paper, including the accuracy of the references, the correctness of the proofs, and the validity of the results.

ETHICS STATEMENT

This work is theoretical in nature, and we do not anticipate any significant ethical concerns arising from it.

REPRODUCIBILITY STATEMENT

This work is theoretical in nature. Our main result is stated in Theorem 1 and proved in Section 4. The proof relies on several properties of the SPR-CALIBRATION algorithm established by Dagan et al. (2025), which we present in Appendices A and B.

REFERENCES

Jacob Abernethy, Peter L Bartlett, and Elad Hazan. Blackwell approachability and no-regret learning are equivalent. In *Proceedings of the 24th Annual Conference on Learning Theory*, pp. 27–46. JMLR Workshop and Conference Proceedings, 2011.

David Blackwell. An analog of the minimax theorem for vector payoffs. 1956.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Olivier Chapelle. Modeling delayed feedback in display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1097–1105, 2014.

Yuval Dagan, Constantinos Daskalakis, Maxwell Fishelson, Noah Golowich, Robert Kleinberg, and Princewill Okoroafor. Breaking the tˆ(2/3) barrier for sequential calibration. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pp. 2007–2018, 2025.

A Philip Dawid. The well-calibrated bayesian. *Journal of the American statistical Association*, 77(379): 605–610, 1982.

Dean P Foster. A proof of calibration via blackwell's approachability theorem. *Games and Economic Behavior*, 29(1-2):73–78, 1999.

Dean P Foster and Rakesh Vohra. Regret in the on-line decision problem. *Games and Economic Behavior*, 29 (1-2):7–35, 1999.

Dean P Foster and Rakesh V Vohra. Calibrated learning and correlated equilibrium. *Games and Economic Behavior*, 21(589):40–55, 1997.

Dean P Foster and Rakesh V Vohra. Asymptotic calibration. *Biometrika*, 85(2):379–390, 1998.

Drew Fudenberg and David K Levine. An easier way to calibrate. *Games and economic behavior*, 29(1-2): 131–137, 1999.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Sergiu Hart. Calibrated forecasts: The minimax proof. In *Matching, Dynamics and Games for the Allocation of Resources: Essays in Celebration of David Gale's 100th Birthday*, pp. 153–159. Springer, 2025.

Ursula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International conference on machine learning*, pp. 1939–1948. PMLR, 2018.

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International conference on machine learning*, pp. 1453–1461. PMLR, 2013.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International conference on machine learning*, pp. 2796–2804. PMLR, 2018.

Shie Mannor and Gilles Stoltz. A geometric proof of calibration. *Mathematics of Operations Research*, 35 (4):721–727, 2010.

Mingda Qiao and Gregory Valiant. Stronger calibration lower bounds via sidestepping. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 456–466, 2021.

Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. *Advances in neural information processing systems*, 28, 2015.

Andrew Chi-Chin Yao. Probabilistic computations: Toward a unified measure of complexity. In *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, pp. 222–227. IEEE, 1977.

A THE SPR-CALIBRATION ALGORITHM

This section specifies the version of SPR-CALIBRATION used by Algorithm 2. For completeness, we present the one-step version of the main algorithm of Dagan et al. (2025) in Algorithm 3, with several key subroutines: `simulateGame` (see Algorithm 5) and the A/B player-L strategy (see Algorithm 6 and 7). The full version is presented in Algorithm 4.

A.1 THE NOTATIONS IN SPR-CALIBRATION

Definition 1 (Sign-preservation with reuse (SPR))**.** *For $n, s \in \mathbb{N}$, the game $\mathrm{SPR}(n, s)$ is played between two players, called Player-P and Player-L. The game consists of n cells, indexed by $[n] = \{1, \ldots, n\}$, all of which are initially empty. The game lasts for at most s rounds. In each round, the following steps occur:*

1. *Player-P may terminate the game. Otherwise, Player-P chooses an empty cell $j \in [n]$.*
2. *After observing j, Player-L may remove any subset of the $-$ signs in cells strictly to the left of j, and any subset of the $+$ signs in cells strictly to the right of j.*
3. *Player-L then places either a $+$ sign or a $-$ sign in cell j.*

A cell whose sign has been removed becomes empty and may therefore be chosen again in a later round. Player-P aims to maximize the number of signs remaining on the board at the end of the game, while Player-L aims to minimize this quantity.

Scales and SPR instances. Recall that $T = 2^\tau$ and $\tau \geqslant 2$. Fix an integer $1 \leqslant h < \tau$. Define $\mathcal{I}_h := \{1, \ldots, \tau - h\}$, $\mathcal{J}_i := \{i + 1, \ldots, i + h\}$ and $C_i := \{1, \ldots, 2^i\}$. For every $i \in \mathcal{I}_h$, $j \in \mathcal{J}_i$, and $\ell \in \{0, 1\}$, the SPR-CALIBRATION algorithm maintains one SPR instance $G_{i,j,\ell}$ with cell set C_i. The index i specifies the spatial resolution, j specifies the time scale, and ℓ separates the even and odd dyadic intervals at resolution $2^{-(i+1)}$.

Cells, intervals, and forecast values. For $z \in [0, 1]$, let $m_i(z) := \min\left\{\left\lfloor 2^{i+1} z \right\rfloor, 2^{i+1} - 1\right\}$, $\ell_i(z) := m_i(z) \bmod 2$ and $c_i(z) := \frac{m_i(z) - \ell_i(z)}{2} + 1$. We define $\mathrm{cell}(i, j, z) := \big(c_i(z), G_{i,j,\ell_i(z)}\big)$. Thus, j selects the SPR instance at the desired time scale, while i and z determine the parity and cell.

For $c \in C_i$ and $\ell \in \{0, 1\}$, write $m = 2(c - 1) + \ell$ and define

$$\mathrm{interval}(c, G_{i,j,\ell}) := \begin{cases} \left[m/2^{i+1}, (m + 1)/2^{i+1}\right), & m < 2^{i+1} - 1, \\ \left[(2^{i+1} - 1)/2^{i+1}, 1\right], & m = 2^{i+1} - 1. \end{cases}$$

In this way, every $z \in [0, 1]$ is assigned to a unique cell[3]. For a sign $\sigma \in \{+, -\}$, define

$$\mathrm{prob}(c, \sigma, G_{i,j,\ell}) := \begin{cases} \dfrac{\max\{0, 2(c - 1) + \ell - 1\}}{2^{i+1}}, & \sigma = +, \\ \dfrac{\min\{2(c - 1) + \ell + 2, 2^{i+1}\}}{2^{i+1}}, & \sigma = -. \end{cases}$$

We then give a precise definition of the state of an SPR-CALIBRATION instance.

[3]We use this half-open interval convention to make the interval assignment explicit.

Definition 2 (State of SPR-CALIBRATION). *For each* SPR *instance $G = G_{i,j,\ell}$, the subroutine stores*

$$\begin{aligned} &\mathrm{bias}_G : C_i \to \mathbb{R}, && \textit{the accumulated surrogate bias in each cell},\\ &\sigma_G : C_i \to \{+,-,\varnothing\}, && \textit{the current SPR sign configuration},\\ &\mathsf{L}_G, && \textit{the mutable state of the explicit } A/B \textit{ labeler}. \end{aligned}$$

Initially, we set $\mathrm{bias}_G(c) = 0$ and $\sigma_G(c) = \varnothing$ for every instance G and cell c. The labeler state L_G is initialized by $A_G^{\mathrm{root}} := A.\,\mathrm{initialize}(1, 2^i, 0)$. The complete state of SPR-CALIBRATION *is*

$$S_{\mathrm{local}}^{\mathrm{SPR}} = \left\{\mathrm{bias}_{G_{i,j,\ell}}, \sigma_{G_{i,j,\ell}}, \mathsf{L}_{G_{i,j,\ell}}\right\}_{i\in\mathcal{I}_h, j\in\mathcal{J}_i, \ell\in\{0,1\}}. \tag{18}$$

Moreover, to deal with the reduced transcript (see Definition 3), we need to keep a record of all historical states as follows:

$$S_t^{\mathrm{SPR}} = \{S_{\mathsf{local},s}^{\mathrm{SPR}}\}_{0\leqslant s\leqslant t-1}, \tag{19}$$

where $S_{\mathsf{local},s}^{\mathrm{SPR}}$ is the local state defined as equation 18 at round s.

During the learning process, the state S_t^{SPR} is updated according to the changes in bias_G, σ_G, and L_G for each instance $G = G_{i,j,\ell}$.

A.2 DETERMINISTIC IMPLEMENTATION OF SPR-CALIBRATION

The feature needed by Algorithm 2 is a deterministic one-step transition. Given the pre-round state S (see Definition 2), a round index t, and an input $z \in [0,1]$, define

$$\mathrm{Step}_{\mathrm{SPR}}(S, t, z) = (p, S^+),$$

where p is the forecast returned by one round of SPR-CALIBRATION and S^+ is the resulting state. We fix all loop orders, tie-breaking rules, and boundary conventions below, so this transition is well-defined. We then set $\mathrm{Predict}_{\mathrm{SPR}}(S, t, z) := p$ and $\mathrm{Update}_{\mathrm{SPR}}(S, t, z) := S^+$.

Now we describe how to implement SPR-CALIBRATION in a deterministic manner. The algorithm consists of three key steps.

- **Placement of bias.** At each round t, given the input $z \in [0,1]$ and pre-round state S, the algorithm first attempts to reduce a previously accumulated bias of sufficiently large magnitude. If no such bias-reduction update is available, the algorithm selects an SPR instance on which the cell containing z has sufficiently small current bias and adds the new bias contribution to that cell. In particular, if the selected cell does not carry an SPR sign, the algorithm first invokes `simulateGame` to place one. By the parameter design at difference scales, the bias-placement phase always finds an admissible cell.
- **The `simulateGame` step.** A call to `simulateGame` performs one legal move of the SPR game on an SPR instance whose queried cell is empty. It removes every minus sign strictly to the left of the queried cell and every plus sign strictly to its right, asks the explicit A/B strategy for a deterministic new sign, and records that sign in the queried cell.
- **The A/B Player-L strategy.** Each SPR instance uses the deterministic Player–L strategy of Dagan et al. (2025). It is defined by two mutually recursive procedures, A and B. An A-object controls an interval $[l, r]$ and an integer bias parameter b. At a leaf it returns $\mathrm{sgn}(b)$. At an internal node it delegates to its currently active B-object and restarts that object whenever the latter returns the restart symbol $\perp$. A B-object splits its interval into two halves, stores one child A-object for each half, and uses four phases to compare how often the two halves have been queried. The parameter M is a running guess for the relevant execution length and implements a doubling mechanism.

Algorithm 3 SPR-CALIBRATION

Require: Pre-round state S, round index t, and input $z \in [0, 1]$.

Bias-removal phase.
for $i \in \mathcal{I}_h$ in increasing order **do**
  for $j \in \mathcal{J}_i$ in increasing order **do**
    $(c, G) \leftarrow \text{cell}(i, j, z)$.
    $\mathcal{B} \leftarrow \{c' < c : \text{bias}_G(c') < -1\} \cup \{c' > c : \text{bias}_G(c') > 1\}$. {Find cells with removable bias.}
    if $\mathcal{B} \neq \varnothing$ **then**
      $\bar{c} \leftarrow \min \mathcal{B}$. {Deterministic rule to rank the cells.}
      $\left.\begin{array}{l} \sigma \leftarrow \text{sgn}(\text{bias}_G(\bar{c})), \\ p \leftarrow \text{prob}(\bar{c}, \sigma, G), \\ \text{bias}_G(\bar{c}) \leftarrow \text{bias}_G(\bar{c}) + (z - p). \end{array}\right\}$ {Update the selected SPR instance G.}
      Update S according to equation 19. {Update the state S after updating the SPR instance G.}
      return (p, S).
    end if
  end for
end for
Bias-placement phase.
for $i \in \mathcal{I}_h$ in increasing order **do**
  for $j \in \mathcal{J}_i$ in increasing order **do**
    $(c, G) \leftarrow \text{cell}(i, j, z)$.
    if $|\text{bias}_G(c)| < 2^{j-i}$ **then**
      if $\sigma_G(c) = \varnothing$ **then**
        `simulateGame`(c, G). {Call the SPR instance; the labeling procedure L_G is updated.}
      end if
      $\left.\begin{array}{l} \sigma \leftarrow \sigma_G(c), \\ p \leftarrow \text{prob}(c, \sigma, G), \\ \text{bias}_G(c) \leftarrow \text{bias}_G(c) + (z - p). \end{array}\right\}$ {Update the selected SPR instance G.}
      Update S according to equation 19. {Update the state S after updating the SPR instance G.}
      return (p, S).
    end if
  end for
end for

Algorithm 4 SPR-CALIBRATION (full version)

Initialize S_0 by resetting $\text{bias}_G, \sigma_G, \mathsf{L}_G$ for every $G = G_{i,j,\ell}$ according to Definition 2.
for $t = 1, 2, \ldots, T$ **do**
  Receive the conditional mean z_t
  $(p_t, S_t) \leftarrow$ SPR-CALIBRATION(S_{t-1}, t, z_t)
  Predict p_t.
end for

All counters, phases, child objects, and pointers are part of the labeler state L_G. Thus, after L_G is included in S_t^{SPR}, a call to `simulateGame` is a deterministic function of the copied state and the queried cell.

After completing the three steps above, the algorithm returns the prediction p for the current round and updates its internal state S. The deterministic conventions specified below ensure that the output pair (p, S) is uniquely determined.

Algorithm 5 `simulateGame`(c, G)

Require: An empty cell $c \in C_i = [2^i]$ of an SPR instance $G = G_{i,j,\ell}$.
{Remove the $-$ signs left to c.}
for each $c' \in C_i$ with $c' < c$ **do**
  if $\sigma_G(c') = -$ **then**
    $\sigma_G(c') \leftarrow \varnothing$.
  end if
end for
{Remove the $+$ signs right to c.}
for each $c' \in C_i$ with $c' > c$ **do**
  if $\sigma_G(c') = +$ **then**
    $\sigma_G(c') \leftarrow \varnothing$.
  end if
end for
$s \leftarrow \mathsf{L}_G.\,\mathrm{label}(c)$. {Run the SPR procedure to decide the sign of c.}
$\sigma_G(c) \leftarrow s$.

Algorithm 6 Player–L procedure A (Algorithm 3 in Dagan et al. (2025))

Require: Integers $1 \leqslant l \leqslant r$ and bias parameter $b \in \mathbb{Z}$.
Procedure $A.\,\mathrm{initialize}(l, r, b)$:
if $l < r$ **then**
  $\mathrm{recentB} \leftarrow B.\,\mathrm{initialize}(l, r, b, 1)$.
else
  $\mathrm{recentB} \leftarrow \varnothing$.
end if
$\mathrm{count} \leftarrow 0$.
Procedure $A.\,\mathrm{label}(s)$, where $s \in [l, r]$:
if $l = r$ **then**
  return $\mathrm{sgn}(b)$.
else
  $\mathrm{count} \leftarrow \mathrm{count} + 1$.
  $\sigma \leftarrow \mathrm{recentB}\,.\,\mathrm{label}(s)$.
  if $\sigma = \perp$ **then**
    $\mathrm{recentB} \leftarrow B.\,\mathrm{initialize}(l, r, b, \mathrm{count})$.
    $\mathrm{count} \leftarrow 1$.
    return $\mathrm{recentB}\,.\,\mathrm{label}(s)$.
  else
    return σ.
  end if
end if

Deterministic conventions. All loops are executed in increasing lexicographic order of their indices. If more than one cell is admissible in the bias-removal phase, the smallest admissible cell is selected. Finally, define $\mathrm{sgn}(b) = +$ for $b \geqslant 0$ and $\mathrm{sgn}(b) = -$ for $b < 0$. These conventions make the algorithm deterministic.

B Theoretical Guarantees of SPR-Calibration

We recall several properties of SPR-Calibration from Dagan et al. (2025)[4]. The half-open interval convention used in our presentation only makes the assignment of dyadic boundary points explicit. It

[4]References to Appendix A of Dagan et al. follow the full version .

Algorithm 7 Player–L procedure B (Algorithm 4 in Dagan et al. (2025))

Require: Integers $1 \leqslant l < r$, bias parameter $b \in \mathbb{Z}$, and guess parameter $M \in \mathbb{N}$.

Procedure $B.\,\mathrm{initialize}(l, r, b, M)$:
$m \leftarrow \lfloor (l + r)/2 \rfloor$.
$A[0] \leftarrow A.\,\mathrm{initialize}(l, m, b)$.
$A[1] \leftarrow A.\,\mathrm{initialize}(m + 1, r, b)$.
$\mathrm{prevHalf} \leftarrow -1$.
$\mathrm{countHalf}[0] \leftarrow 0$, $\mathrm{countHalf}[1] \leftarrow 0$.
$\mathrm{phase} \leftarrow 1$.
Procedure $B.\,\mathrm{label}(s)$, where $s \in [l, r]$:
$\mathrm{half} \leftarrow 0$ if $s \leqslant m$, and $\mathrm{half} \leftarrow 1$ otherwise.
$\mathrm{countHalf}[\mathrm{half}] \leftarrow \mathrm{countHalf}[\mathrm{half}] + 1$.
if $\mathrm{phase} = 1$ **then**
  if $\mathrm{countHalf}[\mathrm{half}] = M$ and $M \leqslant \mathrm{countHalf}[1 - \mathrm{half}] \leqslant 2M$ **then**
    $\mathrm{phase} \leftarrow 2$.
  else if $\mathrm{countHalf}[\mathrm{half}] = M$ and $\mathrm{countHalf}[1 - \mathrm{half}] > 2M$ **then**
    $\mathrm{phase} \leftarrow 3$.
  end if
else if $\mathrm{phase} = 2$ **then**
  if $\mathrm{half} \neq \mathrm{prevHalf}$ **then**
    return $\perp$.
  end if
  if $\mathrm{countHalf}[\mathrm{half}] = 2\,\mathrm{countHalf}[1 - \mathrm{half}] + 1$ **then**
    $\mathrm{phase} \leftarrow 3$.
  end if
else if $\mathrm{phase} = 3$ **then**
  if $\mathrm{countHalf}[\mathrm{half}] = \lfloor \mathrm{countHalf}[1 - \mathrm{half}]/2 \rfloor + 1$ **then**
    $\mathrm{phase} \leftarrow 4$.
    if $\mathrm{half} = 0$ **then**
      $A[0] \leftarrow A.\,\mathrm{initialize}(l, m, b + 1)$.
    else
      $A[1] \leftarrow A.\,\mathrm{initialize}(m + 1, r, b - 1)$.
    end if
  end if
else if $\mathrm{phase} = 4$ **then**
  if $\mathrm{countHalf}[\mathrm{half}] > \mathrm{countHalf}[1 - \mathrm{half}]$ **then**
    return $\perp$.
  end if
end if
$\sigma \leftarrow A[\mathrm{half}].\,\mathrm{label}(s)$.
$\mathrm{prevHalf} \leftarrow \mathrm{half}$.
return σ.

preserves the containment, ordering, and dyadic counting properties used in the cited proofs, and therefore does not affect the following bounds.

Let L^{AB} be the deterministic Player–L strategy implemented by Algorithms 6 and 7, with the root A-instance initialized on the n-cell board with bias parameter $b = 0$. We write SimAB_n for the composite transition implemented by `simulateGame` (see Algorithm 5): on a legal query to an empty cell, it first performs the SPR sign-removal step, then advances the A/B labeling state, and finally places the sign returned by that state.

A *full transcript* of this composite process is a finite sequence $H = \big((t_r, c_r, \sigma_r)\big)_{r=1}^{m}$, where $t_1 < \cdots < t_m$ are the ambient calibration rounds, $c_r \in [n]$ is empty immediately before the r-th call, and $\sigma_r \in \{+, -\}$ is the sign returned by the evolving A/B state.

For a signed call (c, σ) and a later queried cell c', define

$$\mathrm{kill}\big((c,\sigma), c'\big) := \begin{cases} 1, & \sigma = - \text{ and } c < c', \\ 1, & \sigma = + \text{ and } c > c', \\ 0, & \text{otherwise.} \end{cases}$$

Thus, $\mathrm{kill}((c, \sigma), c') = 1$ when the sign-removal step of a call to c' erases the sign σ previously placed at c.

Definition 3 (Reduced transcript)**.** *Set $R_0 := \varnothing$. Having constructed R_{r-1}, repeatedly delete the last entry (t, c, σ) of the current list while $\mathrm{kill}\big((c,\sigma), c_r\big) = 1$, and then append (t_r, c_r, σ_r). Let $\mathrm{Red}(H) := R_m$ and $\mathrm{rlen}(H) := |\mathrm{Red}(H)|$. The signs in $\mathrm{Red}(H)$ are inherited from the full transcript. This is a deterministic realization of the adjacent-deletion reduction used in the proof of Lemma A.2 of Dagan et al. (2025).*

By recording the SPR state associated with every prefix of the reduced transcript, the A/B labeling procedure on reduced transcripts is deterministic and well defined. More precisely, at step r, let R_{r-1} be the current reduced transcript and let c_r be the incoming cell. We initialize a temporary transcript $\widetilde{R} \leftarrow R_{r-1}$. When $\widetilde{R}$ is nonempty and its last entry (t, c, σ) is killed by the request c_r, we remove (t, c, σ) from $\widetilde{R}$ and restore the state associated with the shortened transcript. If $\widetilde{R}$ becomes empty, we restore the initial state. Once this deletion-and-rollback procedure terminates, either $\widetilde{R}$ is empty or its last entry is not killed by c_r. We then query the Player–L strategy at c_r using the restored state, obtaining a label σ_r, and define $R_r := \widetilde{R} \,\|\, (t_r, c_r, \sigma_r)$, where $\|$ denotes concatenation. Finally, we record the state resulting from this query as the state associated with the new reduced transcript R_r.

Let $\mathrm{Surv}(H)$ be the number of occupied cells on the actual board after executing the full transcript H.

Definition 4 (Reduced-execution value of the composite A/B procedure)**.** *For $n, m, s \in \mathbb{N}$, define the horizon-restricted value*

$$\mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}(n; m, s) := \max_{H} \mathrm{Surv}(H),$$

where the maximum is over all full legal transcripts H generated by SimAB_n against arbitrary adaptive choices of Player–P, subject to $|H| \leqslant m$ and $\mathrm{rlen}(H) \leqslant s$. We also define the uniform reduced-execution value $\mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}(n, s) := \sup_{1 \leqslant m \leqslant T} \mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}(n; m, s)$.

Lemma 2. *For any input sequence $z_1, \ldots, z_T \in [0, 1]$, if* SPR-CALIBRATION *is run with input $\{z_t\}_{t=1}^{T}$, then the bias term satisfies*

$$\sum_{p \in \mathcal{P}_T} \left| \sum_{t \in [T] : p_t = p} (z_t - p_t) \right| \leqslant O\left(\sum_{i=1}^{\tau-h} \sum_{j=i+1}^{i+h} 2^{j-i}\, \mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}(2^i, 2^{\tau-j+1}) + \sum_{i=1}^{\tau-h} \sum_{j=i+1}^{i+h} \min\{2^i, 2^{\tau-h-i}\} \right).$$

Proof. By the argument of Corollary A.3 in Dagan et al. (2025), the reduced transcript makes at most $2^{\tau-j+1}$ calls to `simulateGame` for the SPR instance $G_{i,j,\ell}$. Suppose that Player–L follows the recursive A/B labeling strategy L^{AB}. Then, by Lemmas A.1 and A.4 of Dagan et al. (2025), together with the definition of $\mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}(n, s)$, the bias contribution generated by the instance $G_{i,j,\ell}$ is bounded by

$$O\Big(2^{j-i}\, \mathrm{Val}_{\mathrm{AB}}^{\mathrm{red}}\big(2^i, 2^{\tau-j+1}\big) + \min\{2^i, 2^{\tau-h-i}\}\Big).$$

Summing this bound over $1 \leqslant i \leqslant \tau - h$, $i + 1 \leqslant j \leqslant i + h$ and $\ell \in \{0, 1\}$ gives the claimed bound.

□

Lemma 3. *For any input sequence $z_1, \dots, z_T \in [0,1]$, it holds that $|\mathcal{P}_T| \leqslant \sum_{i=1}^{\tau} O\left(\min\{2^i, 2^{\tau-h-i}\}\right) = O\left(2^{(\tau-h)/2}\right)$.*

Proof. This is the active-cell counting bound of Lemma A.4 of Dagan et al. (2025). Its dyadic counting argument is unaffected by the endpoint tie-breaking convention fixed above. □

Lemma 4. *There exist constants $C_{\mathrm{AB}} \geqslant 1$ and $\alpha, \beta > 0$ satisfying $q := \alpha + \beta < 1$ such that, for every $n, s \geqslant 1$, $\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant \min\left\{n,\ s,\ C_{\mathrm{AB}} n^{\alpha} s^{\beta}\right\}$. Consequently, defining $\varepsilon_{\mathrm{AB}} := 1 - q > 0$ and $\gamma := \frac{q}{1+q} = \frac{1-\varepsilon_{\mathrm{AB}}}{2-\varepsilon_{\mathrm{AB}}}$, we have the uniform bound $\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant C_{\mathrm{AB}}^{\frac{1}{(1+q)}} (ns)^{\gamma}$. In particular, for every $\rho \geqslant 0$, $\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}\left(n,\ n^{\rho}\right) = O\left(n^{\gamma(1+\rho)}\right)$.*

Proof. Fix $n, s \geqslant 1$, and consider an arbitrary legal play of length $t \leqslant s$. Let R_{σ} denote the number of surviving signs of type $\sigma \in \{-1, +1\}$. The root A-instance of the A/B strategy has bias parameter zero. Therefore, Lemma 5.1 of Dagan et al. (2025) gives constants $C_0, \alpha, \beta > 0$, with $\alpha + \beta < 1$, such that $R_{\sigma} \leqslant C_0 n^{\alpha} t^{\beta}$ for each $\sigma \in \{-1, +1\}$. Summing over the two sign types and using $t \leqslant s$, we obtain

$$R_{-1} + R_{+1} \leqslant 2C_0 n^{\alpha} s^{\beta}.$$

Thus, after setting $C_{\mathrm{AB}} := 2C_0$,

$$\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant C_{\mathrm{AB}} n^{\alpha} s^{\beta}.$$

The bound $\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant \min\{n,s\}$ follows directly from the rules of the SPR game. This proves

$$\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant \min\left\{n,\ s,\ C_{\mathrm{AB}} n^{\alpha} s^{\beta}\right\}.$$

Therefore, we have that

$$\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}(n,s) \leqslant C_{\mathrm{AB}}^{\frac{1}{1+q}} (ns)^{\frac{q}{1+q}} = C_{\mathrm{AB}}^{\frac{1}{1+q}} (ns)^{\gamma}.$$

By choosing $s = n^{\rho}$, we obtain

$$\mathrm{Val}^{\mathrm{red}}_{\mathrm{AB}}\left(n, n^{\rho}\right) \leqslant C_{\mathrm{AB}}^{1/(1+q)} \left(n \cdot n^{\rho}\right)^{\gamma} = O\left(n^{\gamma(1+\rho)}\right).$$

□

Lemma 5. *For any input sequence $z_1, \dots, z_T \in [0,1]$, when* SPR-CALIBRATION *is executed with input $(z_t)_{t=1}^{T}$, the resulting bias term satisfies*

$$\sum_{p \in \mathcal{P}_T} \left| \sum_{t \in [T]: p_t = p} (z_t - p_t) \right| \leqslant O\left(\sum_{i=1}^{\tau-h} \sum_{j=i+1}^{i+h} 2^{j-i+\gamma(i+\tau-j)} + \log_2(T) \cdot \sum_{i=1}^{\tau-h} \min\{2^i, 2^{\tau-h-i}\} \right).$$

Proof. The conclusion follows by Lemma 2 and Lemma 4. □

C COMPUTATIONAL COST OF ALGORITHM 2

We give a fully explicit implementation of Algorithm 2. The purpose of the argument is only to establish a polynomial running time, so we use direct scans and deep copies rather than more sophisticated persistent data structures. Let $K = \lceil 1/\eta \rceil$. Recall also that $\tau = \lceil \log_2 T \rceil$, $\overline{T} = 2^{\tau} < 2T$ and $h \leqslant \tau - 1$.

Lemma 6. *Algorithm 2, using the deterministic transition from Section A.2 and the explicit A/B labeler, can be implemented in time $O\left(T^{\frac{7}{2}} \log T\right)$.*

Proof. We prove by bounding the cost of one deterministic SPR-CALIBRATION transition and the additional work performed by the outer wrapper.

Size of the SPR-CALIBRATION state. For every $i \in \mathcal{I}_h = \{1, \dots, \tau - h\}$, $j \in \mathcal{J}_i = \{i+1, \dots, i+h\}$ and $\ell \in \{0, 1\}$, the algorithm maintains a board $G_{i,j,\ell}$ with 2^i cells. Its bias array and sign array use $O(2^i)$ words.

The live state of the explicit A/B labeler also uses $O(2^i)$ words. Indeed, its recursively stored objects form a binary tree with $O(2^i)$ nodes, and every node stores only a constant number of counters, pointers, phase variables, and integer parameters. When a labeler object is restarted, the old object is discarded, so obsolete versions are not retained.

Thus, one board requires $O(2^i)$ words, and the local SPR-CALIBRATION state has size

$$\sum_{i=1}^{\tau-h} \sum_{j=i+1}^{i+h} \sum_{\ell \in \{0,1\}} O(2^i) = O\left(h \sum_{i=1}^{\tau-h} 2^i \right) = O\big(h 2^{\tau-h}\big) .$$

Taking the historical recording of the local SPR-CALIBRATION states into consideration, the complete SPR-CALIBRATION state has size $O(Th2^{\tau-h})$. The same bound applies to the one-time initialization cost and to the cost of making a deep copy of the state.

Cost of one SPR-CALIBRATION transition. Consider $\mathrm{Step}_{\mathrm{SPR}}(S, t, z)$ for a fixed input $z \in [0, 1]$. In the bias-removal phase, for each pair (i, j), the parity of z selects one of the two boards $G_{i,j,0}$ and $G_{i,j,1}$. A direct scan of that board finds an admissible cell in $O(2^i)$ time if one exists. Hence a complete bias-removal scan costs $O\left(h \sum_{i=1}^{\tau-h} 2^i\right) = O\big(h 2^{\tau-h}\big)$. The bias-placement phase performs only $O(h\tau)$ constant-time current-cell checks, except that it may make one call to `simulateGame`. Suppose that `simulateGame` is called on a board with 2^i cells. Scanning the sign array, constructing the reduced transcripts, and carrying out all else legal deletions costs $O(2^i)$.

It remains to bound the cost of the A/B label query. A call to the recursive A/B procedure follows a root-to-leaf path of depth at most i. At each depth, the corresponding A-object can restart its current B-object at most once before returning a sign. Eagerly initializing a B-object on an interval of length at most 2^i costs $O(2^i)$. Therefore, the deliberately coarse bound $O(i2^i) \leqslant O\big(\tau 2^{\tau-h}\big)$ holds for one label query.

Combining the bias-removal phase, the bias-placement phase, and the possible game simulation gives that the time cost of $\mathrm{Step}_{\mathrm{SPR}}(S, t, z)$ is bounded by $O(\tau 2^{\tau-h})$.

Computing all preview predictions. For each $x \in \mathcal{X}$, the algorithm:

1. makes a deep copy of S_t^{SPR};
2. evaluates $\mathrm{Step}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, x)$ on the scratch copy;
3. records only the resulting prediction $D_t(x)$; and
4. discards the scratch copy.

Consequently, computing all K values $D_t(x)$ costs $O\big(TK\tau 2^{\tau-h}\big)$ time. Because the scratch state is reused, this requires only one scratch copy, rather than K simultaneous copies.

Solving the finite minimax problem. The minimax optimization problem equation 6 has K variables, corresponding to the distribution over the K grid points, and only two constraints, corresponding to the two possible outcomes $y = 0$ and $y = 1$. Therefore, it can be solved within $O(K^2)$ time.

Putting all together. After sampling $\widetilde{e}_t$, the algorithm commits exactly one SPR transition:

$$(p_t, S_{t+1}^{\mathrm{SPR}}) = \mathrm{Step}_{\mathrm{SPR}}(S_t^{\mathrm{SPR}}, t, \widetilde{e}_t).$$

This costs $O\big(\tau 2^{\tau-h}\big)$. Therefore, the total cost of round t is $O\big(TK\tau 2^{\tau-h} + K^2\big)$. Summing over the T rounds and including the one-time initialization cost gives $O\big(T^2K\tau 2^{\tau-h} + TK^2\big)$. Using $K = O(\sqrt{T})$, $\tau = O(\log T)$, and $2^{\tau-h} \leqslant 2^{\tau} < 2T$, we obtain

$$O\left(T^2K\tau 2^{\tau-h} + TK^2\right) = O\Big(T^{\frac{7}{2}} \log T\Big).$$

□

D Algorithm and Analysis for Delayed Calibration

In this section, we describe the delay-adaptive Blackwell-wrapped SPR-Calibration forecaster and prove Theorem 2. The key observation is that the inner SPR instance advances using surrogate inputs rather than realized outcomes. Delayed feedback therefore affects only the residuals maintained by the Blackwell wrapper. To adapt to unknown delays, the forecaster tracks the cumulative number of pending outcomes and restarts the inner instance with a newly tuned prediction-support budget whenever the corresponding delay scale increases.

Appendix D.1 presents the algorithm, including its feedback convention, parameter selection, and restart rule. Appendix D.2 proves the theorem by controlling the residual potential within each phase, accounting for random phase endpoints, and aggregating the resulting calibration bounds.

D.1 Algorithm Description

Let $\mathcal{O}_{t-1}$ be the indices revealed before prediction t, and recall

$$m_t = t - 1 - |\mathcal{O}_{t-1}|, \qquad \mathcal{D}_T = \sum_{t=1}^{T} m_t = \sum_{s=1}^{T} \min\{d_s, T - s\}, \qquad d = \mathcal{D}_T/T. \tag{20}$$

Define $\lambda = \frac{2}{2-\varepsilon_{\mathrm{AB}}}$, and recall $\tau = \log_2 T$. For every admissible parameter $h \in \{1, \ldots, \tau - 1\}$, let $k = 2^{(\tau-h)/2}$. By Lemma 3 and 5, an SPR-Calibration instance initialized for horizon T satisfies: for every input sequence and every prefix $t \leqslant T$, there exists constants C_B and C_P independent of T and h such that

$$\sum_{p \in \mathcal{P}_t} \left| \sum_{\tau \leqslant t : p_\tau = p} (z_\tau - p_\tau) \right| \leqslant C_B \log^2 T \left(\frac{T}{2^{\lambda(\tau-h)/2}} + 2^{(\tau-h)/2} \right), \tag{21}$$

$$|\mathcal{P}_t| \leqslant C_P 2^{(\tau-h)/2}. \tag{22}$$

Remark 2. Equation 21 differs slightly from the statement of Lemma 5, as it concerns the calibration error accumulated over the first t rounds rather than the full horizon T. Nevertheless, the same bound holds up to a constant factor. Indeed, for any $t \leqslant T$, the adversary can choose one of the two constant continuations, either $y_{t'} = 0$ for all $t < t' \leqslant T$ or $y_{t'} = 1$ for all $t < t' \leqslant T$, so that the calibration error at time T is at least half that at time t. This follows because the all-zero continuation preserves the negative residual contributions, whereas the all-one continuation preserves the positive ones.

Algorithm 8 Delay-Adaptive Blackwell-Wrapped SPR-CALIBRATION

Input: Number of rounds T.
Set $\{G_j, h_j\}_{j=0}^{\tau}$ as described, and $\mathcal{X} = \{i/\lceil\sqrt{T}\rceil : 0 \leqslant i \leqslant \lceil\sqrt{T}\rceil\}$.
Initialize $F \leftarrow 0$, $j \leftarrow -1$, and $\mathcal{O} \leftarrow \varnothing$.
for $t = 1, 2, \ldots, T$ **do**
  Set $m_t \leftarrow t - 1 - |\mathcal{O}|$ and $F \leftarrow F + m_t$.
  Set $j_t \leftarrow \min\{r \in \{0, \ldots, \tau\} : T + D \leqslant G_r\}$.
  if $j_t \neq j$ **then**
    Set $j \leftarrow j_t$ and initialize a fresh SPR-CALIBRATION instance $\mathcal{G}^j$ with with horizon T and parameter h_j. Denote $S^{\text{SPR},j}$ as its internal state.
    Initialize the step count $\ell^j = 0$, and the residual table $\hat{R}_p^j \leftarrow 0$ for all p.
  end if
  Set $\ell^j \leftarrow \ell^j + 1$.
  Preview $D_t(x) \leftarrow \text{Predict}_{\text{SPR}}(S^{\text{SPR},j}, \ell^j, x)$ for all $x \in \mathcal{X}$, without changing $S^{\text{SPR},j}$.
  Choose μ_t by solving equation 24.
  Sample $\tilde{e}_t \sim \mu_t$ and output $p_t \leftarrow D_t(\tilde{e}_t)$.
  Set $S^{\text{SPR},j} \leftarrow \text{Update}_{\text{SPR}}(S^{\text{SPR},j}, \ell^j, \tilde{e}_t)$.
  for all indices s whose outcomes y_s are revealed at the end of round t **do**
    Set $\mathcal{O} \leftarrow \mathcal{O} \cup \{s\}$.
    if $j_s = j$ **then**
      Set $\hat{R}_{p_s}^j \leftarrow \hat{R}_{p_s}^j + y_s - \tilde{e}_s$.
    end if
  end for
end for

For phase indices $j = 0, \ldots, \tau$, define $G_j = 2^j T$ and $u_j = \left(\frac{T^2}{G_j}\right)^{1/(2\lambda+1)}$. Choose $h_j = \max\{\min\{\tau - 2\log_2 u_j, \tau - 1\}, 1\}$ and set $k_j = 2^{(\tau - h_j)/2}$. Then $k_j \asymp u_j$, with uniform constant factors. The choice balances T/k_j^λ and $\sqrt{k_j G_j}$. As the delay budget increases, the prediction-support budget decreases.

Let $F_t = \sum_{s=1}^{t} m_s$. Before predicting at round t, choose

$$j_t = \min\{j \in \{0, \ldots, \tau\} : T + F_t \leqslant G_j\}. \tag{23}$$

Because $T + F_t \leqslant T + T(T-1)/2 \leqslant T^2 = G_\tau$, j_t is well defined and nondecreasing in t. Throughout the algorithm, we maintain at most $\tau + 1$ instances of SPR-CALIBRATION. The j-th instance uses parameter $h = h_j$, with all other settings identical across instances. At each round t, exactly one instance is active, with its index j_t determined by equation 23.

The delay-adaptive algorithm. Set $N = \lceil\sqrt{T}\rceil$, $\eta = 1/N$, and $\mathcal{X} = \{0, 1/N, \ldots, 1\}$. At each step t, maintain

$$\hat{R}_{p,t-1}^{j_t} = \sum_{\substack{s<t: j_s = j_t,\ p_s = p \\ s \in \mathcal{O}_{t-1}}} (y_s - \tilde{e}_s).$$

Residuals at previously unused predictions are zero. Given the current state S^{SPR,j_t} of the active SPR-CALIBRATION instance, and local round index ℓ^{j_t}, define $D_t(x) = \text{Predict}_{\text{SPR}}(S^{\text{SPR},j_t}, \ell^{j_t}, x)$. Choose

$$\mu_t \in \arg\min_{\mu \in \Delta(\mathcal{X})} \max_{y \in \{0,1\}} \sum_{x \in \mathcal{X}} \mu(x) \hat{R}_{D_t(x),t-1}^{j_t}(y - x). \tag{24}$$

D.2 Proof of Theorem 2

Proof of Theorem 2. We first bound the calibration error within each phase and then aggregate over the random set of phases. The established guarantees from SPR-CALIBRATION control the surrogate bias, while a delayed-feedback potential argument bounds the discrepancy between surrogate inputs and realized outcomes.

One-step Blackwell inequality. For fixed pre-round quantities, finite minimax duality gives

$$\min_{\mu\in\Delta(\mathcal{X})}\max_{y\in\{0,1\}}\sum_x \mu(x)\widehat{R}^{j_t}_{D_t(x),t-1}(y-x) = \max_{q\in[0,1]}\min_{x\in\mathcal{X}}\widehat{R}^{j_t}_{D_t(x),t-1}(q-x).$$

For each $q\in[0,1]$, choose a grid point x_q with $|q-x_q|\leqslant\eta$. Evaluating the inner minimum at x_q shows that its value is at most $\eta\|\widehat{R}^{j_t}_{t-1}\|_\infty$. Consequently, for either outcome $y\in\{0,1\}$,

$$\mathbb{E}_{z\sim\mu_t}\left[\widehat{R}^{j_t}_{D_t(z),t-1}(y-z)\right]\leqslant\eta\|\widehat{R}^{j_t}_{t-1}\|_\infty. \tag{25}$$

Full residuals and the delay penalty. Define $I_j=\{t:j_t=j\}$ and $A_j=\{I_j\neq\varnothing\}$. Define the analytical full residual as $R^j_{p,t}=\sum_{\substack{s\leqslant t:j_s=j\\ p_s=p}}(y_s-\tilde{e}_s)$ and the potential as $\Phi_{j,t}=\sum_p\big(R^j_{p,t}\big)^2$.

For an active round $t\in I_j$, the missing residual consists only of pending outcomes from this phase. Therefore

$$\begin{aligned}&\|R^j_{t-1}-\widehat{R}^j_{t-1}\|_\infty\leqslant m_t,\\ &\|\widehat{R}^j_{t-1}\|_\infty\leqslant\sqrt{\Phi_{j,t-1}}+m_t.\end{aligned} \tag{26}$$

Let $\mathcal{W}_t$ be the sigma-field generated by the complete past, including the forecaster's past internal variables and unrevealed past outcomes, together with y_t. The fresh random draw used to sample $\widetilde{e}_t$ is independent of this information, and its conditional distribution is μ_t. Writing $I_{j,t}=\mathbf{1}\{j_t=j\}$, the one-coordinate update gives

$$\Phi_{j,t}-\Phi_{j,t-1}=I_{j,t}\left(2R^j_{p_t,t-1}(y_t-\tilde{e}_t)+(y_t-\tilde{e}_t)^2\right).$$

By equation 25 and 26, and the fact that $|y_t-\tilde{e}_t|\leqslant1$, we obtain

$$\mathbb{E}[\Phi_{j,t}-\Phi_{j,t-1}\mid\mathcal{W}_t]\leqslant I_{j,t}\left(1+2\eta\sqrt{\Phi_{j,t-1}}+2(1+\eta)m_t\right). \tag{27}$$

Random phase endpoints. Set $Z_{j,t}=\mathbb{E}[\Phi_{j,t}]$ and $M_j=\max_{0\leqslant t\leqslant T}\sqrt{Z_{j,t}}$. Cauchy–Schwarz gives

$$\mathbb{E}[I_{j,t}\sqrt{\Phi_{j,t-1}}]\leqslant\sqrt{\mathbb{E}[I_{j,t}\Phi_{j,t-1}]}\leqslant\sqrt{\mathbb{E}[\Phi_{j,t-1}]}\leqslant M_j.$$

Sum equation 27 over t, we have that

$$Z_{j,t}\leqslant\sum_{\tau=1}^{t-1}(1+2(1+\eta)m_j+2\eta M_j)\leqslant t+2(1+\eta)G_j+2\eta tM_j.$$

As a result, $M_j^2\leqslant T+2(1+\eta)G_j+2\eta TM_j$. Because $\eta\leqslant1$, this yields $M_j^2\leqslant(T+4G_j)+2\eta TM_j$.

Solving the quadratic inequality and using $\eta T\leqslant\sqrt{T}$ and $G_j\geqslant T$ gives

$$M_j\leqslant\left(\eta T+\sqrt{\eta^2T^2+T+4G_j}\right)\leqslant4\sqrt{G_j}.$$

Hence $\mathbb{E}[\Phi_{j,T}]\leqslant16G_j$ and $\mathbb{E}[\sqrt{\Phi_{j,T}}]\leqslant4\sqrt{G_j}$.

Calibration error within a phase. Let $\mathcal{P}^j = \{p_t : t \in I_j\}$, and define

$$\mathrm{CalErr}^{(j)} = \sum_{p \in \mathcal{P}^j} \left| \sum_{t \in I_j : p_t = p} (y_t - p_t) \right|.$$

By equation 21 and equation 22, we have

$$\mathrm{CalErr}^{(j)} \leqslant \sum_{p \in \mathcal{P}^j} \left| \sum_{t \in I_j : p_t = p} (y_t - \tilde{e}_t \right| + \sum_{p \in \mathcal{P}^j} \left| \sum_{t \in I_j : p_t = p} (\tilde{e}_t - p_t \right| \leqslant C_B \log^2(T) \left(\frac{T}{k_j^\lambda} + k_j \right) + \sqrt{C_P k_j} \sqrt{\Phi_{j,T}}.$$

Taking expectations, we have that

$$\mathbb{E}[\mathrm{CalErr}^{(j)}] \leqslant C \log^2(T) \left(\frac{T}{k_j^\lambda} + k_j + \sqrt{k_j G_j} \right) \tag{28}$$

for some large enough constant C.

Recall that $\lambda = \frac{2}{2 - \varepsilon_{\mathrm{AB}}}$, $u_j = (T^2/G_j)^{1/(2\lambda+1)}$, $h_j = \max\{\min\{\tau - 2\log_2 u_j, \tau - 1\}, 1\}$ and $k_j = 2^{(\tau - h_j)/2}$. We then have that

$$\mathbb{E}[\mathrm{CalErr}^{(j)}] \leqslant C \log^2(T) T^{\frac{1}{2\lambda+1}} G_j^{\frac{\lambda}{2\lambda+1}}. \tag{29}$$

Putting everything together. By the triangle inequality, we have that

$$\mathrm{CalErr}_T \leqslant \sum_j \mathrm{CalErr}^{(j)}.$$

Let $J = j_T$ be the index of the final phase." There are at most $\tau + 1$ phases. Therefore

$$\begin{aligned}
\mathbb{E}[\mathrm{CalErr}_T] &\leqslant \sum_j \mathbb{E}[\mathrm{CalErr}^{(j)}] \\
&\leqslant \sum_j \mathbb{E}\left[\mathbb{I}[A_j]\right] C \log^2(T) T^{\frac{1}{2\lambda+1}} G_j^{\frac{\lambda}{2\lambda+1}} \\
&\leqslant 10C \log^2(T) T^{\frac{1}{2\lambda+1}} \mathbb{E}\left[G_J^{\frac{\lambda}{2\lambda+1}} \right].
\end{aligned}$$

By definition of j_t for each $t \in [T]$, we have that $G_J \leqslant 2(T + \mathcal{D}_T)$. By Jensen's inequality and the fact that $\frac{\lambda}{2\lambda+1} < 1$, we have that

$$\mathbb{E}[\mathrm{CalErr}_T] \leqslant C \log^2(T)(\log_2 T + 1) T^{\frac{1}{2\lambda+1}} 2(T + \mathbb{E}[\mathcal{D}_T])^{\frac{\lambda}{2\lambda+1}}.$$

Because $\lambda = \frac{2}{2 - \varepsilon_{\mathrm{AB}}}$ and $d = \mathbb{E}[\mathcal{D}_T]/T$, we conclude that

$$\mathbb{E}[\mathrm{CalErr}_T] \leqslant O\Big(\log^2(2T)\, T^{\frac{2}{3} - \frac{\varepsilon_{\mathrm{AB}}}{18 - 3\varepsilon_{\mathrm{AB}}}} (1 + d)^{\frac{1}{3} + \frac{\varepsilon_{\mathrm{AB}}}{18 - 3\varepsilon_{\mathrm{AB}}}} \Big).$$

Computational efficiency. At each round t, it suffices to maintain only the active SPR-CALIBRATION instance indexed by j_t. Thus, the total computational cost remains $O(T^{7/2} \log T)$, matching that of Algorithm 2.

□