Online Conformal Prediction via Universal Portfolio Algorithms

Tuo Liu
KAUST

Edgar Dobriban
University of Pennsylvania

Francesco Orabona
KAUST

Abstract

Online conformal prediction (OCP) seeks prediction intervals that achieve long-run $1-\alpha$ coverage for arbitrary (possibly adversarial) data streams, while remaining as informative as possible. Existing OCP methods often require manual learning-rate tuning to work well, and may also require algorithm-specific analyses. Here, we develop a general regret-to-coverage theory for interval-valued OCP based on the $(1-\alpha)$-pinball loss. Our first contribution is to identify *linearized regret* as a key notion, showing that controlling it implies coverage bounds for any online algorithm. This relies on a black-box reduction that depends only on the Fenchel conjugate of an upper bound on the linearized regret. Building on this theory, we propose UP-OCP, a parameter-free method for OCP, via a reduction to a two-asset portfolio selection problem, leveraging universal portfolio algorithms. We show strong finite-time bounds on the miscoverage of UP-OCP, even for polynomially growing predictions. Extensive experiments support that UP-OCP delivers consistently better size/coverage tradeoffs than prior online conformal baselines.

1 Introduction

Conformal Prediction (CP) (Vovk et al., 2005) constructs prediction sets with finite-sample coverage guarantees, wrapping around any model to ensure the true label is contained with probability at least $1-\alpha$. While CP in its basic form requires exchangeable data, real-world data streams often violate this assumption.

Online Conformal Prediction (OCP) aims to completely do away with making assumptions on the data, and consider instead *deterministic* and *adversarial* data (see, e.g., Gibbs and Candes, 2021; Zaffran et al., 2022; Bastani et al., 2022; Gibbs and Candès, 2024; Angelopoulos et al., 2023; Podkopaev et al., 2024). In this setting, one aims to achieve coverage $1-\alpha$ averaged over time (and any possible algorithmic randomness). This work can be viewed to belong to the setting of *online learning* (Cesa-Bianchi and Lugosi, 2006; Hazan, 2016; Orabona, 2019).

More formally, the observed data $((X_t, Y_t))_{t\geq 1}$ arrive over time $t = 1, 2, \ldots$. At each time point (or round) t, a prediction set $\hat{C}_t$ is constructed for Y_t using all the previously observed datapoints $((X_i, Y_i))_{i\leq t-1}$, as well as the current features X_t. Let $\hat{Y}_t$ represent a point prediction given by a model $\hat{f}_t$ trained using all the information available before the true response Y_t. We are interested in regression problems, focusing on the perhaps most popular form of centered prediction sets (see, e.g., Lei et al., 2018): $\hat{C}_t(b) := [\hat{Y}_t - b, \hat{Y}_t + b]$, defined as empty if $b < 0$. The goal is to design a conformal predictor whose observed long-term miscoverage rate is close to the nominal level, denoted as $\alpha \in (0, 1)$. Formally, we aim to construct a sequence of radii $(b_t)_{t\geq 1}$ so that, as the time horizon T grows, the corresponding prediction sets satisfy

$$\lim_{T\to\infty} \left| \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}\left\{ Y_t \notin \hat{C}_t(b_t) \right\} - \alpha \right| = 0 \ . \qquad (1)$$

We discuss the related work on this topic in Appendix A.

Contributions. We answer the following questions:

Is there a form of regret that implies coverage (see (1))?

In Section 3, we show that the general notion of *linearized regret* directly implies coverage. Since linearized regret bounds have been established for several online learning methods, this will enable us to directly obtain coverage guarantees for these methods. In particular, we answer this question by making a connection to the regret-reward duality from online

Workshop "Towards Trustworthy Predictions: Theory and Applications of Calibration for Modern AI" at AISTATS 2026, Tangier, Morocco. Copyright 2026 by the author(s).

learning (Orabona, 2019). Once a bound on the miscoverage is established, it is natural to ask about optimality:

Is it possible to construct an optimal online algorithm (in terms of regret and coverage) for OCP?

This remained unexplored in the past literature. In Section 4, we design a parameter-free strategy that guarantees the best known finite-time coverage guarantee. Following classical work in parameter-free online learning (see, e.g., Orabona, 2019), this is achieved by observing the equivalence between the OCP problem and a gambling one, then using universal portfolio algorithms Cover and Ordentlich (2002) to optimally solve the gambling problem. We will also show that our algorithm guarantees online coverage with any polynomial growth of the nonconformity scores. We call the resulting algorithm Universal Portfolio OCP (UP-OCP).

Finally, we introduce a way to empirically quantify the *trade-off* between size and coverage, for a wide range of values of α. In our extensive experiments on real and simulated datasets, UP-OCP achieves the best such trade-off among a number of strong baselines (Section 5).

2 Notation and Problem Setup

Notation For a function $f : \mathbb{R} \to \mathbb{R}$, we define a *subgradient* of f in $x \in \mathbb{R}$ as $g \in \mathbb{R}$ that satisfies $f(y) \geq f(x) + g(y - x),\ \forall y \in \mathbb{R}$. The set of subgradients of f in x is called the *subdifferential set* and we denote it by $\partial f(x)$. The *indicator function of the set* $\mathcal{V}$, $\mathbf{1}_{\mathcal{V}} : \mathbb{R} \to (-\infty, +\infty]$, has value 0 for $x \in \mathcal{V}$ and $+\infty$ otherwise. For a function $f : \mathbb{R} \to [-\infty, \infty]$, we define the *Fenchel conjugate* $f^\star : \mathbb{R} \to [-\infty, \infty]$ as $f^\star(\theta) = \sup_{x\in\mathbb{R}} (\theta x - f(x))$. The Fenchel conjugate is always well-defined and convex.

Problem Setup. We consider the problem of OCP, for arbitrary data streams, even adversarially generated ones, as introduced in Section 1. Let $S_t \geq 0$ denote the radius of the smallest prediction set that contains the true response Y_t, i.e., $S_t := \inf\{b \in [0, \infty) : Y_t \in \hat{C}_t(b)\} = |Y_t - \hat{Y}_t|$. We will also refer to S_t as the non-conformity score Vovk et al. (2005). In terms of S_t, the target property (1) is equivalent to

$$\lim_{T\to\infty} \left| \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}\{b_t < S_t\} - \alpha \right| = 0 \ .$$

This can be viewed as the problem of sequentially learning the $(1 - \alpha)$-th quantile of the nonconformity scores $(S_t)_{t\geq 1}$.

A standard approach (see, e.g., Gibbs and Candes, 2021; Podkopaev et al., 2024), to learn this quantile is to use a proper scoring rule (see, e.g., Gneiting and Raftery, 2007), namely the *pinball (or quantile) loss*, defined as

$$\ell^{(1-\alpha)}(b, S) := \max\{(1 - \alpha)(S - b), \alpha(b - S)\}\ , \quad (2)$$

where S is the non-conformity score and b is the radius of the prediction interval. This loss is convex and L-Lipschitz in the first argument, where $L := \max\{1 - \alpha, \alpha\}$. These two properties make it online learnable (see, e.g., Hazan, 2016; Orabona, 2019; Cesa-Bianchi and Orabona, 2021). For $t \geq 1$, let b_t be the prediction, and $\ell_t(b_t) := \ell^{(1-\alpha)}(b_t, S_t)$ be the loss at round t. The *regret* of the algorithm with respect to any fixed comparator $u \in \mathbb{R}$ is defined as

$$\mathrm{Regret}_T(u) := \sum_{t=1}^{T} \ell_t(b_t) - \sum_{t=1}^{T} \ell_t(u)\ . \quad (3)$$

The subdifferential set of ℓ_t is

$$\partial\ell_t(b) = \begin{cases} \{\mathbf{1}\{b \geq S_t\} - (1 - \alpha)\}, & b \neq S_t \\ [-(1 - \alpha), \alpha], & b = S_t\ . \end{cases}$$

At $b = S_t$, there is an infinite set of subgradients. Throughout, we adopt the convention of selecting the right subgradient $g_t = \alpha$ when $b_t = S_t$, so that $g_t \in \{-(1-\alpha), \alpha\}$ for all t. Other choices are possible and essentially equivalent. Thus, an online learning algorithm predicting b_t and receiving the pinball loss ℓ_t will receive the subgradient

$$g_t = \mathbf{1}\{b_t \geq S_t\} - (1 - \alpha)\ . \quad (4)$$

As explained in Gibbs and Candes (2021); Angelopoulos et al. (2025) the miscoverage error is closely related to the observed subgradients, because

$$\mathrm{MisCov}_T := \left| \frac{\sum_{t=1}^{T} \mathbf{1}\{b_t \geq S_t\}}{T} - (1 - \alpha) \right| = \frac{\left|\sum_{t=1}^{T} g_t\right|}{T}. \quad (5)$$

3 Coverage from No-Regret

In this section, we describe our main result providing a coverage guarantee for any online algorithm controlling an appropriate form of linearized regret.

Consider an online learning algorithm that in each round $t = 1, 2, \ldots$ produces an action $b_t \in \mathbb{R}$, and let $g_t \in \partial\ell_t(b_t)$ denote a subgradient of the loss at round t. We consider the *linearized regret* (Gordon, 1999; Zinkevich, 2003) of the algorithm on this sequence at the action $u \in \mathbb{R}$, defined as

$$\mathrm{LinRegret}_T(u) := \sum_{t=1}^{T} g_t(b_t - u)\ . \quad (6)$$

In contrast to the standard notion of regret from (3), this quantity sums up to the linearizations $g_t(b_t - u)$ of the loss differences $\ell_t(b_t) - \ell_t(u)$ around b_t. By the definition of subgradients, we have that $\text{Regret}_T(u) \leq \text{LinRegret}_T(u)$ for all u. Thus, *any algorithm that controls the linearized regret also controls the usual regret.* However, an algorithm may control regret but not linearized regret. Crucially, *our analysis shows that controlling the linearized regret suffices to ensure coverage.*

Specifically, we have the following result which bounds the range of the sum of the gradients $\sum_{t=1}^{T} g_t$ depends on the Fenchel conjugate of a bound on the regret function (proof in Appendix D). Due to (4), this immediately implies a bound on the coverage.

Theorem 3.1. *For an online learning algorithm, let $F_T : \mathbb{R} \to \mathbb{R}$ such that $\text{LinRegret}_T(u) \leq F_T(u)$ on the pinball losses $(\ell_t)_{1 \leq t \leq T}$. Then,*

$$-\sum_{t=1}^{T} g_t \in \left\{ z \in \mathbb{R} : F_T^\star(z) \leq (1-\alpha) \sum_{t=1}^{T} S_t \right\}, \quad (7)$$

where $F_T^\star(\cdot)$ is the Fenchel conjugate of $F_T(\cdot)$.

In Appendix F, we show that a simpler asymptotic coverage result can be obtained more directly from our theory, without requiring the machinery of Fenchel conjugates. Moreover, since, as we discussed, coverage can be achieved in trivial ways, it is also important to have other correctness guarantees. For this reason, in the standard conformal prediction setting of i.i.d. scores, we show in Appendix G that any no-regret algorithm ensures that the averaged thresholds b_t converge to the optimal one. This provides an additional desired correctness guarantee in our framework.

We note that our use of linearized regret is not in contradiction with the example in Angelopoulos et al. (2025, Example 1), which shows that standard sublinear regret does not imply coverage; see Appendix B for details. We also discuss therein how our approach relates to proximal regret (Cai et al., 2024) and no-move regret (Angelopoulos et al., 2025).

4 Universal-Portfolio Based Strategy

Thanks to theorem 3.1, we now have a direct relationship between the linearized regret of an online algorithm and its miscoverage error, through the Fenchel conjugate $F_T^\star$. Since a tighter (smaller) regret bound F_T corresponds to a larger, steeper $F_T^\star$, minimizing regret leads to better coverage bounds. Thus, for a fast bound on coverage, it is desirable to use an online algorithm with optimal regret.

It is known that the optimal linearized regret in unconstrained online learning is achieved only by parameter-free algorithms, as the ones in Zhang et al. (2024) and Podkopaev et al. (2024), see the lower bound in Orabona (2019, Section 5.3). However, these algorithms are not fully optimal for OCP. The reason is that they implicitly assume a degree of symmetry (as explained below), whereas the coverage problem is inherently asymmetric: The target miscoverage rate α is typically chosen to be small (e.g., $\alpha = 0.05$ or 0.01), implying that the positive and negative subgradients of the losses are very different.

Instead, we propose reducing our problem to a portfolio selection problem, and leveraging the *Universal Portfolio* (UP) algorithm (Cover and Ordentlich, 2002) for OCP. In the following, we explain how UP methods lead to optimal solutions to online learning problems with asymmetric subgradients. For our reduction, considering pinball losses, we construct a market with two synthetic stocks, whose market gains are driven by the observed miscoverage.

Definition 4.1 (The Conformal Market)**.** For a miscoverage rate $\alpha \in (0,1)$, given the subgradient $g_t \in \{-(1-\alpha), \alpha\}$, $t \geq 1$ defined in (4), we define the vector of *returns* $\boldsymbol{w}_t = (w_{t,1}, w_{t,2})^\top \in \mathbb{R}^2$ of two synthetic stocks as

$$w_{t,1} = -g_t/\alpha + 1, \quad w_{t,2} = 1 + g_t/(1-\alpha) \ .$$

The returns are the coordinates of a *market gain vector* $\boldsymbol{w}_t$ representing the ratio of the closing price to the opening price for the two stocks.

We have $w_{t,1}, w_{t,2} \geq 0$, because $-g_t \in \{-\alpha, 1-\alpha\}$. Stock 1 yields high returns when coverage is lost ($g_t < 0$), while Stock 2 yields moderate returns when coverage is maintained ($g_t > 0$).

We can now formally define the wealth of an algorithm operating in this market.

Definition 4.2 (Wealth Process)**.** Consider an online algorithm that, at each round t, chooses a portfolio weight $\lambda_t \in [0,1]$ representing the fraction of capital invested in Stock 1. The *wealth* W_t is defined as $W_0 = 1$ and

$$W_t = W_{t-1} \cdot (\lambda_t w_{t,1} + (1-\lambda_t) w_{t,2}) \ . \quad (8)$$

The Universal Portfolio algorithm computes the weight λ_t as the wealth-weighted average over the simplex of all possible constant portfolios. Let $W_{t-1}(\lambda) = \prod_{i=1}^{t-1}(\lambda w_{i,1} + (1-\lambda) w_{i,2})$ be the wealth of a constant portfolio λ. Given a prior μ over $\lambda \in [0,1]$, the prediction is

$$\lambda_t = \frac{\int_0^1 \lambda \cdot W_{t-1}(\lambda)\, \mathrm{d}\mu(\lambda)}{\int_0^1 W_{t-1}(\lambda)\, \mathrm{d}\mu(\lambda)} \ . \quad (9)$$

Choosing $d\mu(\lambda) = 1/[\pi\sqrt{\lambda(1-\lambda)}]d\lambda$, Cover and Ordentlich (2002) proved that the log wealth of the al-

gorithm is at least the log wealth of the best constant λ in each round, up to a slack of $\frac{1}{2}\ln(\pi(T+1))$. This regret guarantee is optimal up to constant additive factors (Cover and Ordentlich, 2002).

The critical insight formalized in the following Theorem—which follows from Orabona and Jun (2023, Lemma 1) and the standard reduction of OCO to coin betting (Orabona, 2019)—is that *maximizing the logarithmic growth of this wealth is equivalent to minimizing the linearized regret on the pinball loss.* This allows us to translate the wealth guarantees of portfolio algorithms to coverage through (7). The proof is in Appendix H.

Theorem 4.3 (Regret of UP-OCP)**.** *Let $\mathcal{A}$ be the universal portfolio algorithm with $d\mu(\lambda) = \frac{1}{\pi\sqrt{\lambda(1-\lambda)}}d\lambda$ that outputs weights $(\lambda_t)_{t\geq 1}$ on the conformal market weights $\boldsymbol{w}_t \in \mathbb{R}^2$. Define b_t via*

$$b_t = W_{t-1} \cdot [-(1-\alpha)^{-1} + \lambda_t/(\alpha(1-\alpha))] \ . \qquad (10)$$

Then, the resulting sequence of actions $(b_t)_{t\geq 1}$ achieves linearized regret $\mathrm{LinRegret}_T(u) \leq F_T(|u|)$ for all $u \in \mathbb{R}$ in online quantile loss minimization, where $F_T(|u|)$ is

$$\max\left\{|u|\sqrt{2T\alpha(1-\alpha)\ln(4(T+1)^{\frac{3/2}{\alpha}}(1-\alpha)u^2+1)},\right.$$
$$\left.\frac{4}{3}|u|\left(\ln(3|u|\sqrt{T+1})-1\right)\right\} \ .$$

In the worst case, the regret is of order $\mathcal{O}(\sqrt{T\ln T})$. However, when $\alpha \to 1$, the bound is of order $\ln T$. In Appendix I, we also show that the regret upper bound for UP-OCP is better than the one for the KT approach from Podkopaev et al. (2024).

Coverage Guarantee for UP-OCP We now provide a coverage guarantee for the proposed UP-OCP strategy. We also quantify its advantage over the Krichevsky-Trofimov (KT) bettor, particularly in the regime of small α.

Theorem 4.4 (Coverage bound for UP-OCP)**.** *Let $\alpha \in (0,1)$ and let $(g_t)_{t=1}^T$ be the sequence of subgradients observed by UP-OCP, with $g_t \in \{-(1-\alpha), \alpha\}$. Let $D > 0$ and $q \geq 0$, and assume that $S_t \leq Dt^q$ for all t. For every integer $T \geq 1$, define*

$$\varepsilon_T := \frac{1}{T}\left[\ln\left(1+\frac{(1-\alpha)D(T+1)^{q+1}}{q+1}\right)+\frac{1}{2}\ln(\pi(T+1))\right] \ .$$

Then, $\mathrm{MisCov}_T = |\frac{1}{T}\sum_{t=1}^T g_t| \leq \varepsilon_T + \sqrt{2\alpha(1-\alpha)\varepsilon_T}$.

In Appendix C, we provide a detailed comparison of the coverage rates of UP-OCP with OSD and KT. In summary, OSD has a worse dependency on T when the scores grow ($q > 0$) and requires tuning a stepsize that depends on unknown quantities. KT has the same dependency on T, D, and q, but is designed for symmetric outcomes ($\alpha = 1/2$) and does not improve when $\alpha \to 0$ or $\alpha \to 1$, unlike UP-OCP whose rate approaches $(\ln T)/T$ in those regimes.

Algorithm 1 Universal Portfolio for OCP (UP-OCP)

Input: Target miscoverage rate $\alpha \in (0,1)$
Initialize: Wealth $W_0 \leftarrow 1$, miscoverage count $N \leftarrow 0$
for $t = 1, 2, \ldots$ **do**
  $\lambda_t \leftarrow \dfrac{N+1/2}{t}$; $b_t \leftarrow \max\left(0, W_{t-1} \cdot \dfrac{\lambda_t - \alpha}{\alpha(1-\alpha)}\right)$
  Output prediction set $\hat{C}_t \leftarrow [\hat{Y}_t - b_t, \hat{Y}_t + b_t]$
  Observe true label Y_t
  Compute nonconformity score $S_t \leftarrow |Y_t - \hat{Y}_t|$
  if $S_t > b_t$ **then**
    $N \leftarrow N+1$; $W_t \leftarrow W_{t-1} \cdot \lambda_t/\alpha$
  else
    $W_t \leftarrow W_{t-1} \cdot (1-\lambda_t)/(1-\alpha)$
  end if
end for

Closed-Form Update for Universal Portfolios For our specific Conformal Market defined in Definition 4.1, there is a simple closed-form update $\lambda_t = \frac{1}{t}(\sum_{i=1}^{t-1}\mathbf{1}\{g_i = -(1-\alpha)\} + \frac{1}{2})$, see Appendix J. Substituting this into (10) yields a parameter-free update rule for the conformal radius b_t that adapts to the asymmetry of the gradients. See Algorithm 1 for the complete pseudocode.

The mapping from wealth to radius in (10) can produce negative values. So, in Algorithm 1, we clip the radius to zero. In Appendix I.2, we show that the regret of the truncated sequence is upper bounded by the regret of the original sequence, and that the subgradients remain unchanged, preserving the theoretical guarantees.

5 Experiments

We support our theoretical findings through evaluations on both synthetic data and empirical time series. Our experiments cover the finance and energy domains, where data can be highly non-stationary. Across these settings, we compare the proposed UP-OCP method (Algorithm 1) with state-of-the-art parameter-free and tuned baselines.

Datasets and Models. We follow the OCP literature (Gibbs and Candes, 2021; Angelopoulos et al., 2023; Gibbs and Candès, 2024; Podkopaev et al., 2024) and use standard benchmark datasets: (1) daily open-

Table 1: Quantitative Comparison on the AXP Dataset. Performance metrics for UP-OCP versus parameter-free (KT, DtACI) baselines and tuned SF-OGD (lr=25).

Metric	UP-OCP	KT	DtACI	SF-OGD
Marginal Coverage	0.932	0.920	0.956	0.948
Longest Err. Seq.	**4**	15	6	**3**
Avg. Set Size	**14.8**	16.9	∞	**16.4**
Median Set Size	**11.5**	12.9	12.6	13.8
75% Quantile Size	**18.8**	24.9	21.8	21.5
90% Quantile Size	**32.3**	32.5	∞	**32.3**
95% Quantile Size	38	36.1	∞	36.1

ing prices (log-scale) of four US stocks from 2008 to 2018 (Nguyen, 2018), predicted with `Prophet` (Taylor and Letham, 2018); (2) electricity demand from New South Wales (Harries, 1999), with an AR(3) model; and (3) synthetic nonconformity scores following Angelopoulos et al. (2023, Appendix F.5). We focus our main analysis on the American Express (AXP) dataset; full results are in the Appendix.

Baselines. We compare UP-OCP with: KT (Podkopaev et al., 2024), DtACI Gibbs and Candès (2024), SF-OGD (Orabona and Pál, 2018), and Conformal P/PI Control (Angelopoulos et al., 2023). UP-OCP, KT, and DtACI are parameter-free; for the rest, we report the best ex-post tuned hyperparameters, granting them an oracle advantage (details in Appendix K).

We focus on the trade-off between coverage and prediction set sizes via Pareto frontier plots, since coverage alone is meaningless (a trivial predictor can achieve it; see Appendix K).

5.1 Results for AXP

We begin by evaluating the efficiency-coverage trade-off on the American Express (AXP) dataset, setting $\alpha = 0.05$. An initial warm-up period of 100 days is used for training the initial base forecaster. Table 1 reports the performance metrics. Due to space limitations, the full results for control-based methods are in Appendix N.

UP-OCP achieves valid coverage (93%) with the shortest error sequence (4 days, matching the best tuned SF-OGD), while KT suffers from 15 consecutive miscoverage days. UP-OCP achieves the smallest set sizes across average, median, and 75% quantile metrics. DtACI's 90% quantile size is infinite, indicating reliance on trivial predictions. Overall, UP-OCP matches oracle-tuned SF-OGD while being fully parameter-free. Local coverage plots in Appendix N confirm that UP-OCP tracks the 95% target tightly.

Pareto Frontiers. We use the *average* set size as our primary metric, as it captures unreasonably large prediction sets better than quantiles.

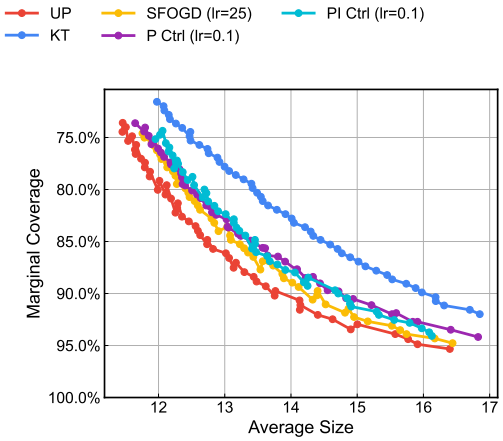

Figure 1: Pareto frontiers for average prediction set size on the AXP Dataset, for 50 target miscoverage rates α uniformly from 0.05 to 0.25. Better performance is closer to the bottom-left corner.

Figure 1 shows that UP-OCP (red) achieves the best Pareto trade-off, consistently achieving the smallest average set size for any target coverage in $[0.75, 0.95]$. KT (blue) is strictly suboptimal.[1] UP-OCP also tracks the target $1 - \alpha$ accurately; see Appendix N.

Results on synthetic sinusoidal data and additional datasets (stocks, electricity, other synthetic patterns) are in Appendix O; UP-OCP consistently dominates the baselines across all benchmarks.

6 Discussion

This paper shows that coverage guarantees for online conformal prediction (OCP) can be derived from linearized regret bounds. Then it proposes UP-OCP, a parameter-free strategy for OCP using Universal Portfolio methods. Open directions include extending the framework beyond symmetric intervals toward richer prediction sets or conditional/feature-dependent validity.

Acknowledgements

Edgar's work was supported in part by the US NSF, ARO, AFOSR, ONR, the Simons Foundation and the Sloan Foundation.

[1] DtACI produces infinite-radius sets at all target levels, making its mean size infinite.

References

Anastasios Angelopoulos, Emmanuel Candes, and Ryan J Tibshirani. Conformal PID control for time series prediction. *Advances in neural information processing systems*, 36:23047–23074, 2023.

Anastasios N Angelopoulos, Michael I Jordan, and Ryan J Tibshirani. Gradient equilibrium in online learning: Theory and applications. *arXiv preprint arXiv:2501.08330*, 2025.

Felipe Areces, Christopher Mohri, Tatsunori Hashimoto, and John Duchi. Online conformal prediction via online optimization. In *Forty-second International Conference on Machine Learning*, 2025.

Osbert Bastani, Varun Gupta, Christopher Jung, Georgy Noarov, Ramya Ramalingam, and Aaron Roth. Practical adversarial multivalid conformal prediction. *Advances in neural information processing systems*, 35:29362–29373, 2022.

Aadyot Bhatnagar, Huan Wang, Caiming Xiong, and Yu Bai. Improved online conformal prediction via strongly adaptive online learning. In *International Conference on Machine Learning*, pages 2337–2363. PMLR, 2023.

Yang Cai, Constantinos Daskalakis, Haipeng Luo, Chen-Yu Wei, and Weiqiang Zheng. On tractable ϕ-equilibria in non-concave games. *Advances in Neural Information Processing Systems*, 37:140366–140404, 2024.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Nicolò Cesa-Bianchi and Francesco Orabona. Online learning algorithms. *Annual Review of Statistics and Its Application*, 8, 2021.

Thomas M Cover and Erik Ordentlich. Universal portfolios with side information. *IEEE Transactions on Information Theory*, 42(2):348–363, 2002.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672, 2021.

Isaac Gibbs and Emmanuel J Candès. Conformal inference for online prediction with arbitrary distribution shifts. *Journal of Machine Learning Research*, 25(162):1–36, 2024.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102 (477):359–378, 2007.

G. J. Gordon. Regret bounds for prediction problems. In *Proc. of the twelfth annual conference on Computational learning theory (COLT)*, pages 29–40, 1999.

Michael Harries. Splice-2 comparative evaluation: Electricity pricing. Technical report, University of New South Wales, School of Computer Science and Engineering, 1999.

Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2 (3-4):157–325, 2016.

Sunay Joshi, Shayan Kiyani, George Pappas, Edgar Dobriban, and Hamed Hassani. Conformal inference under high-dimensional covariate shifts via likelihood-ratio regularization. *arXiv preprint arXiv:2502.13030*, 2025.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-Free Predictive Inference for Regression. *Journal of the American Statistical Association*, 113(523): 1094–1111, 2018. ISSN 1537274X. doi: 10.1080/01621459.2017.1307116.

Cam Nguyen. S&P 500 stock data, 2018. URL `https://www.kaggle.com/datasets/camnugent/sandp500`.

F. Orabona and D. Pál. Coin betting and parameter-free online learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 577–585. Curran Associates, Inc., 2016.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019. Version 8.

Francesco Orabona and Kwang-Sung Jun. Tight concentrations and confidence sequences from the regret of universal portfolio. *IEEE Transactions on Information Theory*, 70(1):436–455, 2023.

Francesco Orabona and Dávid Pál. Scale-free online learning. *Theoretical Computer Science*, 716:50–69, 2018.

Aleksandr Podkopaev, Dong Xu, and Kuang-Chih Lee. Adaptive conformal inference by betting. In *International Conference on Machine Learning*, pages 40886–40907. PMLR, 2024.

Vaidehi Srinivas. Online conformal prediction with efficiency guarantees. In *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 6696–6726. SIAM, 2026.

Sean J Taylor and Benjamin Letham. Forecasting at scale. *The American Statistician*, 72(1):37–45, 2018.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. Adaptive conformal predictions for time series. In *International Conference on Machine Learning*, pages 25834–25866. PMLR, 2022.

Zhiyu Zhang, David Bombara, and Heng Yang. Discounted adaptive online learning: Towards better regularization. In *International Conference on Machine Learning*, pages 58631–58661. PMLR, 2024.

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proc. of the International Conference on Machine Learning*, pages 928–936, 2003.

Supplementary Materials

A Related Work

Adaptive Conformal Inference (ACI) (Gibbs and Candes, 2021) maintains a quantile threshold α_t over time $t = 1, 2, \ldots,$ and updates it via Online Subgradient Descent (OSD) on the pinball (aka quantile) loss. ACI achieves long-term coverage close to the target $1 - \alpha$ level. However, its performance depends heavily on the stepsize. A small stepsize results in slow adaptation and miscoverage after a large distribution shift, while a large stepsize induces high variance and instability in the prediction set widths (Gibbs and Candès, 2024; Angelopoulos et al., 2023).

To address this limitation, later work introduces alternatives. Multi-valid Conformal Prediction (MVP) (Bastani et al., 2022) selects a threshold with the best historical coverage from a discretized grid. MVP guarantees long-term and threshold-calibrated coverage at multiple levels, but lacks rapid adaptivity to abrupt changes (Bastani et al., 2022).

However, the above approaches require tuning a stepsize, which is challenging in the online setting. Because the sequences are adversarial, we cannot rely, for example, on train-test validation to tune hyperparameters. To avoid this problem, Zaffran et al. (2022) propose Aggregated ACI, which uses online expert aggregation, running multiple copies of ACI with various stepsizes and forming a weighted ensemble. Relatedly, Dynamically-Tuned ACI (DtACI) (Gibbs and Candès, 2024) re-weights experts in order to emphasize recent data; in effect it tunes ACI's stepsize online by minimizing a quantile loss. However, as noted by Angelopoulos et al. (2023), ACI-based methods can sometimes output infinite or null prediction sets, when α_t drifts below zero or above unity, respectively. Strongly Adaptive OCP (Bhatnagar et al., 2023) aggregates a number of different base algorithms to guarantee the worst-case regret over each sub-intervals, but it requires uniformly bounded predictions and the knowledge of the maximum range of the predictions, and both assumptions often fail to hold in practice. Zhang et al. (2024) and Podkopaev et al. (2024) instead avoid the use of a stepsize by using "parameter-free" online learning algorithms (Orabona and Pál, 2016), such as scale-free online gradient descent Podkopaev et al. (2024).

A different approach has been recently proposed by Srinivas (2026), where one directly addresses the optimal trade-off between coverage and size of the confidence sets in a competitive ratio framework. Notably, their analysis confirms that robust coverage in the worst-case setting fundamentally necessitates larger prediction sets. However, their algorithm, as the one by Bhatnagar et al. (2023), requires uniformly bounded predictions and the knowledge of the maximum range of the predictions.

A complementary line of work frames online conformal calibration as a feedback-control problem and proposes conformal P/PI/PID controllers (Angelopoulos et al., 2023). Unlike these controller-based schemes—which introduce gain hyperparameters and typically require controller-specific analyses—we are interested in parameter-free approaches, more suited to the online setting.

Moreover, prior work used specialized analyses to prove that the algorithms can guarantee asymptotic coverage. To date, the precise connection between online learning and OCP is unclear. In online learning, the central goal is to obtain a sublinear *regret*, i.e., the difference between the cumulative loss of the algorithm and the one of the best fixed predictor chosen in hindsight. Algorithms that satisfy this property are said to be *no regret*. However, Angelopoulos et al. (2025) also show that achieving coverage is, in general, completely distinct from achieving sublinear regret. Similarly, as we discuss later, the notion of proximal regret (Cai et al., 2024) also implies coverage, but has been established only for gradient descent. This makes it unclear when one can port methods from online learning for OCP.

B Remarks on Regret and Coverage

Remark B.1. Angelopoulos et al. (2025, Example 1) show that sublinear regret does not imply coverage. Their

example reduces to the following: consider $S_t = S > 0$ for all t, and an online learning algorithm that predicts $S + 1/\sqrt{t}$. While the regret with respect to the optimal prediction S is sub-linear, $\alpha \sum_{t=1}^{T} 1/\sqrt{t} \leq 2\alpha\sqrt{T}$, we have that $g_t = \alpha$ for all t, so the coverage error does not vanish. Our results are not in contradiction. Specifically, as already discussed, the linearized regret is stronger than the standard one. By taking $u = S - \epsilon$, where $\epsilon > 0$, as the competitor in $\text{LinRegret}_T(u)$, we see that the linearized regret is *not controlled*, growing at least as $\alpha\epsilon T$. Hence, $b_t = S + 1/\sqrt{t}$ does not control the linearized regret. This shows that our results are not in contradiction.

Remark B.2. Cai et al. (2024) derive coverage guarantees for OSD by showing that it minimizes a notion called *proximal regret* for the specific class of linear functions. However, they only prove this property for OSD, while we handle any online algorithm with a suitable linearized regret. Angelopoulos et al. (2025) discuss another notion, *no-move regret*, as a special case of proximal regret, and show a corresponding asymptotic coverage result for smooth losses. Our guarantees are instead derived for algorithms minimizing the non-smooth pinball loss.

C Comparison of UP-OCP with Previous OCP Algorithms

Here, we compare the coverage rates of UP-OCP with existing bounds. While we only compare upper bounds and not actual coverage, we show in Section 5 that the bounds are consistent with the empirical results.

First, we compare with the coverage guarantee for OSD. In OSD, one has to choose a stepsize η. The OSD bound decreases with η, so one might be tempted to set η to be large. However, in that case, OSD would predict zero on the first round and, if the scores were bounded by D, values larger than D later, until predicting a non-positive number. The cycle would then repeat. This behavior would not be informative for uncertainty quantification.

A more meaningful setting of η is the one that minimizes the worst-case regret, that is $\eta = \frac{\sqrt{DT^q}}{\max(\alpha, 1-\alpha)\sqrt{T}}$. This gives a coverage bound of $\sqrt{DT^q}\max(\alpha, 1-\alpha)/\sqrt{T} + 1/T$. Contrary to the bound we derived for the KT strategy, this rate deteriorates with q. However, things are even worse: the choice of η that depends on q and D cannot be used, because D and q are not available to the algorithm. In this situation one can only use the stepsize $\eta = c/[\max(\alpha, 1-\alpha)\sqrt{T}]$ where $c > 0$ is a hyperparameter. With this choice the coverage will converge with the worse rate of $\frac{DT^q \max(\alpha, 1-\alpha)}{c\sqrt{T}} + \frac{1}{T}$.

Overall, we can see that OSD, tuned or untuned with the oracle knowledge of q and D, has a worse dependency in T if $q > 0$. Moreover, when $\alpha \to 1$ or $\alpha \to 0$, the coverage rate of UP-OCP approaches $(\ln T)/T$ while the one of OSD cannot be faster than $1/\sqrt{T}$.

Next, let us now consider the KT strategy. UP-OCP and the KT strategy have the same dependency on T, D, and q. However, KT is designed to be min-max optimal for coin-betting games with symmetric outcomes, that is, with $\alpha = 1/2$. So, as for OSD, we see that the rate of the KT strategy does not improve when $\alpha \to 1$ or $\alpha \to 0$. A similar rate was shown for a parameter-free algorithm in Zhang et al. (2024), but only in an asymptotic sense. Moreover, the betting strategy implicit in the algorithm in Zhang et al. (2024) is provably inferior to the one of universal portfolio, because it matches only the leading term of the growth rate of best rebalanced portfolio.

D Proof of Theorem 3.1

Our main result leverage the following simple but fundamental lemma. This lemma allows us to lower bound the term $g_t b_t$ which arises in the linearized regret, without proving a bound on the iterates of the algorithm.

Lemma D.1. *Let $b_t \in \mathbb{R}$, $t \geq 1$ be generated by any online learning algorithm and let $g_t \in \partial \ell_t(b_t)$ denote a subgradient of the pinball loss $\ell^{(1-\alpha)}(b, S_t)$ at $b = b_t$, for all t. Then, we have*

$$-g_t b_t \leq (1-\alpha) S_t, \qquad \forall t \ . \tag{11}$$

Proof. We consider cases. If $b_t < 0$, then $b_t < S_t$ since $S_t \geq 0$, hence $g_t = -(1-\alpha)$ and so $-g_t b_t = (1-\alpha) b_t \leq 0 \leq (1-\alpha) S_t$. If $0 \leq b_t < S_t$, then again $g_t = -(1-\alpha)$ and thus $-g_t b_t = (1-\alpha) b_t \leq (1-\alpha) S_t$. Finally, if $b_t \geq S_t$, then (including the tie case $b_t = S_t$) we have $g_t = \alpha$, so $-g_t b_t = -\alpha b_t \leq 0 \leq (1-\alpha) S_t$. □

We can now prove our main result.

Proof of theorem 3.1. By the assumed linearized-regret bound (13), for every $u \in \mathbb{R}$ we have

$$\sum_{t=1}^{T} g_t(b_t - u) \leq F_T(u) \ .$$

Rearranging yields

$$\left(-\sum_{t=1}^{T} g_t\right) u - F_T(u) \leq -\sum_{t=1}^{T} g_t b_t \ .$$

Taking the supremum over $u \in \mathbb{R}$ gives

$$F_T^\star\left(-\sum_{t=1}^{T} g_t\right) \leq -\sum_{t=1}^{T} g_t b_t \ .$$

Finally, Lemma D.1 implies $-g_t b_t \leq (1-\alpha) S_t$ for all t, hence $-\sum_{t=1}^{T} g_t b_t \leq (1-\alpha)\sum_{t=1}^{T} S_t$. Combining the two inequalities completes the proof. □

E Proof of Coverage of OSD

We show that our analysis can recover the known coverage bound for online subgradient descent (Gibbs and Candes, 2021). From the well-known regret guarantee of online subgradient descent—OSD—see, e.g., Orabona (2019), for all $u \in \mathbb{R}$, we have the following bound on the linearized regret:

$$\text{LinRegret}_T(u) \leq \frac{u^2}{2\eta} + \frac{\eta}{2}\sum_{t=1}^{T} g_t^2 =: F_T(u), \tag{12}$$

where g_t are defined in (4). It follows that $F_T^\star(\theta) = \sup_{u\in\mathbb{R}} \ \theta u - F_T(u) = \frac{\eta\theta^2}{2} - \frac{\eta}{2}\sum_{t=1}^{T} g_t^2$, Let $D > 0$ and $q \geq 0$. Assuming that $S_t \leq Dt^q$ for all t, by Lemma D.1 and Theorem 3.1, the miscoverage can be bounded as

$$\frac{1}{T}\left|\sum_{t=1}^{T} g_t\right| \leq \frac{1}{\sqrt{T}}\sqrt{\frac{2(1-\alpha)D}{(1+q)\eta}T^q + \max(\alpha^2, (1-\alpha)^2)} \ .$$

For the case of $q = 0$, Gibbs and Candes (2021) proved the better bound of $\mathcal{O}(1/(\eta T) + 1/T)$. The reason is that Lemma D.1 is not sharp in this specific case. Indeed, we can prove the following improved bound for OSD.

Lemma E.1 (Improved Guarantee for OSD)**.** *Let $D > 0$ and $q \geq 0$, and assume that $S_t \leq Dt^q$ for all t. The OSD algorithm with fixed stepsize η and $b_1 = 0$ over a sequence of pinball losses ℓ_t guarantees, with $g_t \in \partial\ell_t(b_t)$,*

$$-\sum_{t=1}^{T} g_t b_t \leq \frac{(DT^q + \eta)^2}{2\eta} - \frac{\eta}{2}\sum_{t=1}^{T} g_t^2 \ .$$

Proof. The (unconstrained) OSD update is $b_{t+1} = b_t - \eta g_t$. Expanding gives $b_{t+1}^2 = b_t^2 - 2\eta g_t b_t + \eta^2 g_t^2$. Summing from $t = 1$ to T and using $b_1 = 0$ yields

$$\frac{b_{T+1}^2}{2\eta} = -\sum_{t=1}^{T} g_t b_t + \frac{\eta}{2}\sum_{t=1}^{T} g_t^2 \ .$$

Thus, it suffices to show $|b_{T+1}| \leq DT^q + \eta$. Similarly to Gibbs and Candes (2021), we will show a boundedness property for the iterates b_t. We prove by induction that $|b_t| \leq D(t-1)^q + \eta$ for all t. The base case $b_1 = 0$ is immediate. Assume $|b_t| \leq D(t-1)^q + \eta$ and let's prove that $|b_{t+1}| \leq Dt^q + \eta$. If $0 \leq b_t < Dt^q$, then $|b_{t+1}| = |b_t - \eta g_t| \leq |b_t| + \eta|g_t| \leq Dt^q + \eta$ since $|g_t| \leq 1$. If $b_t \geq Dt^q$, then $b_t \geq S_t$ because $S_t \leq Dt^q$, hence $g_t = \alpha$ and $Dt^q - \eta \leq b_{t+1} \leq b_t \leq D(t-1)^q + \eta$. Finally, if $b_t < 0$, then $b_t < S_t$ and $g_t = -(1-\alpha)$, so $\eta \geq b_{t+1} \geq b_t \geq -D(t-1)^q - \eta$. In all cases, $|b_{t+1}| \leq Dt^q + \eta$, completing the induction.

Substituting $b_{T+1}^2 \leq (DT^q + \eta)^2$ into the identity yields the claim. □

Using this improved guarantee instead of Lemma D.1 in the proof of Theorem 3.1, we have that

$$\frac{1}{T}\left|\sum_{t=1}^{T} g_t\right| \leq \frac{1}{T}\left(\frac{DT^q}{\eta}+1\right) .$$

F Asymptotic Coverage from Sublinear Regret

Now, we show an asymptotic coverage result, for any algorithm whose linearized regret grows sufficiently slowly, as a function of time t and at the radius $|r| = t^p$, when the growth rate of the scores S_t is not too rapid.

Theorem F.1 (No-linearized-regret implies coverage)**.** *Consider an online learning algorithm that in each round $t = 1, 2, \ldots$ produces $b_t \in \mathbb{R}$. Assume that for all rounds $T \geq 1$ there exists a function $F_T : [0, \infty) \to [0, \infty)$ such that the linearized regret from* (6) *on the pinball losses $(\ell_t)_{1\leq t\leq T}$ is bounded as*

$$\text{LinRegret}_T(r) \leq F_T(|r|), \quad \forall r \in \mathbb{R} . \tag{13}$$

Moreover, assume that the nonconformity scores have a bounded growth: for some $D > 0$, $q \geq 0$, we have $0 \leq S_t \leq Dt^q$ for all $t \geq 1$. Finally, assume that there exists $q < p < 1$ such that[2]

$$\lim_{t\to\infty} \frac{F_t(t^p)}{t} = 0 . \tag{14}$$

Then, the algorithm satisfies the long-term coverage guarantee (1)*.*

Proof. Rearranging (13) gives, for all real r

$$-\sum_{t=1}^{T} g_t b_t \geq \left(-\sum_{t=1}^{T} g_t\right) r - F_T(|r|) .$$

Hence, by Lemma D.1, we have

$$\left(-\sum_{t=1}^{T} g_t\right) r - F_T(|r|) \leq (1-\alpha)\sum_{t=1}^{T} S_t \leq \frac{(1-\alpha)D(T+1)^{q+1}}{q+1} . \tag{15}$$

Now, set $r = r_T = -\text{sign}\left(\sum_{t=1}^{T} g_t\right) \cdot T^p$, where we define $\text{sign}(0) = 0$. Invoking (14) yields

$$\frac{(1-\alpha)D(T+1)^{q+1}}{q+1} \geq T^p \left|\sum_{t=1}^{T} g_t\right| - F_T\left(T^p\right), \tag{16}$$

which implies that

$$\frac{1}{T}\left|\sum_{t=1}^{T} g_t\right| \leq \frac{F_T\left(T^p\right)}{T^{p+1}} + \frac{(1-\alpha)D(T+1)^{q+1}}{(q+1)T^{p+1}} .$$

Now, taking the lim sup of the first term on the r.h.s., we have

$$\limsup_{T\to\infty} \frac{F_T\left(T^p\right)}{T^{p+1}} = \limsup_{T\to\infty} \frac{F_T\left(T^p\right)}{T} T^{-p} = 0 .$$

For the second term, given that $q < p$, we have that

$$\limsup_{T\to\infty} \frac{(1-\alpha)D(T+1)^{q+1}}{(q+1)T^{p+1}} = 0 .$$

We conclude by the equivalence (5). □

This result shows that for conformal prediction, the optimal Universal Portfolio strategy is computationally efficient, requiring only $\mathcal{O}(1)$ time per step. Alternatively, using a uniform prior yields the Laplace rule

$$\frac{\sum_{i=1}^{t-1} \mathbf{1}\{g_i = -(1-\alpha)\} + 1}{t+1},$$

which is known to have slightly higher regret (Orabona, 2019, Theorem 13.1).

[2] These two conditions can be summarized into $\max\{\text{LinRegret}_t(t^p), \text{LinRegret}_t(-t^p)\} = o(t)$.

G The Average Size in Stochastic Settings

While the previous sections established coverage guarantees under adversarial settings, characterizing the *efficiency*, i.e., the size of the prediction sets relative to the optimum, seems to require additional assumptions on the data generating process. In a fully adversarial setting, a globally optimal interval width is ill-defined, as there is no guarantee that the Y_t share any common behaviour over the course of the T rounds.

Therefore, we consider a stochastic setting where the nonconformity scores S_t are sampled i.i.d. from a fixed distribution $\mathcal{D}$. In this setting, the optimal fixed prediction interval of the form $[0, b^\star]$ is achieved when $b^\star$ is any $(1-\alpha)$-th quantile of the distribution. Our goal is to show that regret-minimizing algorithms do not merely satisfy coverage constraints, but also converge to this optimal radius $b^\star$ at a rate characterized by their regret.

We define the expected loss $L(b)$ with respect to $\mathcal{D}$ as $L(b) := \mathbb{E}_{S\sim\mathcal{D}}\left[\ell^{(1-\alpha)}(b, S)\right]$. We denote the optimal radius as $b^\star := \operatorname{argmin}_{b\geq 0} L(b)$; our conditions below will ensure that this is uniquely defined. To rigorously connect the regret R_T to the convergence of the distance $|b_t - b^\star|$, we require mild regularity conditions on $\mathcal{D}$.

Assumption G.1 (Regularity of $\mathcal{D}$)**.** The nonconformity scores $S_1, S_2, \ldots$ are i.i.d. draws from a distribution $\mathcal{D}$ supported on a compact interval $\mathcal{B} \subseteq [0, \infty)$, such that:

1. $\mathcal{D}$ admits a Probability Density Function (PDF) ϕ and a Cumulative Distribution Function (CDF) $\tilde{\Phi}$;
2. The density ϕ is uniformly bounded from below, so there is a constant $\kappa > 0$ such that $\phi(b) \geq \kappa$, $\forall b \in \mathcal{B}$.

In the above notation, $b^\star := \tilde{\Phi}^{-1}(1-\alpha)$.

Remark G.2. Clearly, the continuity of ϕ is not necessary for the existence of the derivatives of L. The expected loss $L(b) = \mathbb{E}[\ell^{(1-\alpha)}(b, S)]$ is a convolution of the continuous pinball loss with the measure $\mathcal{D}$. This acts as a smoothing operator: as long as $\mathcal{D}$ has no point masses, $L(b)$ is continuously differentiable (see, e.g., Joshi et al., 2025). Assumption G.1.2 is imposed primarily to ensure *strong convexity* (positive curvature), which allows us to convert the regret bound into a variance bound.

Under these conditions, we prove that a sublinear regret implies the convergence of the average radius $\bar{b}_T = \frac{1}{T}\sum_{t=1}^{T} b_t$ to the optimal oracle radius $b^\star$.

Theorem G.3 (Width Convergence via Regret)**.** *Let an online algorithm generate radii $b_1, b_2, \ldots, b_T$ with $b_t \in \mathcal{B}$ for all t such that the expected regret is bounded by R_T:*

$$\mathbb{E}\left[\sum_{t=1}^{T} \ell^{(1-\alpha)}(b_t, S_t) - \sum_{t=1}^{T} \ell^{(1-\alpha)}(b^\star, S_t)\right] \leq R_T \ . \tag{17}$$

Under Assumption G.1, the squared distance between the average radius $\bar{b}_T = \frac{1}{T}\sum_{t=1}^{T} b_t$ and the true $(1-\alpha)$-quantile is bounded by $\mathbb{E}\left[\left(\bar{b}_T - b^\star\right)^2\right] \leq \frac{2}{\kappa} \cdot \frac{R_T}{T}$.

Proof. The proof proceeds in two parts: a first- and second-order analysis of the expected loss, and applying an online-to-batch conversion.

First and second-order analysis. This part is a review of well-known results, which were used either explicitly or implicitly to various degrees in a number of prior works, see e.g., Gibbs and Candes (2021); Joshi et al. (2025), etc; which we include here only for the sake of being self-contained. Recall the $(1-\alpha)$-pinball loss: $\ell^{(1-\alpha)}(b, S) = \max\{(1-\alpha)(S-b), \alpha(b-S)\}$. The expected loss is given by

$$L(b) = \int_0^b \alpha(b-s)\phi(s)\,\mathrm{d}s + \int_b^\infty (1-\alpha)(s-b)\phi(s)\,\mathrm{d}s \ .$$

Differentiating with respect to b yields

$$L'(b) = \alpha\tilde{\Phi}(b) - (1-\alpha)(1-\tilde{\Phi}(b)) = \tilde{\Phi}(b) - (1-\alpha) \ .$$

Setting $L'(b) = 0$, we confirm that the minimizer is unique and satisfies $\tilde{\Phi}(b^\star) = 1-\alpha$. Thus, $b^\star$ is exactly the $(1-\alpha)$-quantile of $\mathcal{D}$. Differentiating $L'(b)$ again, we obtain the Hessian of the expected loss: $L''(b) =$

$\frac{d}{db}(\tilde{\Phi}(b) - 1 + \alpha) = \phi(b)$. By Assumption G.1, we have $L''(b) = \phi(b) \geq \kappa$ for all $b \in \mathcal{B}$, hence L is κ-strongly convex on $\mathcal{B}$. In particular,

$$L(b) - L(b^\star) \geq \frac{\kappa}{2}(b - b^\star)^2, \quad \forall b \in \mathcal{B} \ . \tag{18}$$

Online-to-batch conversion. Since the loss function $L(b)$ is convex, Jensen's inequality implies $L(\bar{b}_T) \leq \frac{1}{T}\sum_{t=1}^{T} L(b_t)$. Using standard online-to-batch conversion results (see, e.g., Orabona, 2019, Theorem 3.1), the average regret upper bounds the excess risk:

$$\mathbb{E}[L(\bar{b}_T)] - L(b^\star) \leq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} L(b_t) - L(b^\star)\right] \leq \frac{R_T}{T} \ .$$

Combining this with the strong convexity bound in (18), we have

$$\frac{\kappa}{2}\mathbb{E}\left[(\bar{b}_T - b^\star)^2\right] \leq \mathbb{E}[L(\bar{b}_T)] - L(b^\star) \leq \frac{R_T}{T} \ .$$

Rearranging the terms yields the claim. □

Theorem G.3 implies that *any* algorithm minimizing pinball loss regret *automatically* converges to the statistically efficient oracle width $b^\star$. These results are different in nature from those of Srinivas (2026). Srinivas (2026) frames the problem as direct length optimization, aiming to compete with the best fixed interval length in hindsight. In contrast, our framework specifically leverages the pinball loss as a proper scoring rule Gneiting and Raftery (2007).

Recent work by Areces et al. (2025) also investigates the efficiency of online conformal prediction in stochastic settings. Their main result (Theorem 6.1) establishes the convergence of the *last iterate* to the optimal parameters. However, obtaining this guarantee requires a specific decaying schedule for the learning rate ($\eta_t \propto t^{-c}$), which creates an explicit trade-off: faster decay improves efficiency but renders the adversarial coverage bounds vacuous (Areces et al., 2025, Section 6). In contrast, our width convergence result (Theorem G.3) is more general, showing that efficiency is an automatic consequence of regret minimization for *any* algorithm, regardless of its specific update rule. Furthermore, because UP-OCP is parameter-free, it naturally achieves this efficiency through low regret without requiring manual step-size tuning or sacrificing robust coverage guarantees under adversarial distribution shifts.

H Proof of Theorem 4.3

This appendix details the reduction used in Section 4. The main point is that, once we express the pinball subgradients as an asymmetric coin sequence, our two-stock conformal market in Definition 4.1 is exactly a two-asset encoding of an asymmetric coin-betting game. Standard coin-betting duality then translates wealth guarantees into linearized-regret guarantees, which in turn control the pinball-loss regret.

H.1 From pinball subgradients to an asymmetric coin

Recall that for the pinball loss $\ell_t(b) = \ell^{(1-\alpha)}(b, S_t)$ we use the subgradient $g_t = \mathbf{1}\{b_t \geq S_t\} - (1-\alpha)$, as in (4). Hence, with our tie-breaking convention at $b_t = S_t$, we have $g_t \in \{-(1-\alpha), \alpha\}$.

It is convenient to flip signs and work with the bounded outcome of a coin:

$$c_t := -g_t \in \{-\alpha, 1-\alpha\} \ . \tag{19}$$

Under our tie-breaking, c_t takes only the two values: $c_t = 1 - \alpha$ if $b_t < S_t$ (miscoverage) and $c_t = -\alpha$ if $b_t \geq S_t$ (coverage).

Consider now the following asymmetric coin-betting game. At each round t, a bettor chooses a *signed betting fraction* β_t and then the outcome c_t is revealed. We define the wealth process as

$$W_t := W_{t-1}(1 + \beta_t c_t), \qquad W_0 := 1 \ . \tag{20}$$

To ensure *no bankruptcy* for all outcomes $c_t \in [-\alpha, 1-\alpha]$, the betting fraction must satisfy

$$\beta_t \in \left[-\frac{1}{1-\alpha}, \frac{1}{\alpha}\right] . \tag{21}$$

Indeed, the condition $1 + \beta_t c_t \geq 0$ for all $c_t \in \{-\alpha, 1-\alpha\}$ is equivalent to simultaneously requiring $1 - \alpha\beta_t \geq 0$ and $1 + (1-\alpha)\beta_t \geq 0$, which is exactly (21).

We also define the *bet amount*

$$b_t := \beta_t W_{t-1} . \tag{22}$$

Plugging (22) into (20) yields the additive form

$$W_t = W_{t-1} + c_t b_t = W_{t-1} - g_t b_t . \tag{23}$$

H.2 The conformal market is a two-stock encoding of coin betting

We now show that the conformal market in Definition 4.1 is precisely a two-asset representation of the asymmetric coin game above.

Using $c_t = -g_t$, Definition 4.1 can be rewritten as

$$w_{t,1} = \frac{c_t + \alpha}{\alpha}, \qquad w_{t,2} = \frac{(1-\alpha) - c_t}{1-\alpha} . \tag{24}$$

These returns are nonnegative for every $c_t \in [-\alpha, 1-\alpha]$. Moreover, when $c_t = 1-\alpha$ (miscoverage), we have $w_{t,1} = 1/\alpha$ and $w_{t,2} = 0$, while when $c_t = -\alpha$ (coverage), we have $w_{t,1} = 0$ and $w_{t,2} = 1/(1-\alpha)$.

Given a portfolio weight $\lambda_t \in [0,1]$ (fraction of capital invested in Stock 1), the wealth update in Definition 4.2 is $W_t = W_{t-1}(\lambda_t w_{t,1} + (1-\lambda_t) w_{t,2})$. The next lemma shows that this is exactly (20) under an affine reparameterization of λ_t.

Lemma H.1 (Portfolio-to-coin equivalence)**.** *Fix $\alpha \in (0,1)$ and let $c_t \in [-\alpha, 1-\alpha]$. Define the two-stock returns by* (24)*. For any $\lambda_t \in [0,1]$, define*

$$\beta_t := -\frac{1}{1-\alpha} + \frac{\lambda_t}{\alpha(1-\alpha)} = \frac{\lambda_t - \alpha}{\alpha(1-\alpha)} . \tag{25}$$

Then,

$$\lambda_t w_{t,1} + (1-\lambda_t) w_{t,2} = 1 + \beta_t c_t . \tag{26}$$

Consequently, the wealth recursion of Definition 4.2 is identical to the coin-betting recursion (20)*, and the bet amount $b_t = \beta_t W_{t-1}$ equals the mapping* (10)*.*

Proof. Using (24),

$$\begin{aligned}
\lambda_t w_{t,1} + (1-\lambda_t) w_{t,2} &= \lambda_t \left(\frac{c_t + \alpha}{\alpha}\right) + (1-\lambda_t)\left(\frac{(1-\alpha) - c_t}{1-\alpha}\right) \\
&= \lambda_t \left(1 + \frac{c_t}{\alpha}\right) + (1-\lambda_t)\left(1 - \frac{c_t}{1-\alpha}\right) \\
&= 1 + c_t \left(\frac{\lambda_t}{\alpha} - \frac{1-\lambda_t}{1-\alpha}\right) .
\end{aligned}$$

The coefficient of c_t simplifies as $\frac{\lambda_t}{\alpha} - \frac{1-\lambda_t}{1-\alpha} = -\frac{1}{1-\alpha} + \frac{\lambda_t}{\alpha(1-\alpha)} = \beta_t$, which proves (26). Substituting into the portfolio recursion gives $W_t = W_{t-1}(1 + \beta_t c_t)$, and $b_t = \beta_t W_{t-1}$ is exactly (10). □

Two immediate consequences are worth recording. First, since $\lambda_t \in [0,1]$, the induced β_t in (25) always lies in the safe interval (21). Second, the one-dimensional action b_t in (10) is simply the coin-betting bet amount associated with the portfolio choice λ_t.

H.3 From wealth lower bounds to linearized-regret bounds

We now connect wealth to the linearized regret on the pinball loss. Recall that linearized regret is $\mathrm{LinRegret}_T(u) = \sum_{t=1}^T g_t(b_t - u)$. Using $c_t = -g_t$ from (19), this can be rewritten as

$$\mathrm{LinRegret}_T(u) = \sum_{t=1}^T c_t(u - b_t) \ . \tag{27}$$

Moreover, by telescoping (23), we have

$$W_T = 1 + \sum_{t=1}^T c_t b_t \ . \tag{28}$$

The standard coin-betting duality is that a *lower bound* on the achievable wealth as a function of the cumulative outcome sum implies an *upper bound* on linearized regret via Fenchel conjugacy (see, e.g., Orabona, 2019).

Lemma H.2 (Wealth lower bound ⇒ linearized regret bound)**.** *Let $\Psi_T : \mathbb{R} \to (-\infty, +\infty]$ be a function. Assume that an algorithm produces $(b_t)_{t=1}^T$ and wealth W_T satisfying*

$$W_T - 1 \geq \Psi_T\left(\sum_{t=1}^T c_t\right) \tag{29}$$

for every sequence $(c_t)_{t=1}^T \subseteq [-\alpha, 1-\alpha]$. Then, for every comparator $u \in \mathbb{R}$, its linearized regret satisfies

$$\mathrm{LinRegret}_T(u) \leq \Psi_T^\star(u), \tag{30}$$

where $\Psi_T^\star$ is the Fenchel conjugate of Ψ_T.

Proof. Let $\theta_T := \sum_{t=1}^T c_t$. By Fenchel-Young duality, for every u we have $u\theta_T \leq \Psi_T(\theta_T) + \Psi_T^\star(u)$. By the assumption (29) and the identity (28), we have $\sum_{t=1}^T c_t b_t = W_T - 1 \geq \Psi_T(\theta_T)$. Therefore,

$$\sum_{t=1}^T c_t(u - b_t) = u\theta_T - \sum_{t=1}^T c_t b_t \leq u\theta_T - \Psi_T(\theta_T) \leq \Psi_T^\star(u),$$

where in the last equality we used Fenchel-Young inequality. This is exactly (30) using (27). □

Lemma H.2 is the precise mathematical sense in which "maximizing wealth" (in the coin-betting/portfolio game) corresponds to "minimizing linearized regret" (in online convex optimization). The only remaining ingredient is to identify an explicit lower bound Ψ_T for $W_T - 1$ and calculate (an upper bound to) its Fenchel conjugate.

H.4 Instantiating the potential via the best constant rebalanced portfolio

We now connect the portfolio regret guarantee to a wealth lower bound of the form (29).

Let $W_T^\star$ denote the wealth of the best constant rebalanced portfolio in the conformal market, i.e., $W_T^\star = \max_{\lambda \in [0,1]} \prod_{t=1}^T (\lambda w_{t,1} + (1-\lambda) w_{t,2})$. By Lemma H.1, this is equivalently the best constant betting fraction in the asymmetric coin game: $W_T^\star = \max_{\beta \in [-1/(1-\alpha), 1/\alpha]} \prod_{t=1}^T (1 + \beta c_t)$.

When we use the extreme subgradient convention (so $c_t \in \{-\alpha, 1-\alpha\}$), $W_T^\star$ depends on the data only through the number of miscoverages. Let $M_T := \sum_{t=1}^T \mathbf{1}\{c_t = 1-\alpha\}$ and $C_T := T - M_T$. Then, for a fixed $\lambda \in [0,1]$, we have

$$W_T(\lambda) = \left(\frac{\lambda}{\alpha}\right)^{M_T} \left(\frac{1-\lambda}{1-\alpha}\right)^{C_T} \ .$$

Maximizing over λ yields $\lambda^\star = M_T/T$, and hence

$$W_T^\star = \left(\frac{M_T}{\alpha T}\right)^{M_T} \left(\frac{C_T}{(1-\alpha)T}\right)^{C_T} = \exp\left(T \cdot \mathrm{KL}\left(\frac{M_T}{T} \middle\| \alpha\right)\right), \tag{31}$$

where $\mathrm{KL}(p\|q) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}$ is the Bernoulli KL divergence.

Moreover, M_T can be expressed directly in terms of $\theta_T = \sum_{t=1}^T c_t$: since c_t equals $1-\alpha$ on miscoverage and $-\alpha$ on coverage, we have $\theta_T = (1-\alpha)M_T - \alpha C_T = M_T - \alpha T$, hence $M_T = \alpha T + \theta_T$. Substituting into (31) gives an explicit function of θ_T:

$$W_T^\star = \exp\left(T \cdot \mathrm{KL}\left(\alpha + \frac{\theta_T}{T}\middle\|\alpha\right)\right) . \tag{32}$$

From the assumption that the portfolio algorithm guarantees $\log W_T \geq \log W_T^\star - \mathcal{R}_T$, exponentiating yields

$$W_T \geq \exp\left(-\mathcal{R}_T\right) W_T^\star = \exp\left(-\mathcal{R}_T\right) \exp\left(T \cdot \mathrm{KL}\left(\alpha + \frac{\theta_T}{T}\middle\|\alpha\right)\right) := \Psi_T(\theta_T) + 1 . \tag{33}$$

Using Lemma H.2, we need to calculate the Fenchel conjugate of Ψ_T. Unfortunately, it does not have a closed form expression. Hence, we use Lemma I.1, to obtain an easier lower bound:

$$\exp\left(T \cdot \mathrm{KL}\left(\alpha + \frac{\theta_T}{T}\middle\|\alpha\right)\right) \geq \exp\left(T \cdot \frac{\theta_T^2/T^2}{2\alpha(1-\alpha) + 2/3|\theta_T|/T}\right) \geq \min\left\{\exp\left(\frac{\theta_T^2}{4\alpha(1-\alpha)}\right), \exp\left(\frac{3}{4}|\theta_T|\right)\right\} . \tag{34}$$

Now, observe that if $f(x) = \min\{h_1(x), h_2(x)\}$, then we have

$$\begin{aligned} f^\star(y) &= \sup_y xy - f(x) = \sup_y xy - \min\{h_1(x), h_2(x)\} = \sup_y \max\{xy - h_1(x), xy - h_2(x)\} \\ &\leq \max\{\sup_y xy - h_1(x), \sup_y xy - h_2(x)\} = \max\{h_1^\star(y), h_2^\star(y)\} . \end{aligned}$$

Hence, it suffices to find the Fenchel conjugates (or upper bounds) of the two functions in the min in the right-hand side of (34). This can be done immediately by Orabona (2019, Example 6.18, Lemma 6.24, and Theorems C.3 and C.4):

$$\Psi_T^\star(u) \leq \max\left\{|u|\sqrt{2T\alpha(1-\alpha)\ln(2T\alpha(1-\alpha)u^2\exp\left(\mathcal{R}_T\right)+1)}, \frac{4}{3}|u|\left(\ln\frac{4|u|\exp(\mathcal{R}_T)}{3} - 1\right)\right\} .$$

Using the regret of universal portfolio of $\frac{1}{2}\ln(\pi(T+1))$ and overapproximating completes the proof.

I Regret Guarantee of UP-OCP vs KT

The proof the regret guarantee of Podkopaev et al. (2024) goes exactly through the same steps of the one of UP-OCP. Moreover, the KT approach can also be written as a universal portfolio algorithm, with the same prior. The only difference is the transformation of the subgradients into 2 stocks, that will change the wealth of the best constant rebalanced portfolio, $W_T^\star$.

The transformation for KT is the following one:

$$w_{t,1} = 1 - g_t, \quad w_{t,2} = 1 + g_t .$$

Given that $g_t \in \{\alpha - 1, \alpha\} \subset \{-1, 1\}$, we have that $w_{t,1} \geq 0$ and $w_{t,2} \geq 0$. In the coin-betting view, this corresponds to the wealth process

$$W_t = W_{t-1}(1 - g_t\beta_t),$$

where we constrain $\beta_t \in [-1, 1]$. This means that $W_t^\star = \max_{\beta\in[-1,1]} \prod_{t=1}^T (1 - g_t\beta)$. Contrast this with the one we derived for UP-OCP:

$$\max_{\beta\in[-1/(1-\alpha),1/\alpha]} \prod_{t=1}^T (1 - \beta g_t) .$$

Given that $[-1, 1] \subset [-1/(1-\alpha), 1/\alpha]$, the wealth of the best constant rebalanced portfolio in the UP-OCP reduction is always at least the same of the KT one, but it is potentially much larger.

I.1 Missing proofs in section 4

Lemma I.1. *Let $p, q \in [0,1]$, then*

$$\mathrm{KL}(p\|q) = p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q} \geq \frac{(p-q)^2}{2q(1-q) + \frac{2}{3}|p-q|} \ .$$

Proof of Lemma I.1. Define $h(x) = (1+x)\ln(1+x) - x$ for $x > -1$ and extend it by continuity in $x = -1$ with $h(-1) := 1$. Also, define $\Delta = p - q \in [-1, 1]$.

We have that

$$p \ln \frac{p}{q} = (q+\Delta) \ln \left(1 + \frac{\Delta}{q}\right) = q \left(h\left(\frac{\Delta}{q}\right) + \frac{\Delta}{q}\right) = q\, h\left(\frac{\Delta}{q}\right) + \Delta \ .$$

Similarly, we have

$$(1-p) \ln \frac{1-p}{1-q} = (1-q-\Delta) \ln \left(1 - \frac{\Delta}{1-q}\right) = (1-q)\left(1 - \frac{\Delta}{1-q}\right) \ln \left(1 - \frac{\Delta}{1-q}\right) = (1-q)\, h\left(-\frac{\Delta}{1-q}\right) - \Delta \ .$$

Hence, overall we have

$$\mathrm{KL}(p\|q) = q\, h\left(\frac{\Delta}{q}\right) + (1-q)\, h\left(-\frac{\Delta}{1-q}\right) \ .$$

Observe that $\frac{\Delta}{q} \geq -1$ and $-\frac{\Delta}{1-q} \geq -1$. Hence, we use the elementary inequality $h(x) \geq \frac{x^2}{2+\frac{2}{3}|x|}$ for $x \geq -1$, to have

$$\mathrm{KL}(p\|q) \geq \Delta^2 \left(\frac{1}{2q + \frac{2}{3}|\Delta|} + \frac{1}{2(1-q) + \frac{2}{3}|\Delta|}\right) = \Delta^2 \frac{2 + \frac{4}{3}|\Delta|}{4q(1-q) + \frac{4}{3}|\Delta| + \frac{4}{9}\Delta^2} \geq \frac{\Delta^2}{2q(1-q) + \frac{2}{3}|\Delta|} \ .$$

Using the value of Δ finishes the proof. □

Proof of theorem 4.4. Let $k := |\{t \in [T] : g_t < 0\}|$ and $\hat{p} := k/T$. Since $g_t = \alpha$ on coverage rounds and $g_t = -(1-\alpha)$ on miscoverage rounds,

$$\sum_{t=1}^{T} g_t = \alpha(T-k) - (1-\alpha)k = \alpha T - k, \qquad \frac{1}{T}\sum_{t=1}^{T} g_t = \alpha - \hat{p} \ .$$

Best-constant wealth equals an exact KL term. For a constant betting strategy $\lambda \in [0,1]$, the wealth is

$$W_T(\lambda) = \left(\frac{\lambda}{\alpha}\right)^k \left(\frac{1-\lambda}{1-\alpha}\right)^{T-k} \ .$$

The maximizer is $\lambda^\star = \hat{p}$, and substituting yields

$$\begin{aligned} \ln W_T^\star &= k \ln \frac{\hat{p}}{\alpha} + (T-k) \ln \frac{1-\hat{p}}{1-\alpha} \\ &= T\left(\hat{p} \ln \frac{\hat{p}}{\alpha} + (1-\hat{p}) \ln \frac{1-\hat{p}}{1-\alpha}\right) = T \cdot \mathrm{KL}(\hat{p}\|\alpha) \ . \end{aligned} \tag{35}$$

For two assets with Jeffreys prior, the Universal Portfolio wealth is the Beta$(1/2, 1/2)$ mixture, whose regret is known (Cover and Ordentlich, 2002):

$$\ln W_T \geq \ln W_T^\star - \frac{1}{2} \ln(\pi(T+1)) \ . \tag{36}$$

Moreover, by Lemma D.1 and the assumption that $S_t \leq Dt^q$, we have that

$$W_T = 1 - \sum_{t=1}^{T} b_t g_t \leq 1 + \frac{(1-\alpha)D}{q+1}(T+1)^{q+1} \ .$$

Combining (35) and (36) gives

$$\mathrm{KL}(\hat{p}\|\alpha) \leq \varepsilon_T := \frac{1}{T}\left[\ln\left(1 + \frac{(1-\alpha)D(T+1)^{q+1}}{q+1}\right) + \frac{1}{2}\ln(\pi(T+1))\right] \ . \tag{37}$$

Explicit inversion from KL to miscoverage deviation. Let $\delta := |\hat{p} - \alpha|$. From Lemma I.1, we have

$$\mathrm{KL}(a\|b) \geq \frac{(a-b)^2}{2b(1-b) + \frac{2}{3}|a-b|} \ . \tag{38}$$

Applying (38) with $a = \hat{p}$ and $b = \alpha$ and using (37) yields

$$\varepsilon_T \geq \frac{\delta^2}{2\alpha(1-\alpha) + \frac{2}{3}\delta},$$

equivalently

$$\delta^2 - \frac{2\varepsilon_T}{3}\delta - 2\alpha(1-\alpha)\varepsilon_T \leq 0 \ .$$

Solving this quadratic inequality for the nonnegative root gives

$$\delta \leq \frac{\varepsilon_T}{3} + \sqrt{2\alpha(1-\alpha)\varepsilon_T + \frac{\varepsilon_T^2}{9}} \leq \varepsilon_T + \sqrt{2\alpha(1-\alpha)\varepsilon_T} \ .$$

Since $\delta = |\hat{p} - \alpha| = \left|\frac{1}{T}\sum_{t=1}^{T} g_t\right|$, this proves the stated bound. □

I.2 Radius Clipping

We slightly abuse notation here by letting $\tilde{b}_t$ be the *raw* output of the wealth mapping (10). In Algorithm 1, the prediction radius is truncated to be non-negative: $b_t \leftarrow \max(0, \tilde{b}_t)$. Here we show that this operation preserves the validity of the regret guarantees.

First, we examine the consistency of the gradients used for the wealth update. The algorithm updates the wealth based on whether coverage was attained:

- Case 1: If $\tilde{b}_t \geq 0$, then trivially the clipping step is not active and hence $b_t = \tilde{b}_t$. The subgradient g_t is computed at the same point as in the unclipped case.
- Case 2: If $\tilde{b}_t < 0$, then $\tilde{b}_t < S_t$ (since $S_t \geq 0$) and hence the algorithm receives the subgradient $g_t = -(1-\alpha)$. In terms of the clipped value $b_t = 0$, we also have $b_t \leq S_t$, yielding the same gradient $g_t = -(1-\alpha)$.

We conclude that the clipping does not alter the subgradient sequence seen by the algorithm in the standard case.

Now, we compare the linearized regret terms. Let $\mathrm{LinRegret}_t(u) = g_t(b_t - u)$ and $\widetilde{\mathrm{LinRegret}}_t(u) = \tilde{g}_t(\tilde{b}_t - u)$. The difference is:

$$\mathrm{LinRegret}_t(u) - \widetilde{\mathrm{LinRegret}}_t(u) = g_t(b_t - u) - g_t(\tilde{b}_t - u) = g_t(b_t - \tilde{b}_t) \ .$$

Again, we do a case analysis:

- If $\tilde{b}_t \geq 0$, then $b_t = \tilde{b}_t$, so the difference is 0.
- If $\tilde{b}_t < 0$, then $b_t = 0$ and $g_t = -(1-\alpha)$. The difference is:

$$-(1-\alpha)(0 - \tilde{b}_t) = (1-\alpha)\tilde{b}_t < 0 \ .$$

In all cases, $g_t(b_t - u) \leq \tilde{g}_t(\tilde{b}_t - u)$. Summing over t confirms that the linearized regret of the truncated algorithm satisfies the same bound as the original wealth process.

J Closed-Form UP Update

Theorem J.1 (Closed-Form Update (Jeffreys prior))**.** *For the universal portfolio update* (9) *with the Jeffreys prior on* $\Delta = [0,1]$ *(equivalently a* Beta$(1/2, 1/2)$ *prior in the two-asset case), the weights admit the closed-form update*

$$\lambda_t = \frac{1}{t}\left(\sum_{i=1}^{t-1} \mathbf{1}\left\{g_i = -(1-\alpha)\right\} + \frac{1}{2}\right) . \tag{39}$$

Proof. In the conformal market of Definition 4.1, at round t the two synthetic stocks have gross returns

$$(w_{t,1}, w_{t,2}) = \left(1 - \frac{g_t}{\alpha},\, 1 + \frac{g_t}{1-\alpha}\right) .$$

A constant-rebalanced portfolio that invests fraction $\lambda \in [0,1]$ in stock 1 and $1-\lambda$ in stock 2 achieves one-step gross return $\lambda\, w_{t,1} + (1-\lambda)\, w_{t,2}$. Hence, the universal portfolio weight is

$$\lambda_{T+1} = \frac{1}{K}\int_0^1 \lambda \prod_{t=1}^{T}\left[\lambda\left(1 - \frac{g_t}{\alpha}\right) + (1-\lambda)\left(1 + \frac{g_t}{1-\alpha}\right)\right]\mu(\lambda)\,\mathrm{d}\lambda,$$

where

$$K = \int_0^1 \prod_{t=1}^{T}\left[\lambda\left(1 - \frac{g_t}{\alpha}\right) + (1-\lambda)\left(1 + \frac{g_t}{1-\alpha}\right)\right]\mu(\lambda)\,\mathrm{d}\lambda .$$

Now use that $g_t \in \{-(1-\alpha), \alpha\}$. Let a be the number of rounds with $g_t = -(1-\alpha)$ (miscoverage), so $T-a$ is the number of rounds with $g_t = \alpha$. For a miscoverage round $g_t = -(1-\alpha)$, we have $1 - g_t/\alpha = 1 + (1-\alpha)/\alpha = 1/\alpha$ and $1 + g_t/(1-\alpha) = 1 - 1 = 0$, hence

$$\lambda\left(1 - \frac{g_t}{\alpha}\right) + (1-\lambda)\left(1 + \frac{g_t}{1-\alpha}\right) = \frac{\lambda}{\alpha} .$$

For a coverage round $g_t = \alpha$, we have $1 - g_t/\alpha = 0$ and $1 + g_t/(1-\alpha) = 1 + \alpha/(1-\alpha) = 1/(1-\alpha)$, hence

$$\lambda\left(1 - \frac{g_t}{\alpha}\right) + (1-\lambda)\left(1 + \frac{g_t}{1-\alpha}\right) = \frac{1-\lambda}{1-\alpha} .$$

Therefore,

$$\prod_{t=1}^{T}\left[\lambda\left(1 - \frac{g_t}{\alpha}\right) + (1-\lambda)\left(1 + \frac{g_t}{1-\alpha}\right)\right] = \left(\frac{\lambda}{\alpha}\right)^a \left(\frac{1-\lambda}{1-\alpha}\right)^{T-a} .$$

Plugging this into numerator and denominator gives

$$\begin{aligned}\lambda_{T+1} &= \frac{\int_0^1 \lambda \left(\frac{\lambda}{\alpha}\right)^a \left(\frac{1-\lambda}{1-\alpha}\right)^{T-a} \mu(\lambda)\,\mathrm{d}\lambda}{\int_0^1 \left(\frac{\lambda}{\alpha}\right)^a \left(\frac{1-\lambda}{1-\alpha}\right)^{T-a} \mu(\lambda)\,\mathrm{d}\lambda} \\ &= \frac{\int_0^1 \lambda^{a+1}(1-\lambda)^{T-a}\mu(\lambda)\,\mathrm{d}\lambda}{\int_0^1 \lambda^{a}(1-\lambda)^{T-a}\mu(\lambda)\,\mathrm{d}\lambda},\end{aligned}$$

since the constant factor $\alpha^{-a}(1-\alpha)^{a-T}$ cancels.

Now, let's specialize it to the Dirichlet$(1/2, 1/2)$ prior, whose marginal density is $\mu(\lambda) = \frac{\Gamma(1)}{\Gamma(1/2)^2}\lambda^{-1/2}(1-\lambda)^{-1/2}$ on $(0,1)$. Define, for $p, q > -1/2$,

$$I(p,q) = \int_0^1 \lambda^p (1-\lambda)^q \mu(\lambda)\,\mathrm{d}\lambda .$$

Then,

$$I(p,q) = \frac{\Gamma(1)}{\Gamma(1/2)^2} \int_0^1 \lambda^{p-1/2}(1-\lambda)^{q-1/2}\,\mathrm{d}\lambda = \frac{\Gamma(1)}{\Gamma(1/2)^2} \cdot \frac{\Gamma(p+1/2)\Gamma(q+1/2)}{\Gamma(p+q+1)} \ .$$

Using $\Gamma(u+1) = u\Gamma(u)$,

$$\frac{I(p+1,q)}{I(p,q)} = \frac{\Gamma(p+3/2)}{\Gamma(p+1/2)} \cdot \frac{\Gamma(p+q+1)}{\Gamma(p+q+2)} = \frac{p+1/2}{p+q+1} \ .$$

Finally, taking $p = a$ and $q = T - a$ yields

$$\lambda_{T+1} = \frac{I(a+1, T-a)}{I(a, T-a)} = \frac{a+1/2}{T+1} \ .$$

For a uniform distribution as the prior, a similar derivation yields $\lambda_{T+1} = \frac{a+1}{T+2}$. □

K Baseline Update Rules

K.1 Krichevsky-Trofimov

The Krichevsky-Trofimov (KT) bettor is a parameter-free approach for adaptive conformal inference that addresses the sensitivity of traditional methods to learning rate tuning (Podkopaev et al., 2024). By framing the selection of the conformal radius b_t as a coin-betting game, the algorithm avoids the need for a manually specified learning rate. In this framework, the algorithm starts with an initial wealth $W_0 = 1$ and places bets on the outcome of coins $c_t \in [-1, 1]$, which are defined as the negative subgradients of the pinball loss: $c_t = -g_t$. The radius at each step t is determined by the betting fraction β_t and the current wealth:

$$b_t = \beta_t W_{t-1} \ . \tag{40}$$

The wealth is updated recursively based on the bet's success:

$$W_t = W_{t-1} + b_t c_t = W_{t-1} - g_t b_t \ . \tag{41}$$

The KT estimator provides a practical betting scheme that adapts to the observed sequence of subgradients by updating the betting fraction as follows:

$$\beta_{t+1} = \frac{t}{t+1}\beta_t - \frac{1}{t+1} g_t \ . \tag{42}$$

This strategy is proven to control long-term miscoverage frequency at the nominal level α provided the nonconformity scores are bounded (Podkopaev et al., 2024).

Algorithm 2 KT-based Adaptive Conformal Predictor

Initialize: $\alpha \in (0,1)$, $W_0 = 1$, $\beta_1 = 0$, $b_1 = 0$.
for $t = 1, 2, \ldots$ **do**
  Produce a forecast $\hat{Y}_t$ and output a set: $\hat{C}_t = [\hat{Y}_t - b_t, \hat{Y}_t + b_t]$
  Observe Y_t and compute error: $S_t = \left|Y_t - \hat{Y}_t\right|$
  Compute $g_t \in \partial \ell^{(1-\alpha)}(b, S_t)|_{b=b_t}$ as per (4);
  Set $W_t = W_{t-1} - g_t b_t$
  Set $\beta_{t+1} = \frac{t}{t+1}\beta_t - \frac{1}{t+1} g_t$
  Set $b_{t+1} = \beta_{t+1} W_t$
end for

K.2 Dynamically-tuned Adaptive Conformal Inference

Dynamically-tuned Adaptive Conformal Inference (DtACI) is an extension of ACI designed to eliminate the sensitivity to the fixed step-size parameter γ, which governs the adaptation rate to distribution shifts (Gibbs

and Candès, 2024). Instead of relying on a single γ, DtACI maintains a set of expert ACI instances running in parallel, each using a different step size γ_k from a candidate grid. At each time step t, the algorithm aggregates the predictions of these experts using an online learning procedure (e.g., exponentially weighted average) to produce a robust conformal radius b_t. The aggregation relies on the empirical quantiles of the current score, defined as $\beta_t := \hat{F}_{t-1}(S_t)$, where $\hat{F}_{t-1}$ is the empirical distribution of the past nonconformity scores. The experts are weighted based on their performance with respect to the pinball loss evaluated at these levels. Specifically, the loss for an expert proposing target level α_t^i is given by:

$$\ell(\beta_t, \alpha_t^i) = \begin{cases} \alpha(\beta_t - \alpha_t^i) & \text{if } \beta_t \geq \alpha_t^i \\ (\alpha - 1)(\beta_t - \alpha_t^i) & \text{if } \beta_t < \alpha_t^i. \end{cases}$$

Algorithm 3 Dynamically-tuned Adaptive Conformal Inference

Input: Observed values $\{\beta_t\}_{1 \leq t \leq T}$, set of candidate γ values $\{\gamma_i\}_{1 \leq i \leq k}$, starting points $\left\{\alpha_1^i\right\}_{1 \leq i \leq k}$, and parameters σ and η.
Initialize: $w_1^i = 1$, $1 \leq i \leq k$.
for $t = 1, 2, \ldots, T$ **do**
  Define the probabilities $p_t^i = w_t^i / \sum_{1 \leq j \leq k} w_t^j$, $\forall 1 \leq i \leq k$
  Output $\overline{\alpha}_t = \sum_{1 \leq i \leq k} p_t^i \alpha_t^i$
  $\overline{w}_t^i = w_t^i \exp(-\eta \ell(\beta_t, \alpha_t^i))$, $\forall 1 \leq i \leq k$
  $\overline{W}_t = \sum_{1 \leq i \leq k} \overline{w}_t^i$
  $w_{t+1}^i = (1 - \sigma)\overline{w}_t^i + \overline{W}_t \sigma / k$
  $err_t^i = \mathbf{1}\left\{Y_t \notin \hat{C}_t(\alpha_t^i)\right\}$, $\forall 1 \leq i \leq k$
  $err_t = \mathbf{1}\left\{Y_t \notin \hat{C}_t(\overline{\alpha}_t)\right\}$
  $\alpha_{t+1}^i = \alpha_t^i + \gamma_i(\alpha - err_t^i)$, $\forall 1 \leq i \leq k$
end for

The default hyperparameters for DtACI follow the experimental setup detailed in Section 4 of Gibbs and Candès (2024). The candidate step sizes are set to be on a logarithmic grid $\Gamma = \{0.001 \cdot 2^k\}_{k=0}^7$, ranging from 0.001 to 0.128, designed to cover a spectrum of regimes from stable to highly reactive. The fixed share parameter is set to $\sigma = 0.001$, and the learning rate η for the expert algorithm is chosen to be e. The starting points $\left\{\alpha_1^i\right\}_{1 \leq i \leq k}$ are all initialized to the nominal miscoverage level α.

K.3 Scale-Free Online Gradient Descent

As mentioned in the main text, standard Online Gradient Descent (OGD) is sensitive to the scale of the nonconformity scores, often requiring careful tuning of the learning rate to ensure reasonable performance. To address this, Bhatnagar et al. (2023) adapted Scale-Free Online Gradient Descent (SF-OGD) (Orabona and Pál, 2018) for conformal prediction. SF-OGD dynamically adjusts the step size by normalizing the current gradient by the Euclidean norm of the history of past gradients. This normalization makes the algorithm robust to the magnitude of the scores without requiring prior knowledge of their bounds. The update rule for the conformal radius b_t is given by:

$$b_{t+1} = b_t - \eta \frac{g_t}{\sqrt{\epsilon + \sum_{i=1}^t g_i^2}}, \tag{43}$$

where g_t is the subgradient of the pinball loss, η is a scalar multiplier controlling the learning rate, and ϵ is a small constant (e.g., 10^{-6}) to ensure numerical stability.

Algorithm 4 SF-OGD Adaptive Conformal Predictor

Input: Target miscoverage $\alpha \in (0, 1)$, learning rate $\eta > 0$.
Initialize: Radius $b_1 = 0$, sum of squared gradients $G_0 = 0$, $\epsilon = 10^{-6}$.
for $t = 1, 2, \ldots$ **do**
  Output set $\hat{C}_t = [\hat{Y}_t - b_t, \hat{Y}_t + b_t]$
  Observe Y_t and compute score $S_t = \left|Y_t - \hat{Y}_t\right|$
  Compute gradient $g_t \in \partial \ell^{(1-\alpha)}(b, S_t)|_{b=b_t}$ as per (4);
  Update sum of squares: $G_t = G_{t-1} + g_t^2$
  Update radius: $b_{t+1} = b_t - \eta \frac{g_t}{\sqrt{G_t + \epsilon}}$
end for

While robust to scale, SF-OGD still requires the selection of the global learning rate η. In our experiments, we select η via grid search from the set $\{0.01, 0.1, 0.25, 1, 10, 25, 100\}$. These candidate values are adopted from the well-tuned choices reported in Podkopaev et al. (2024), ensuring the method can accommodate selected datasets.

K.4 Conformal P/PI Control

Conformal P/PI Control frames the selection of the conformal radius b_t as a feedback control problem, where the goal is to calibrate the coverage error $err_t = \mathbf{1}(y_t \notin \hat{C}_t)$ to the set point α (Angelopoulos et al., 2023). The algorithm combines two components: a Proportional (P) controller, also known as Quantile Tracking, and an Integral (I) controller. The P controller updates the radius using online gradient descent on the quantile loss, adjusting b_t proportional to the instantaneous error $err_t - \alpha$. While effective, P control can suffer from steady-state error. To mitigate this, the PI controller adds an integrator term $r_t(\cdot)$ that acts on the cumulative sum of past errors, $E_t = \sum_{i=1}^{t}(err_i - \alpha)$. The update rule for the radius at time $t + 1$ is given by combining these terms with the previous radius:

$$b_{t+1} = b_t + \eta(err_t - \alpha) + r_t(E_t), \tag{44}$$

where η is the learning rate and r_t is a saturation function (e.g., a tangent function) designed to stabilize coverage under arbitrary distribution shifts (Angelopoulos et al., 2023).

Algorithm 5 Conformal P/PI Control

Input: Target miscoverage α, learning rate $\eta > 0$, integrator function $r_t(\cdot)$.
Note: For P Control, set $r_t(x) = 0$.
Initialize: Radius $b_1 = 0$, cumulative error $E_0 = 0$.
for $t = 1, 2, \ldots$ **do**
  Receive input x_t and forecast $\hat{Y}_t$
  Output set $\hat{C}_t = [\hat{Y}_t - b_t, \hat{Y}_t + b_t]$
  Observe Y_t and compute error indicator $err_t = \mathbf{1}\{Y_t \notin \hat{C}_t\}$
  Update cumulative error: $E_t = E_{t-1} + (err_t - \alpha)$
  Compute P-step: $\delta_P = \eta(err_t - \alpha)$
  Compute I-step: $\delta_I = r_t(E_t)$
  Update radius: $b_{t+1} = b_t + \delta_P + \delta_I$
end for

The hyperparameters for the P/PI controller are set following the heuristics provided in Appendix B of Angelopoulos et al. (2023). The proportional gain (learning rate) is typically set adaptively as $\eta = \lambda \hat{B}_t$, where $\hat{B}_t$ is the maximum score observed in a trailing window (or a global bound) and $\lambda \in (0, 1]$ is a scaling factor (0.1 was recommended as a good default). For the PI controller, the integrator is defined as $r_t(E) = K_I \tan(E \log(T)/(TC_{sat}))$. The constant K_I aligns the integrator's output with the scale of the nonconformity scores; it is recommended to be set to a hypothesized upper bound on the scores (e.g., $K_I \approx \max S_t$). The parameter C_{sat} controls the saturation point of the integrator and is derived from a theoretical guarantee

to ensure the miscoverage does not exceed a tolerance δ by time T (e.g., $C_{sat} \approx \frac{2}{\pi}\log(T\delta)$). In the original implementation, both K_I and C_{sat} are often pre-tuned or fixed heuristically for specific datasets to ensure stability.

Specifically in our experiments, the scalar multipliers λ are uniformly selected from the fixed grid $\{0, 0.05, 0.1, 0.5, 1\}$[3] $\{0, 0.05, 0.1, 0.5, 1\}$. for all datasets. For the PI controller, the integrator parameters are pre-tuned heuristically as described previously, while the same grid of λ values is used for the proportional component.

Note that we do not compare against the full Conformal PID framework, specifically its derivative (D) component known as *Scorecasting* (Angelopoulos et al., 2023). Scorecasting introduces a secondary modeling layer that fundamentally alters the score distribution by predicting and residualizing errors, effectively transforming the problem into an easier one. To ensure a fair assessment of the conformal update rules themselves and to maintain consistency with the standard evaluation established in prior literature (Gibbs and Candes, 2021; Gibbs and Candès, 2024), we restrict our comparison to methods that adapt to the original sequence of nonconformity scores without modification.

K.5 Trivial Predictor

The Trivial Predictor serves as a minimal baseline designed to verify the validity of coverage metrics. It guarantees that the empirical coverage exactly matches the target level $1-\alpha$ at fixed periodic intervals, completely independent of the data distribution. The method approximates the target coverage probability as a rational fraction $K/N \approx 1-\alpha$. It then generates a deterministic, periodic sequence of prediction sets consisting solely of infinite sets ($b_t = \infty$) and zero-radius sets ($b_t = 0$). By distributing the K infinite sets as evenly as possible over each cycle of length N, the predictor ensures that the cumulative coverage error returns to exactly zero at the end of every cycle, providing perfect validity but practically useless set predictions.

Algorithm 6 Trivial Predictor (Deterministic Cyclic Coverage)

Input: Target miscoverage $\alpha \in (0, 1)$.
Initialize: Rational approximation $1-\alpha \approx \frac{K}{N}$ (e.g., via continued fractions).
Initialize: Time step $t = 0$.
for $t = 1, 2, \ldots$ **do**
  Determine current radius b_t based on previous update.
  Output set $\hat{C}_t = [\hat{Y}_t - b_t, \hat{Y}_t + b_t]$
  Calculate position in cycle: $i = t \pmod N$
  Compute accumulated coverage credits:
    $acc_{curr} = \lfloor \frac{i \cdot K}{N} \rfloor$
    $acc_{next} = \lfloor \frac{(i+1) \cdot K}{N} \rfloor$
  if $acc_{next} > acc_{curr}$ **then**
    Set next radius: $b_{t+1} = \infty$ {Output Full Set}
  else
    Set next radius: $b_{t+1} = 0$ {Output Empty Set}
  end if
end for

L Score Growth Demonstration

In Section 3, we introduced the polynomial growth assumption $S_t \leq Dt^q$, which relaxes the standard boundedness ($S_t \leq C$) used in prior literature. Figure 2 illustrates the practical implication of this relaxation. To see why this matters in practice, consider the Apple (AAPL) dataset used in our experiments. It is clear that the nonconformity scores follow a roughly quadratic trend ($q \approx 2$). Financial time series frequently exhibit volatility clustering and price drift that violate static bounds. Our analysis extends guarantees to the polynomial-growth regime, which provides a tighter fit for such non-stationary real-world data.

[3] In fact, 0.1 was recommended by Angelopoulos et al. (2023) as a good default.

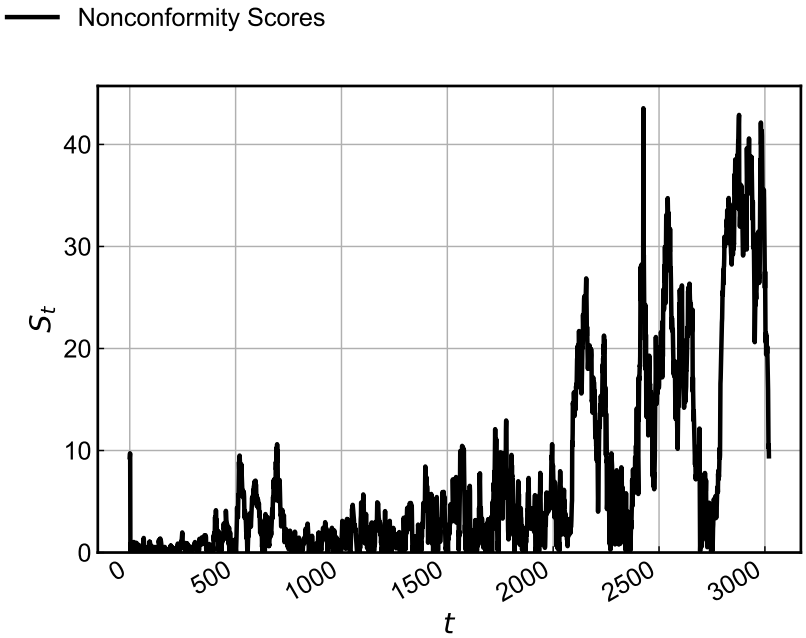

Figure 2: Observed nonconformity scores for AAPL stock returns.

M α-Correction

In this section, we show how to minimally change *any* OCP algorithm to obtain a possibly better tracking of the desired coverage level, without changing the Pareto frontier curves or the rate of the theoretical guarantees.

First of all, in the online setting *we cannot guarantee coverage at any time step t, but only asymptotically.* Hence, what we propose is only a heuristic, but a theoretically principled one. The basic idea is that, from the plots, it seems that most of the algorithms tend to undercover. Hence, one can run the algorithm with a slightly inflated parameter α, to obtain a better coverage.

Now, by how much do we inflate α? Setting the desired coverage to $\alpha + \frac{k}{\sqrt{T}}$, for any constant $k > 0$, and assuming that the algorithm guarantees a convergence of the coverage of $\mathcal{O}(1/\sqrt{T})$ or worse, will produce exactly the same rate of convergence for the coverage. Also, the Pareto frontier will not change as well, because we are simply using a different setting of α, so we are just moving along the Pareto frontier curve.

N Full Results for AXP Dataset

In this section, we provide a comprehensive analysis of the American Express (AXP) dataset, complementing the efficiency and calibration results presented in Section 5. Note that an initial warm-up period of 100 days is used for training the initial base forecaster. Additionally, for all plots presented in this section, we discard the first 50 days of the evaluation period as a burn-in.

N.1 Target Calibration

Beyond efficiency, a reliable OCP algorithm must also track the user-specified target $1 - \alpha$. Figure 3 plots the realized versus target coverage for $\alpha \in [0.05, 0.25]$. The results confirm that UP-OCP and the baselines maintain calibration within a tight ± 0.03 tolerance band (dashed lines) over all targets. In Appendix M, we also provide a heuristic to improve the tracking of any OCP algorithm.

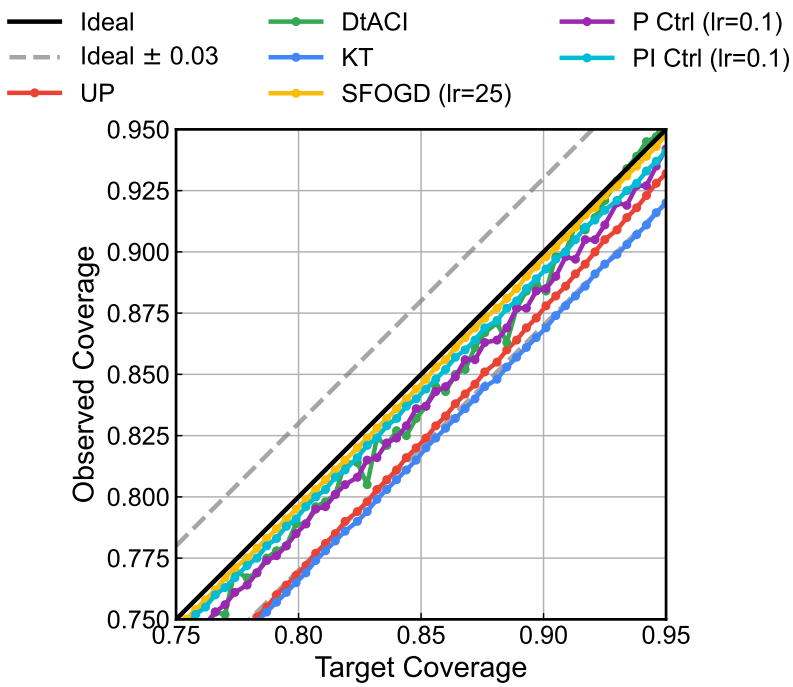

Figure 3: Realized vs. target coverage on the AXP dataset.

Local Adaptivity. In the main text, we noted that aggregate metrics like marginal coverage can mask significant local failures, such as error clustering or instability. To visualize these behaviors, Figures 4 through 8 provide 1-vs-1 comparisons of local adaptivity between UP-OCP and all parameter-free and tuned baselines.

In each figure, the top left panel shows the local coverage computed over a rolling window of 100 days, with the dashed black line indicating the target coverage level ($1-\alpha = 95\%$). The top right panel displays the local width of the prediction intervals ($2b_t$) using the same window size. The bottom panel illustrates the raw prediction sets $\hat{C}_t$ around the true observation (central black line).

We observe that UP-OCP maintains local coverage tightly around the 95% target, with no significant swings. In contrast, KT exhibits volatility, where the local coverage drops below 75% (e.g., during 2009 and 2015).

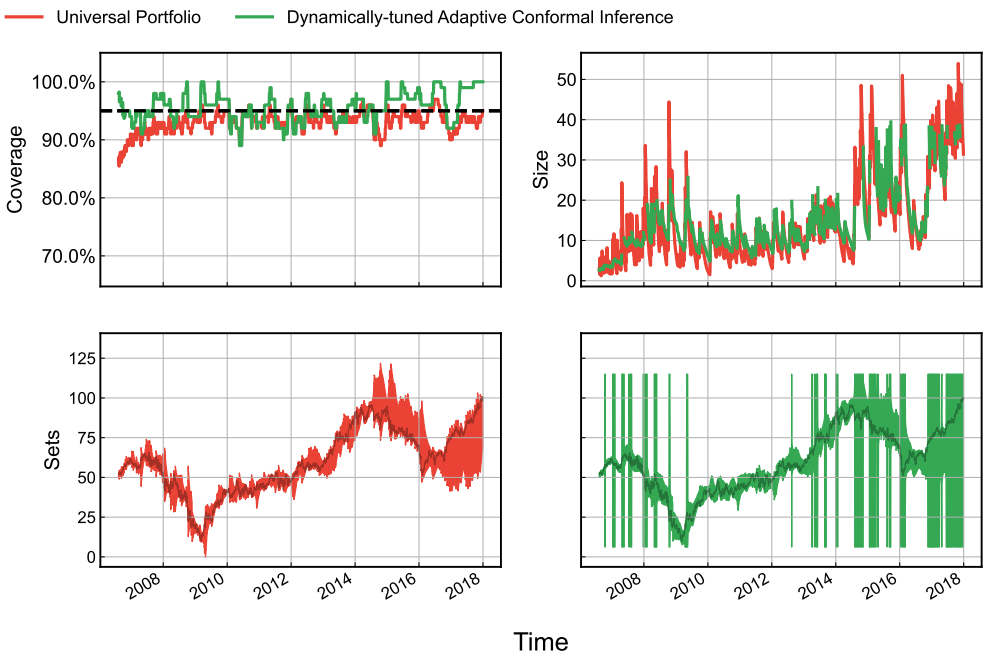

Figure 4: UP-OCP vs. DtACI for forecasting AXP stock return.

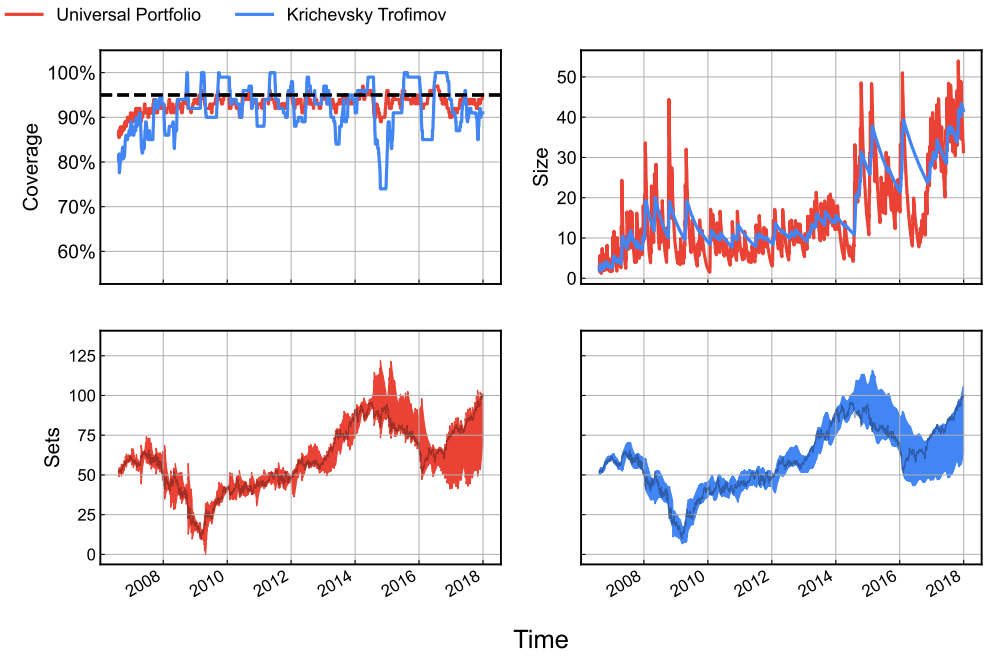

Figure 5: As in Figure 4, UP-OCP vs. KT.

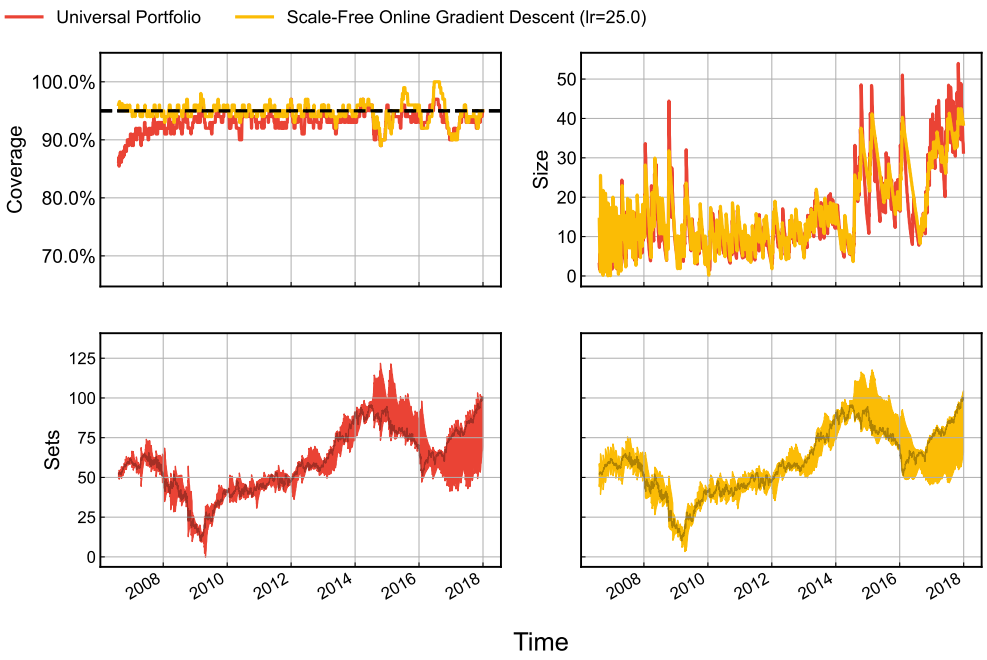

Figure 6: As in Figure 4, UP-OCP vs. SFOGD (lr=25).

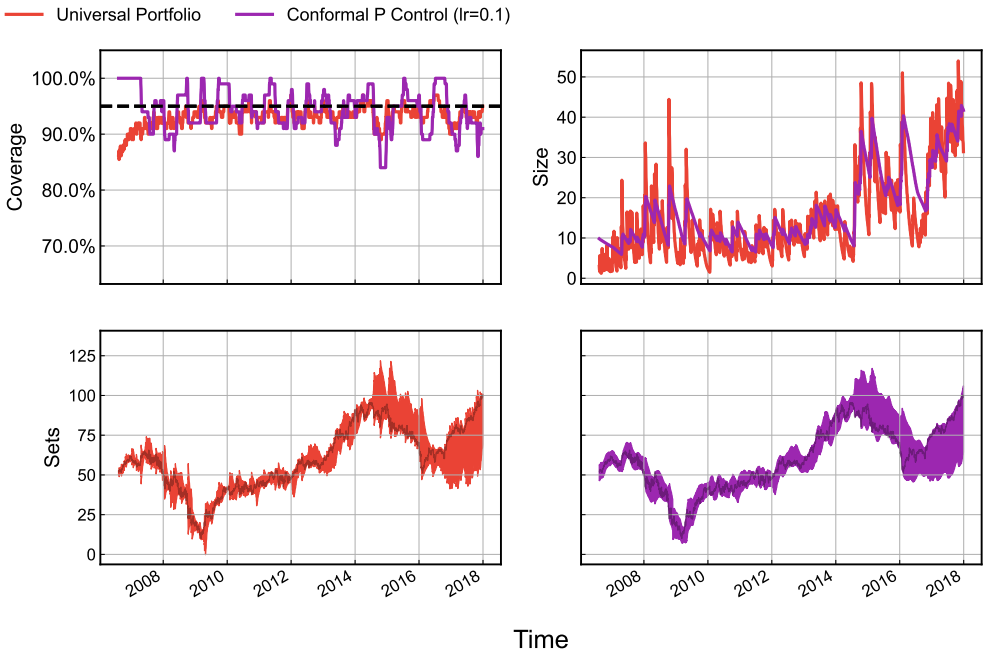

Figure 7: As in Figure 4, UP-OCP vs. P Ctrl (lr=0.1).

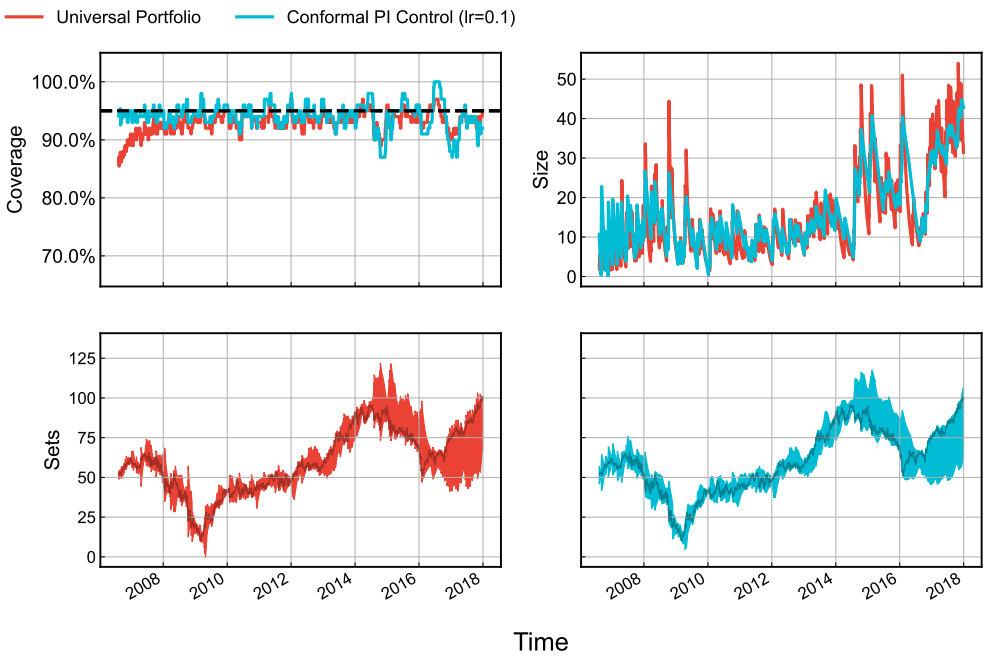

Figure 8: As in Figure 4, UP-OCP vs. PI Ctrl (lr=0.1).

For completeness, Table 2 extends the results from the main text to include the heuristic Conformal P and PI Controllers (Angelopoulos et al., 2023). These methods were tuned via grid search to maximize performance.

Table 2: Quantitative Comparison on the AXP Dataset.

	UP	KT	DtACI	SFOGD (lr=25)	P Ctrl (lr=0.1)	PI Ctrl (lr=0.1)
Marginal coverage	0.932	0.92	0.956	0.948	0.942	0.941
Longest err sequence	**4**	15	6	**3**	7	**5**
Average set size	**14.8**	16.9	∞	16.4	16.8	16.1
Median set size	**11.5**	12.9	12.6	13.8	13.1	13.2
75% quantile set size	**18.8**	24.9	21.8	21.5	21.5	20.9
90% quantile set size	**32.3**	32.5	∞	32.3	32.4	**32**
95% quantile set size	38	36.1	∞	36.1	36.6	35.9

More Pareto Frontiers and Target-level Tracking. In Section 5, we prioritized the presentation of *average* prediction set size to penalize the infinite sets produced by algorithms like DtACI. However, robust statistics such as the median and 75% quantile provide insight into the typical performance of the algorithms, ignoring the heavy tails. We present additional Pareto frontiers for these metrics in Figures 9 through 11.

These results clarify that DtACI is not inefficient on average days; its median performance overlaps with UP-OCP. The inefficiency is driven entirely by its inability to handle tail events properly. UP-OCP, however, dominates or matches all baselines across both mean and robust metrics, proving it is both stable in the worst case and efficient in the typical case.

Figure 10 clarifies that DtACI is not inefficient on average days; its median performance overlaps with UP-OCP. The inefficiency is driven entirely by its inability to handle tail events properly, which is shown by the divergence in the mean (Figure 9) and 75% quantile metrics (Figure 11). UP-OCP, however, dominates or matches all baselines across both mean and robust metrics, proving it is both stable in the worst case and efficient in the typical case.

Target-level Tracking. Finally, Figure 12 illustrates the ability of the algorithms to track the user-specified

target coverage across a spectrum of α values.

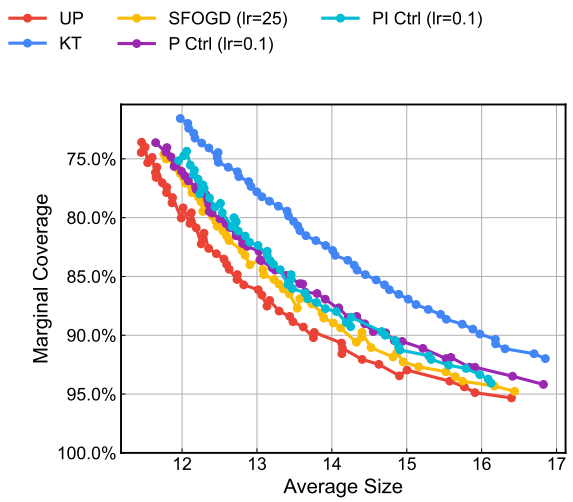

Figure 9: Mean prediction set sizes on AXP.

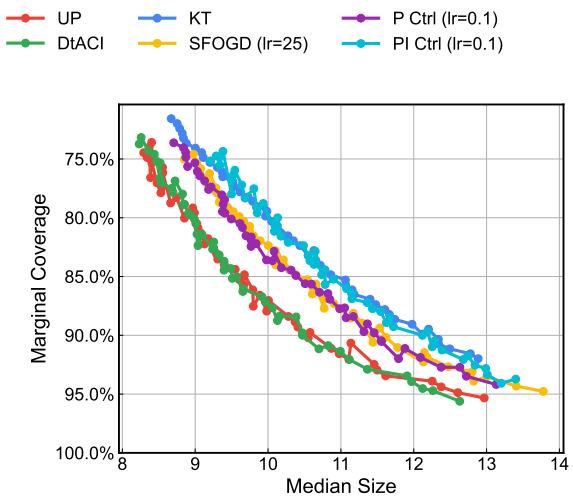

Figure 10: Median prediction set sizes on AXP.

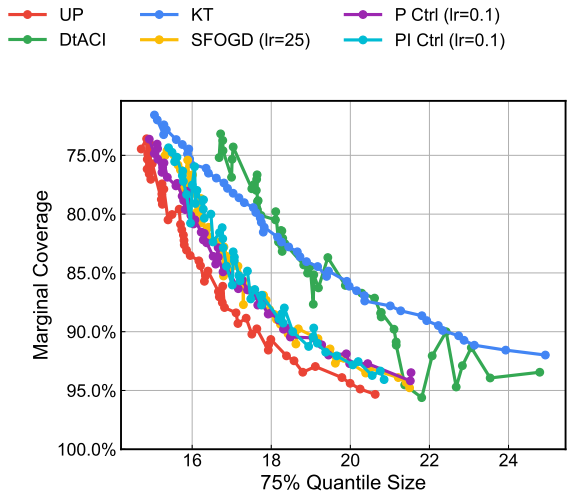

Figure 11: 75% quantile prediction set sizes on AXP.

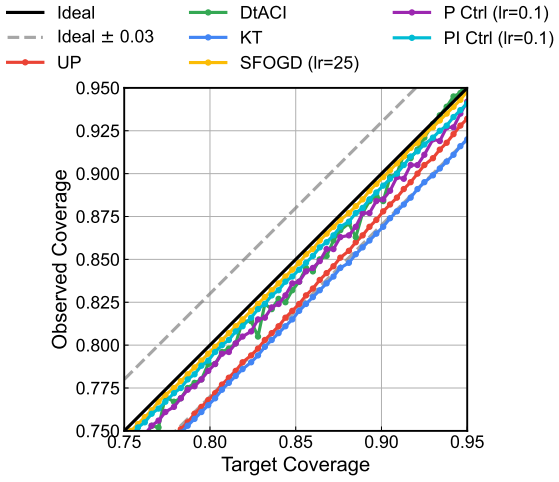

Figure 12: Realized vs. target coverage on AXP. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

O Results for Synthetic Sinusoid

We analyze the performance on a synthetic dataset designed to test adaptivity to periodic volatility. Following Angelopoulos et al. (2023, Appendix F.5), we generate nonconformity scores S_t as a sinusoid with Gaussian noise: $S_t = \max(0, [\sin((2\pi t)/P) + 0.5]S_{\text{mag}} + S_{\text{min}} + \epsilon_t)$, where $\epsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$. We fix the period $P = 200$, magnitude $S_{\text{mag}} = 10$, minimum offset $S_{\text{min}} = 2$, and noise scale $\sigma = 0.3$. The total sequence length is $T = 3000$, and we report results averaged over 10 independent trials.

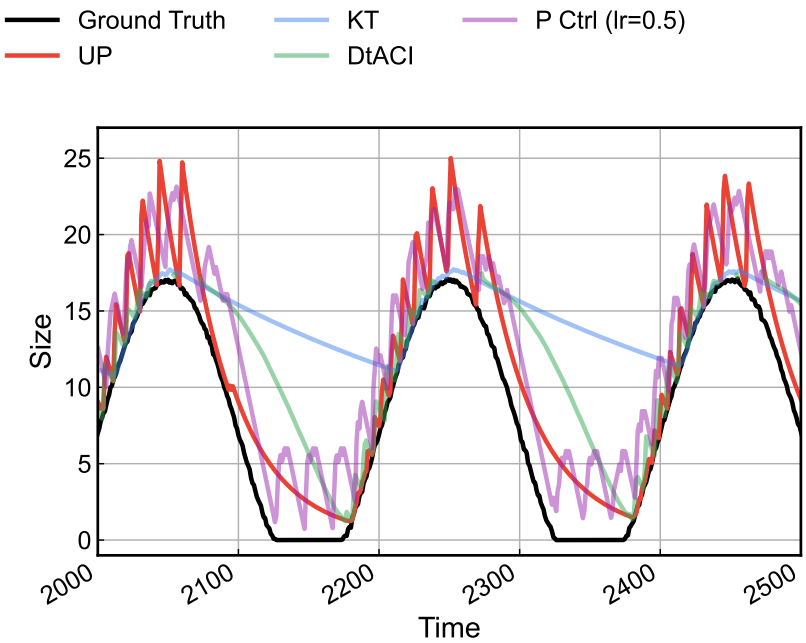

Figure 13: Evolution of interval widths on the synthetic sinusoidal dataset. Optimal performance is the bottom-left corner (tightest sets for highest coverage).

Tracking Dynamics. Figure 13 visualizes the evolution of interval widths over a representative window $t \in [2000, 2500]$. The ground truth (black) exhibits a clear periodic pattern. UP-OCP (red) tracks this, expanding rapidly during high-volatility phases ($t \approx 2050$) to guarantee coverage, then shrinking as the noise variance decreases. In contrast, KT (blue) shows significant lag and fails to reduce interval widths sufficiently during low-noise periods. DtACI (green) shows strong instability; the trace terminates after $t > 2400$ (marked by missing values in the plot), indicating the algorithm has diverged and is outputting *infinite* sets to reach coverage. Again, UP-OCP matches the best tuned baseline, P-Control (purple), without hyperparameter tuning. Other parameterized baselines behave similarly to the P-Controller; see Appendix P.5.

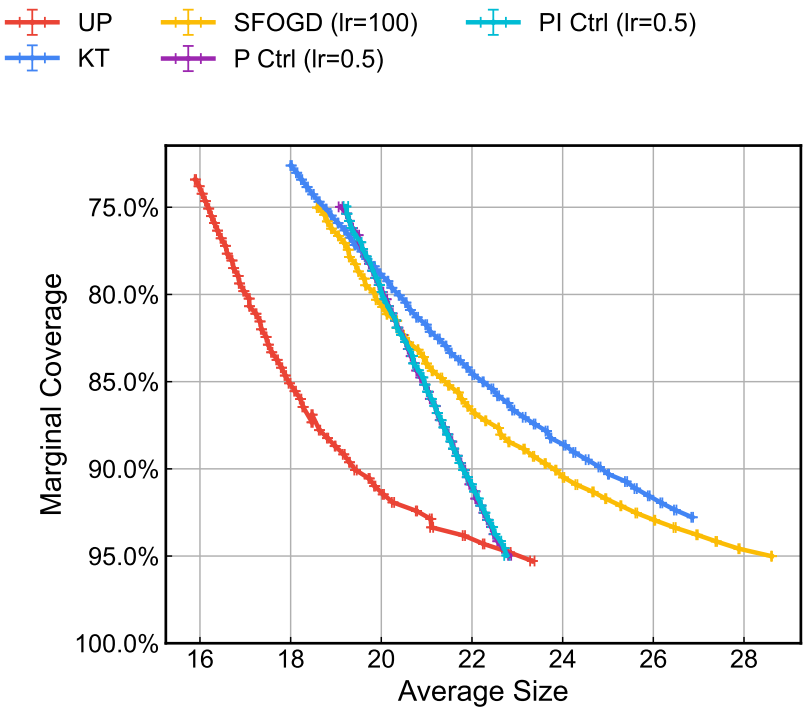

Figure 14: Plot of the realized marginal coverage against the average prediction set size, averaged over 10 independent random seeds. Error bars indicate the standard error of the mean along both axes.

Pareto Frontiers. Figure 14 quantifies the stability of these findings by showing error bars over random repetitions. Consistent with the AXP results, UP-OCP (red) dominates the baselines. The vertical error bars are negligible, confirming that all methods satisfy validity. In contrast, the size metrics show larger oscillations.

P Additional Experiments

To demonstrate generalization of our method, we extend our evaluation to a diverse set of real-world and synthetic benchmarks. These include three additional major stocks (AAPL, AMZN, GOOGL), an electricity demand dataset (NSW), and three synthetic environments designed to test adaptivity (sinusoid, stationary wavelet, and quadratic drift). The observed behaviors across these experiments remain qualitatively consistent with the findings from the AXP dataset. Therefore, we omit a detailed discussion to avoid redundancy and present the following figures and tables for completeness. We will pinpoint interesting observations wherever applicable.

More on Experimental Setup. For the financial datasets (AAPL, AMZN, GOOGL), we follow the same protocol as the AXP experiment: we employ the Prophet model (Taylor and Letham, 2018) as the base forecaster, targeting a miscoverage rate of $\alpha = 0.05$ with an initial burn-in period of $T_{\text{burnin}} = 100$ days. For the electricity demand dataset, we utilize a standard Autoregressive (AR) model with a burn-in of $T_{\text{burnin}} = 300$ steps to capture the high-frequency intraday seasonality. The synthetic experiments generate nonconformity scores directly to isolate specific distributional shifts (periodicity, sparse spikes, and drift), also using a burn-in of 300 steps. To ensure statistical significance, all synthetic results are reported as averages over 10 independent random seeds, with error bars in figures denoting the standard error of the mean. In all comparisons, we evaluate UP-OCP against the full suite of parameter-free (KT, DtACI) and optimized baselines (SF-OGD, P-Control, PI-Control) described in Section 5.

Take Away. It is worth noting that the optimal hyperparameters for the baseline methods vary across datasets. We illustrate this variability in Appendix P, highlighting that no single hyperparameter configuration yields consistent performance across all benchmarks. This sensitivity necessitates dataset-specific tuning, a requirement that our parameter-free UP-OCP method avoids.

P.1 AAPL Dataset

Local Adaptivity.

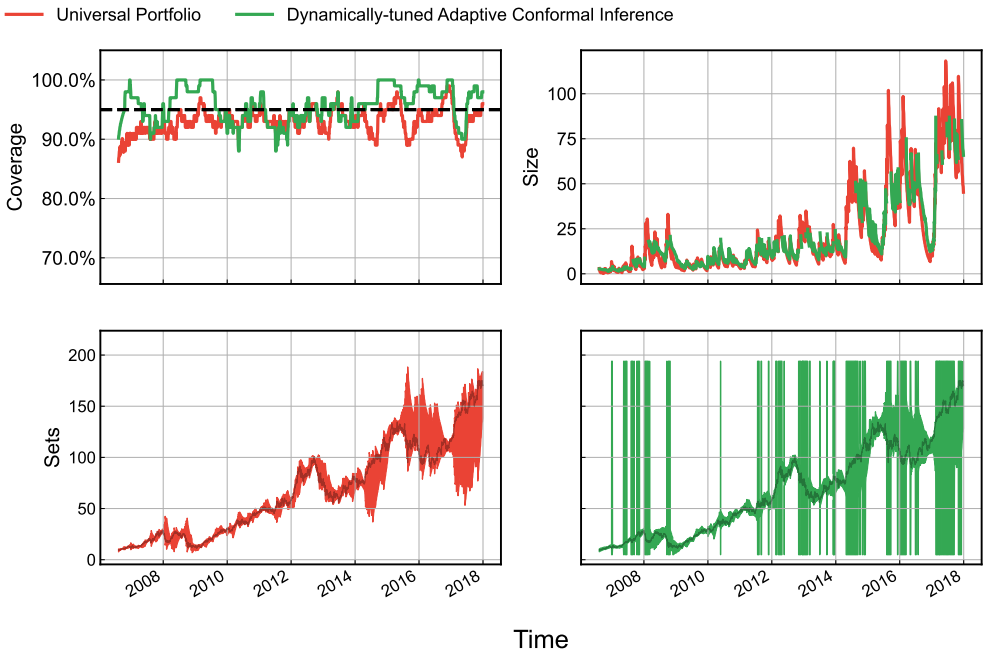

Figure 15: UP-OCP vs. DtACI for forecasting AAPL stock return.

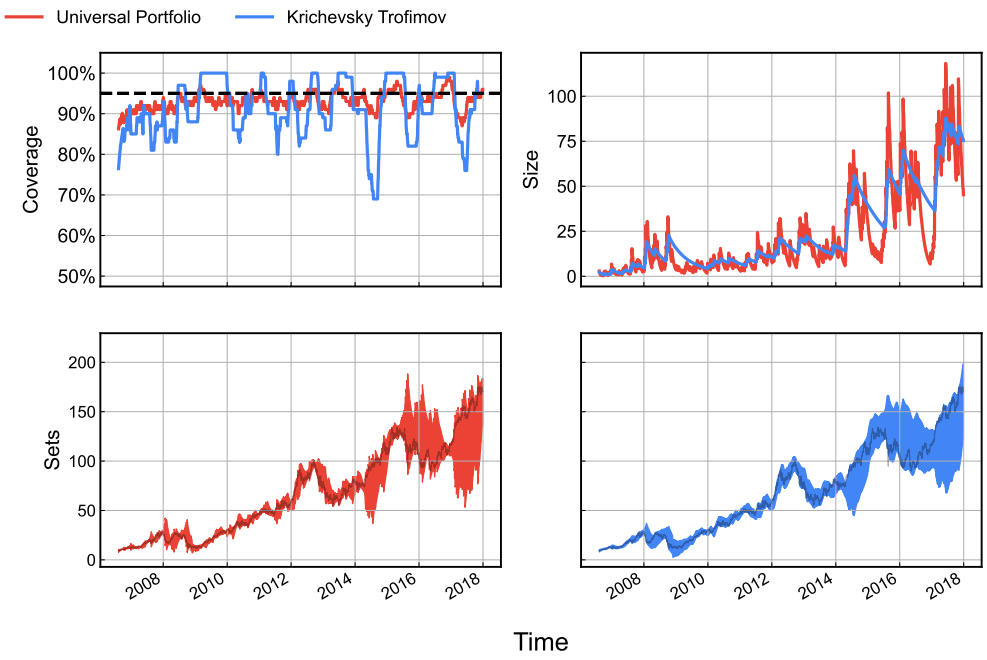

Figure 16: As in Figure 15, UP-OCP vs. KT.

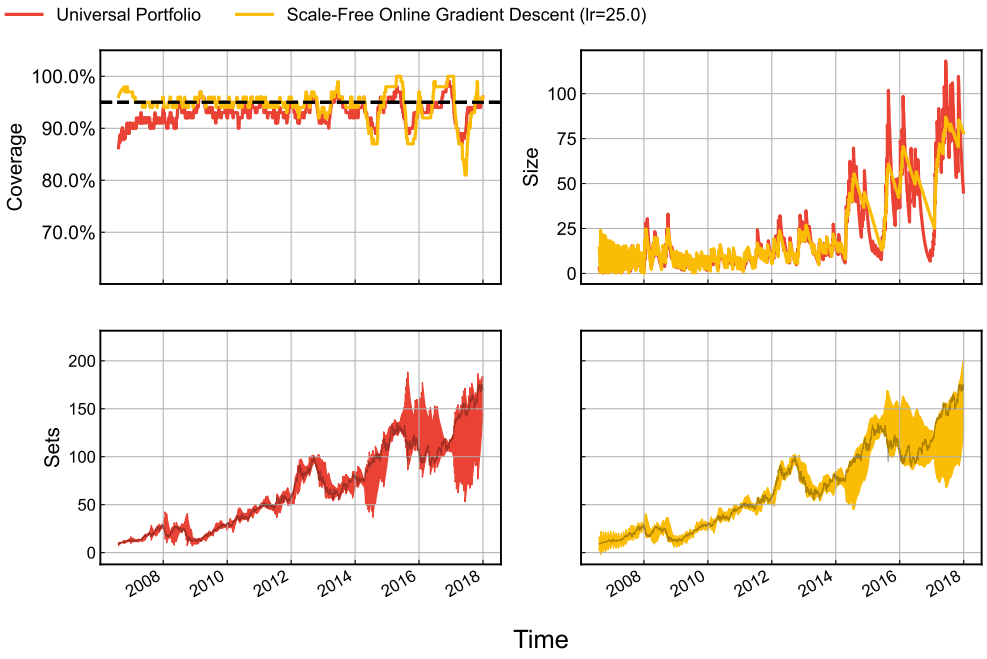

Figure 17: As in Figure 15, UP-OCP vs. SFOGD (lr=25).

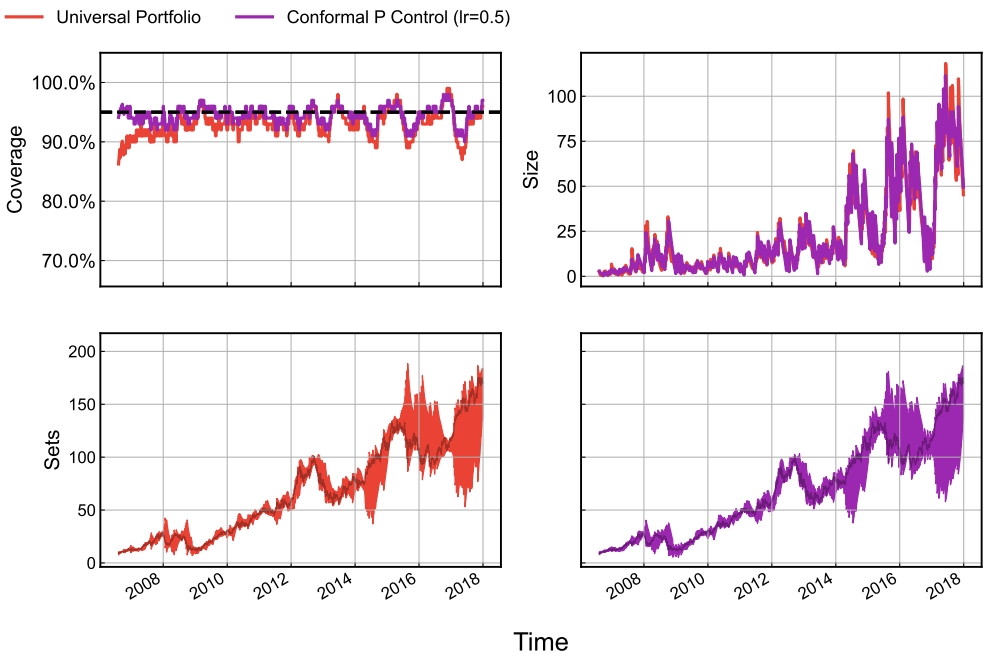

Figure 18: As in Figure 15, UP-OCP vs. P Ctrl (lr=0.5).

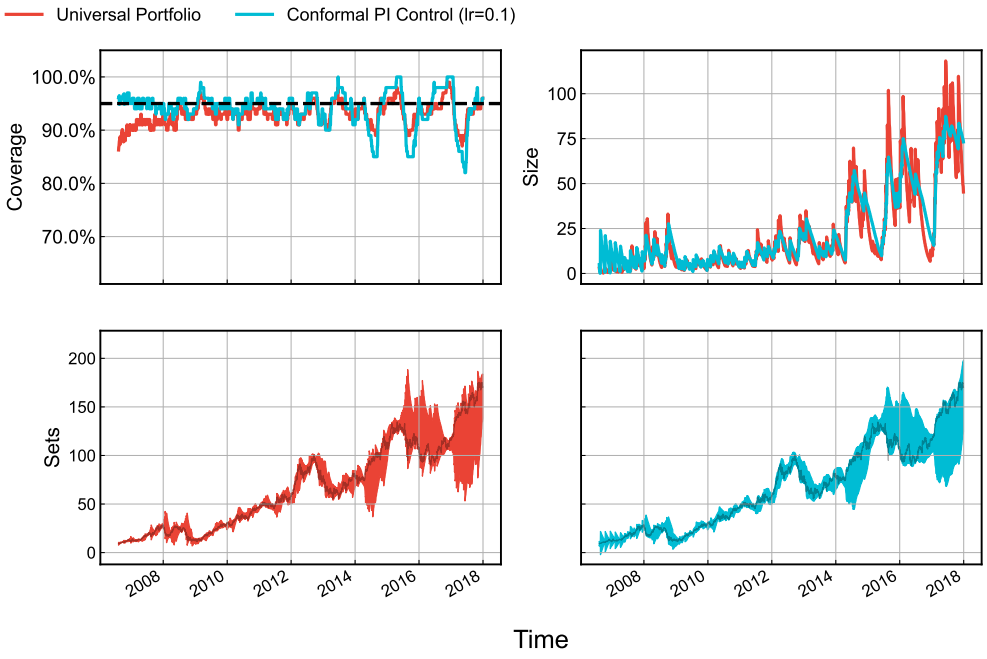

Figure 19: As in Figure 15, UP-OCP vs. PI Ctrl (lr=0.1).

Table 3: Quantitative Comparison on the AAPL Dataset.

	UP	KT	DtACI	SFOGD (lr=25)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.1)
Marginal coverage	0.932	0.915	0.958	0.945	0.945	0.941
Longest err sequence	**2**	16	4	4	**2**	5
Average set size	**21.6**	24.4	∞	23.7	22.8	23.2
Median set size	**11.7**	14.5	14	13.8	13	13.7
75% quantile set size	**27.3**	39.8	55.8	36.9	29.2	33.3
90% quantile set size	60.1	58.8	∞	59	63.7	59.8
95% quantile set size	75.9	75.9	∞	74	77.4	74.5

More Pareto Frontiers and Target-level Tracking.

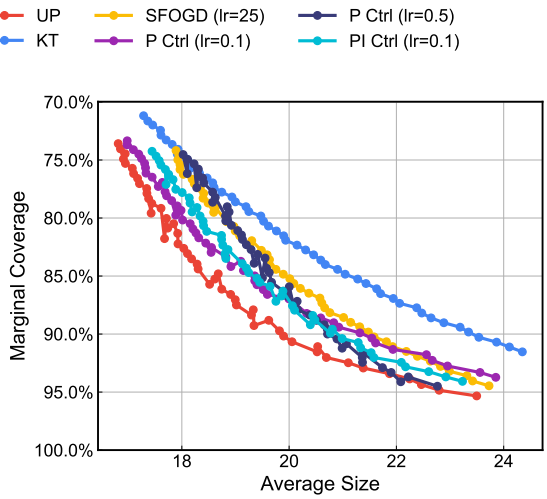

Figure 20: Mean prediction set sizes on AAPL.

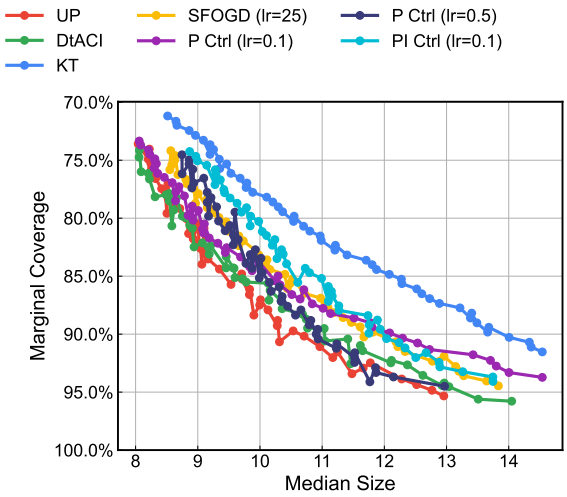

Figure 21: Median prediction set sizes on AAPL.

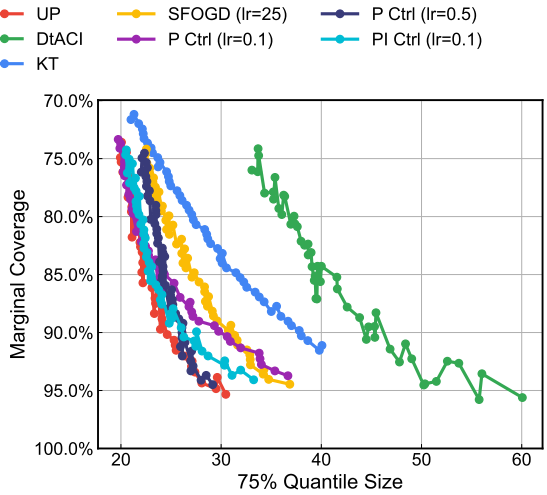

Figure 22: 75% quantile prediction set sizes on AAPL.

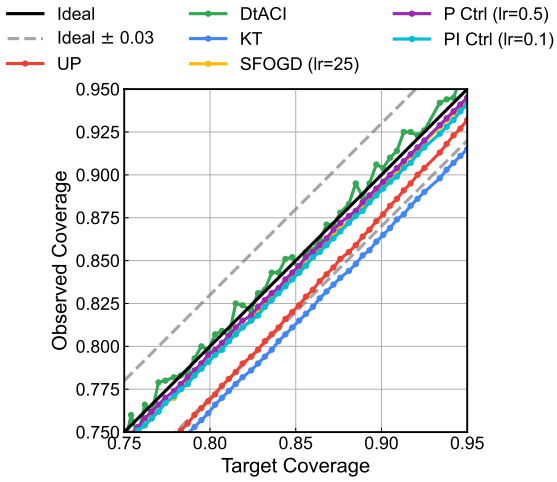

Figure 23: Realized vs. target coverage on AAPL. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

Remark P.1. The results here on all three metric offer a compelling demonstration of adaptivity. UP-OCP (red) does not simply outperform a single baseline; rather, it effectively automates the hyperparameter selection process across the target levels.

Observe the behavior of the tuned P-Controllers (lr=0.1, purple): at moderate targets (75–85% coverage), the controller is optimal, while the higher-gain controller (lr=0.5, dark blue) is less efficient. Conversely, at high targets (90–95%), the high-gain controller becomes necessary to maintain tight sets, while the low-gain version falls behind. UP-OCP remarkably matches the performance of the *best-tuned* baseline in *each* specific regime. It is able to align with the purple curve at lower targets and the dark blue curve at higher targets, without requiring any manual tuning or gain scheduling.

P.2 AMZN Dataset

Local Adaptivity.

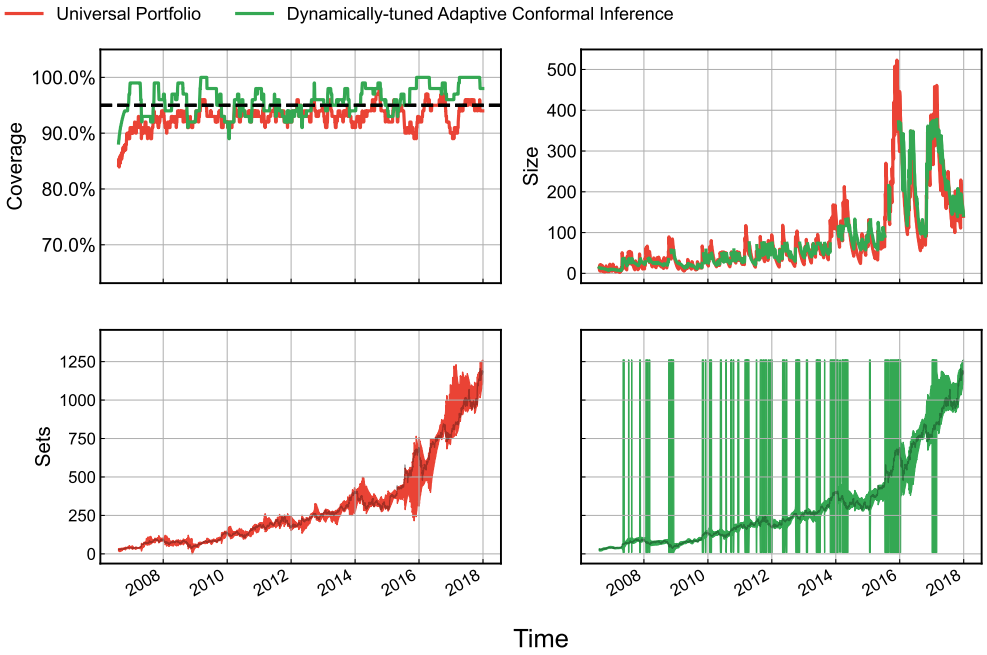

Figure 24: UP-OCP vs. DtACI for forecasting AMZN stock return.

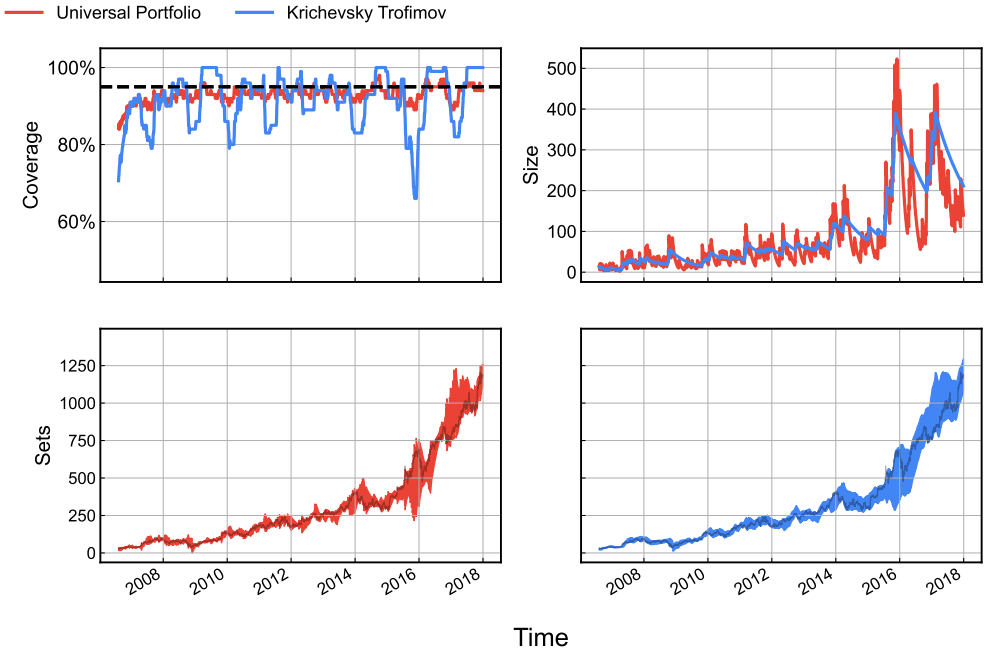

Figure 25: As in Figure 24, UP-OCP vs. KT.

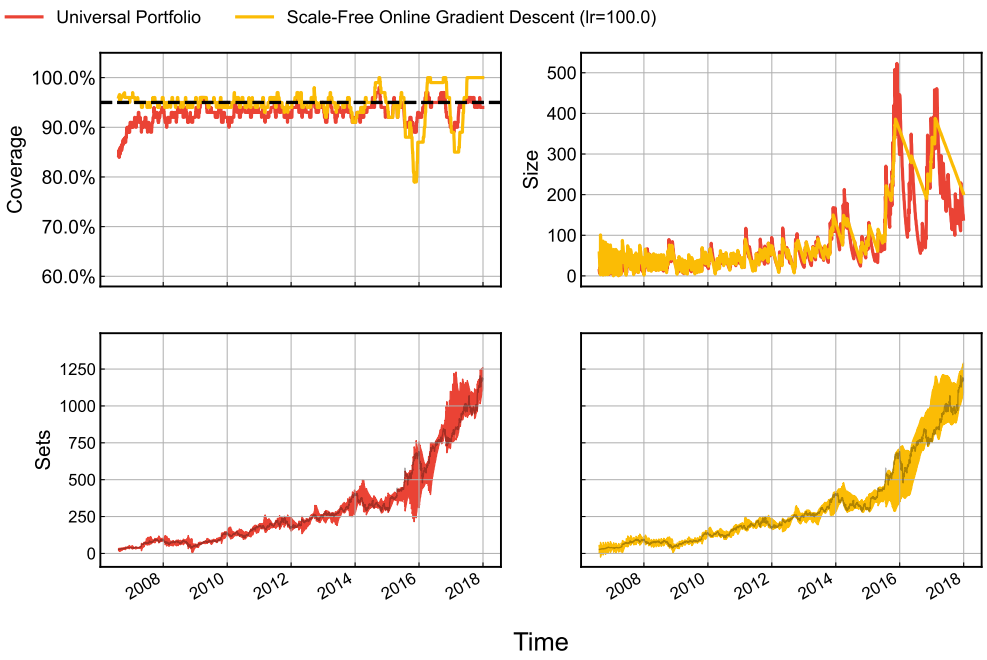

Figure 26: As in Figure 24, UP-OCP vs. SFOGD (lr=100).

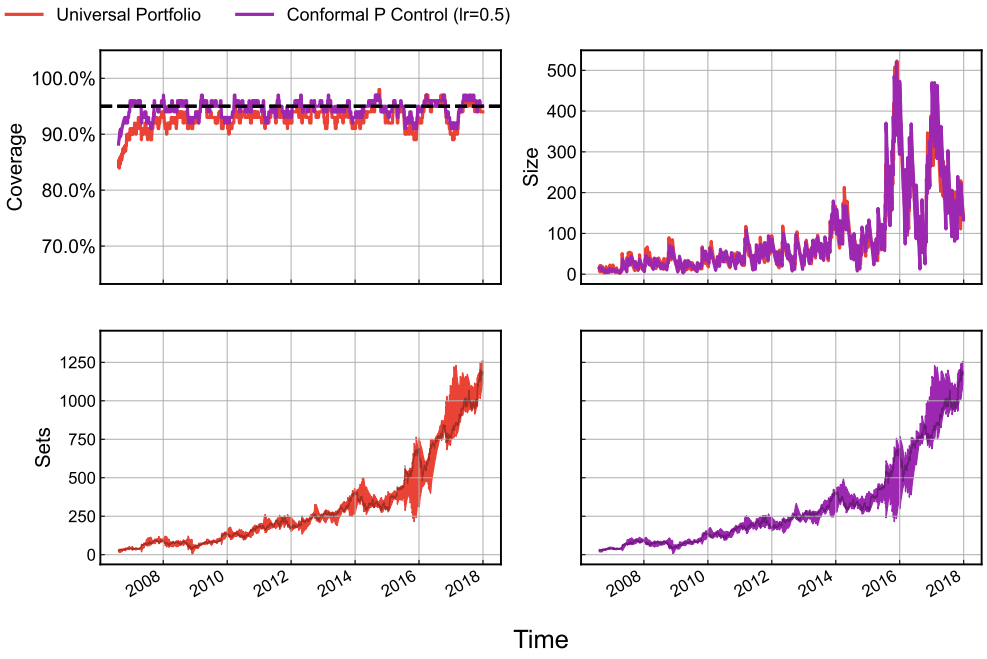

Figure 27: As in Figure 24, UP-OCP vs. P Ctrl (lr=0.5).

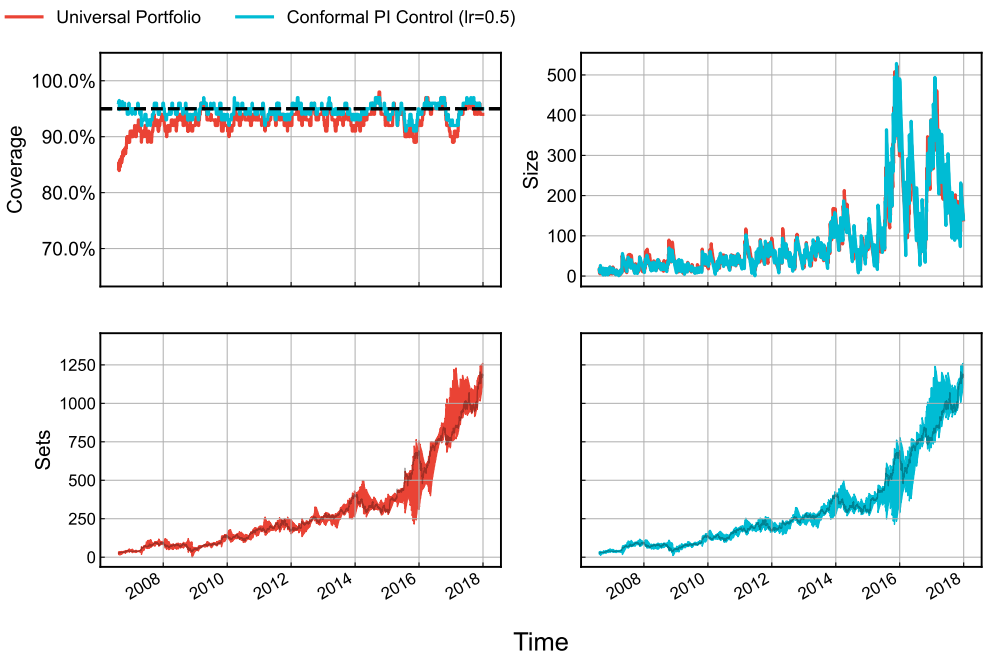

Figure 28: As in Figure 24, UP-OCP vs. PI Ctrl (lr=0.5).

Table 4: Quantitative Comparison on the AMZN Dataset.

	UP	KT	DtACI	SFOGD (lr=100)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.5)
Marginal coverage	0.931	0.919	0.962	0.947	0.946	0.946
Longest err sequence	**3**	18	4	4	**3**	**3**
Average set size	**82.8**	99.4	∞	102	88.9	89.3
Median set size	**49.1**	57.2	60.4	57.5	51.5	51
75% quantile set size	**97.4**	113	274	117	107	108
90% quantile set size	**208**	276	∞	288	234	239
95% quantile set size	**292**	330	∞	335	324	321

More Pareto Frontiers and Target-level Tracking.

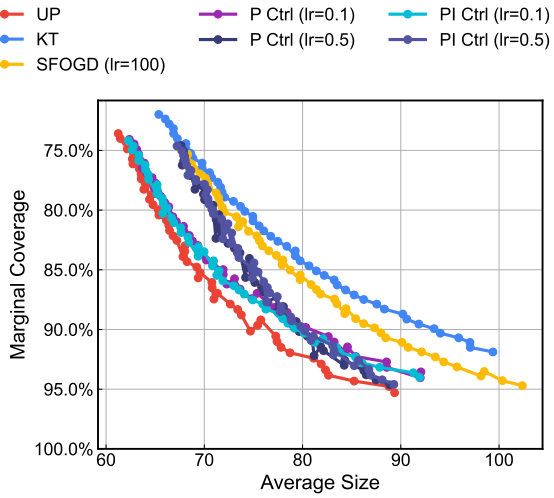

Figure 29: Mean prediction set sizes on AMZN.

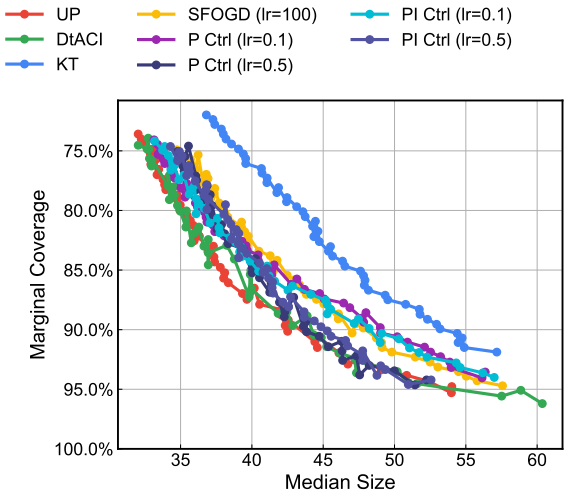

Figure 30: Median prediction set sizes on AMZN.

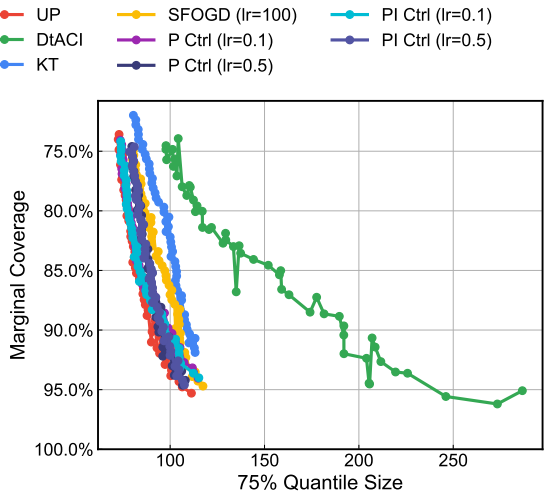

Figure 31: 75% quantile prediction set sizes on AMZN.

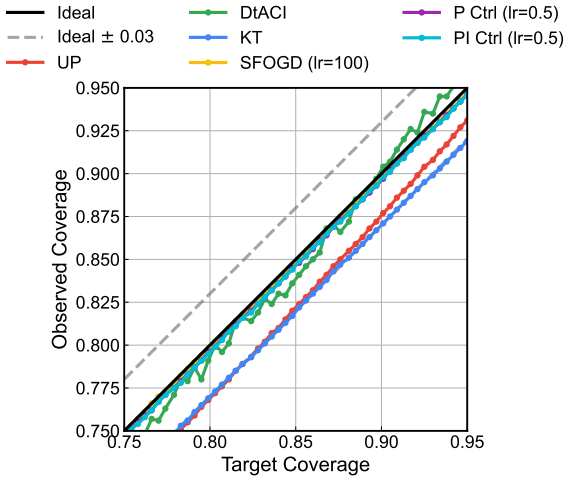

Figure 32: Realized vs. target coverage on AMZN. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

P.3 GOOGL Dataset

Local Adaptivity.

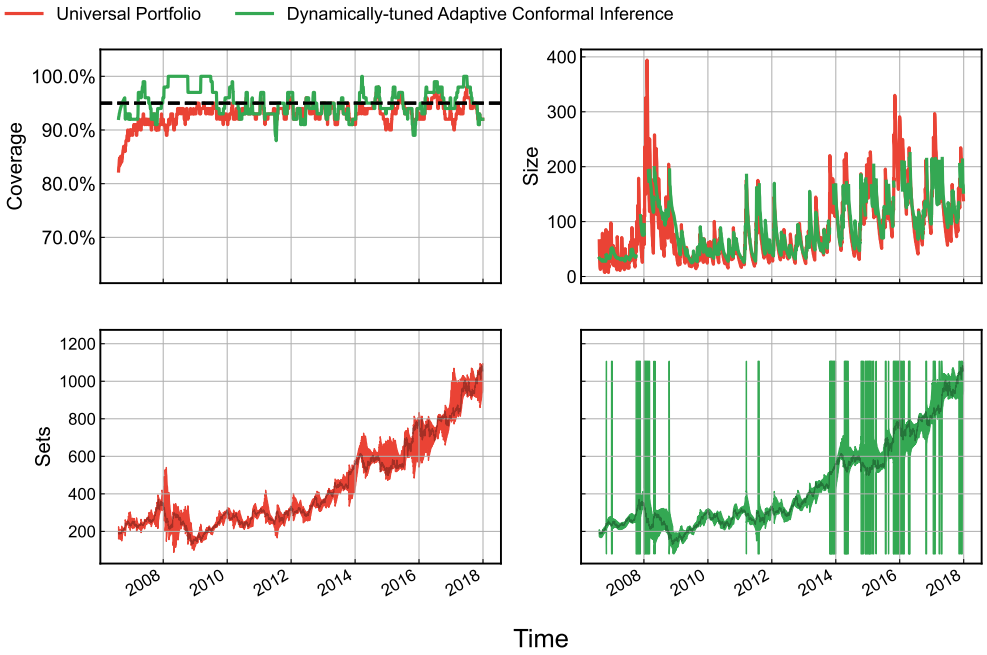

Figure 33: UP-OCP vs. DtACI for forecasting GOOGL stock return.

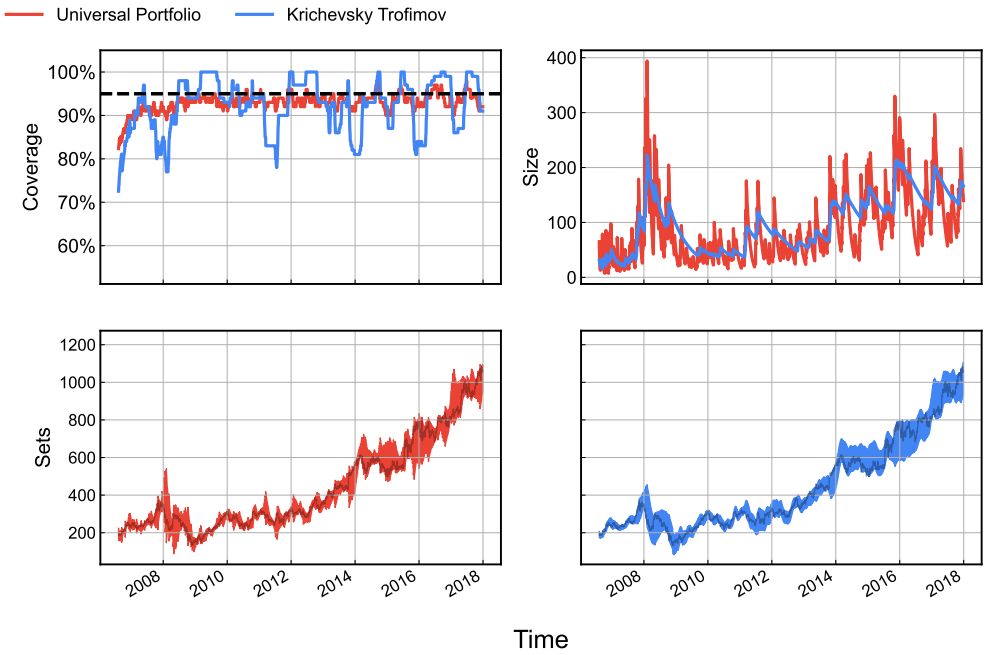

Figure 34: As in Figure 33, UP-OCP vs. KT.

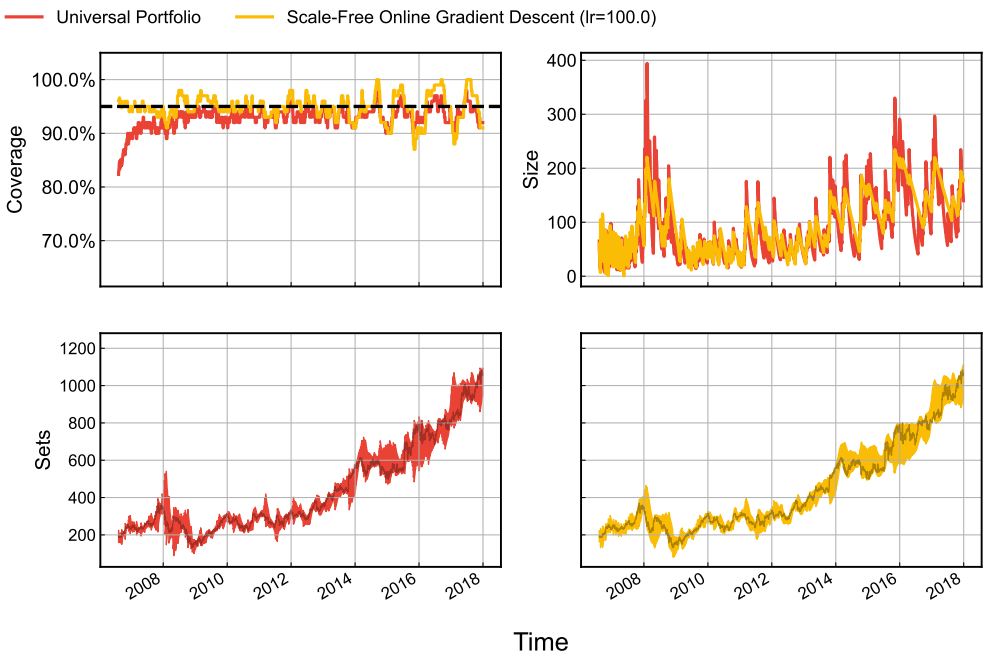

Figure 35: As in Figure 33, UP-OCP vs. SFOGD (lr=100).

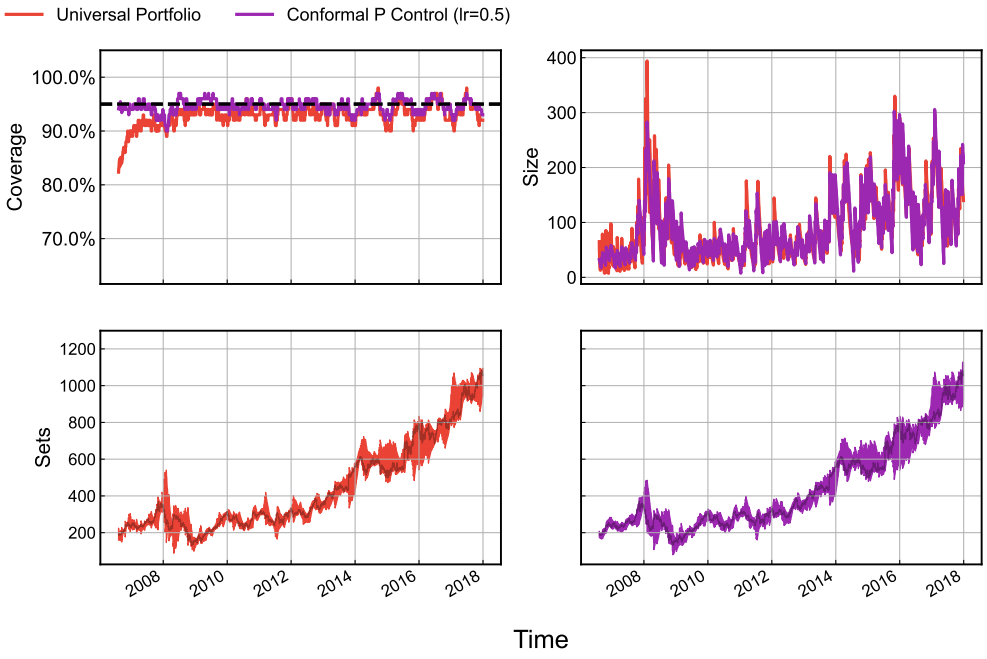

Figure 36: As in Figure 33, UP-OCP vs. P Ctrl (lr=0.5).

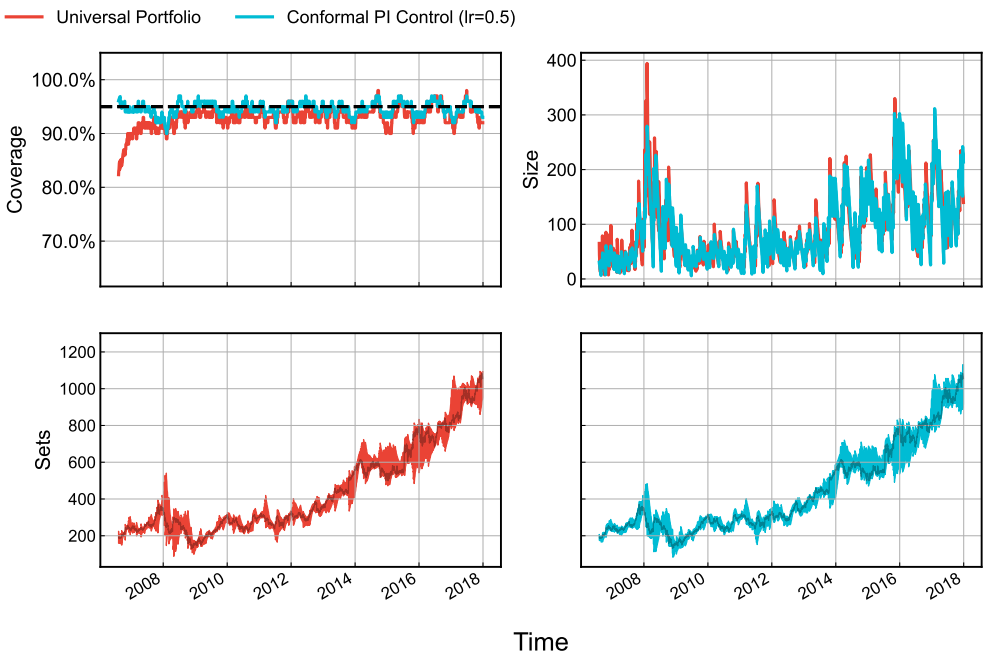

Figure 37: As in Figure 33, UP-OCP vs. PI Ctrl (lr=0.5).

Table 5: Quantitative Comparison on the GOOGL Dataset.

	UP	KT	DtACI	SFOGD (lr=100)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.5)
Marginal coverage	0.932	0.925	0.952	0.948	0.946	0.946
Longest err sequence	**2**	17	5	4	**2**	**2**
Average set size	**86.7**	98.5	∞	95.8	91.8	90.4
Median set size	**67.8**	90.6	79.2	86.1	74.4	72
75% quantile set size	**121**	137	133	132	124	126
90% quantile set size	**173**	169	∞	**171**	176	178
95% quantile set size	204	188	∞	193	207	211

More Pareto Frontiers and Target-level Tracking.

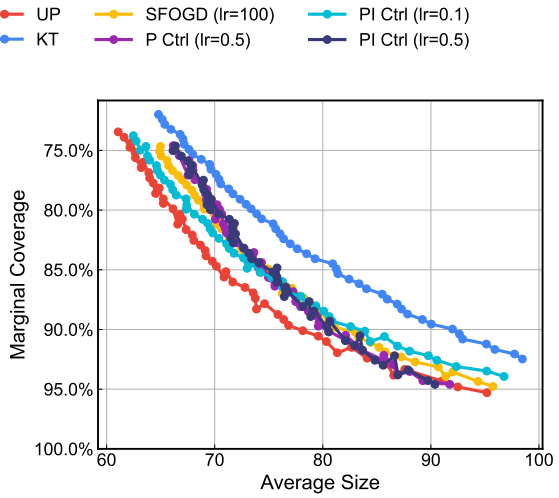

Figure 38: Mean prediction set sizes on GOOGL.

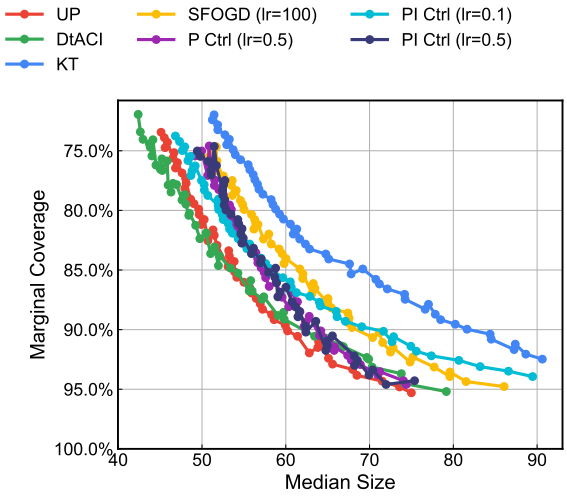

Figure 39: Median prediction set sizes on GOOGL.

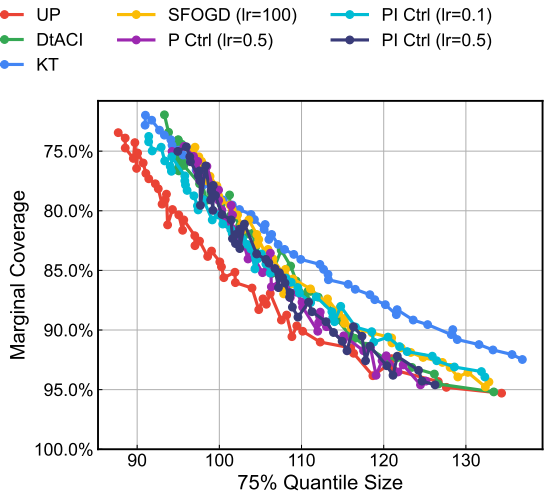

Figure 40: 75% quantile prediction set sizes on GOOGL.

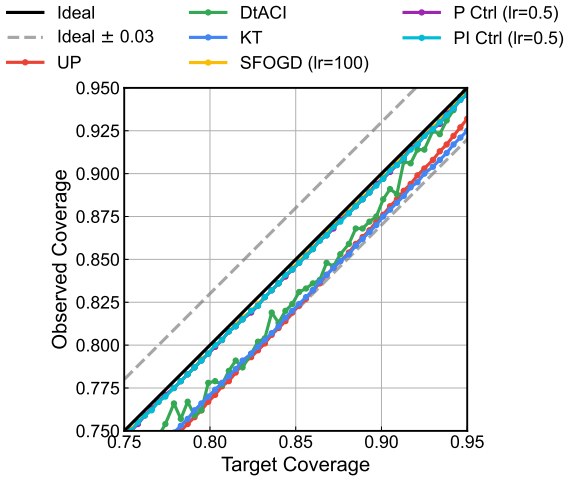

Figure 41: Realized vs. target coverage on GOOGL. Most methods track the diagonal within a small tolerance (± 0.03).

P.4 Electricity Demand Dataset

This data set measures electricity demand in New South Wales collected at half-hour increments from May 7th, 1996 to December 5th, 1998 (we zoom in on the first 2000 time points).

Local Adaptivity.

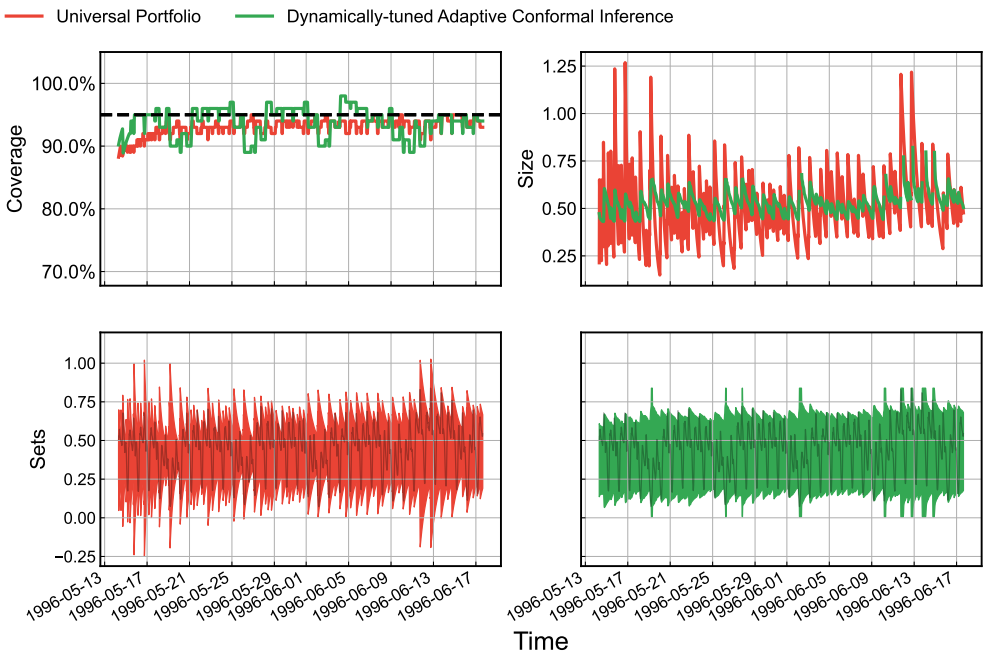

Figure 42: UP-OCP vs. DtACI for forecasting electricity demand.

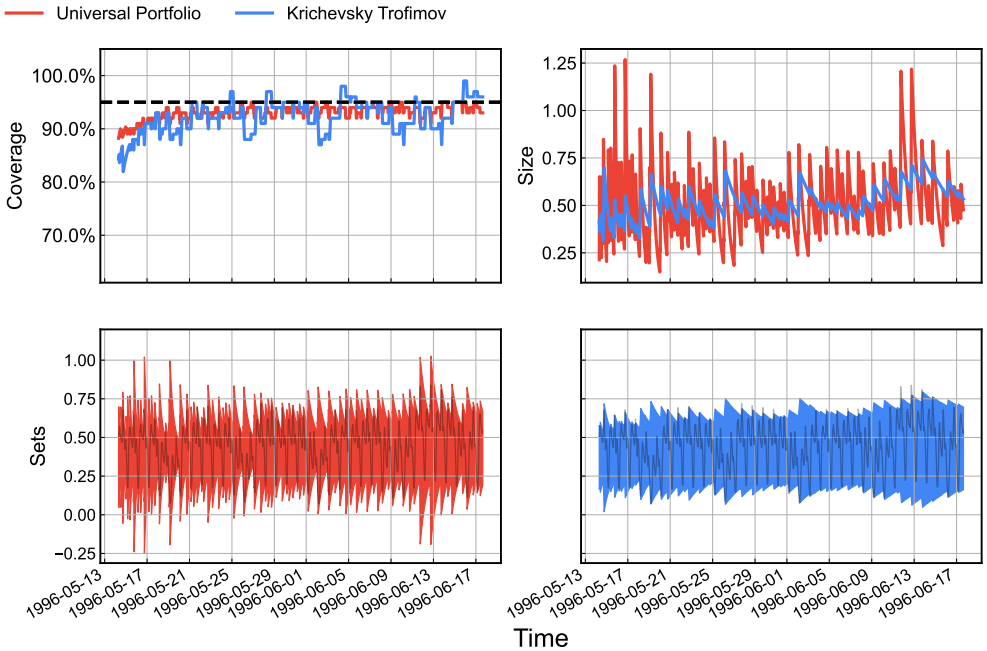

Figure 43: As in Figure 42, UP-OCP vs. KT.

Figure 44: As in Figure 42, UP-OCP vs. SFOGD (lr=1.0).

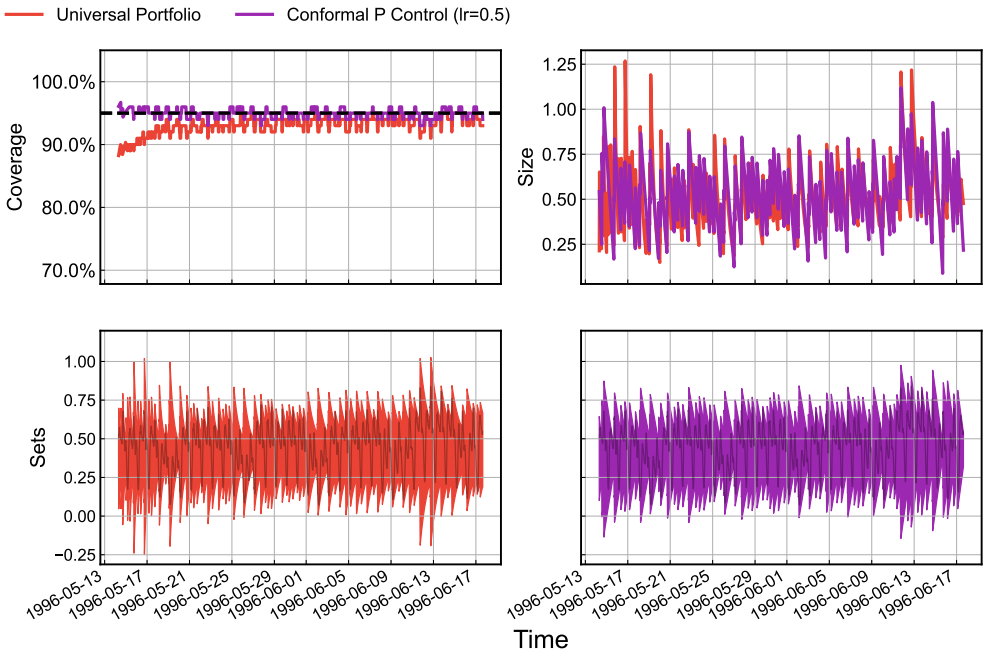

Figure 45: As in Figure 42, UP-OCP vs. P Ctrl (lr=0.5).

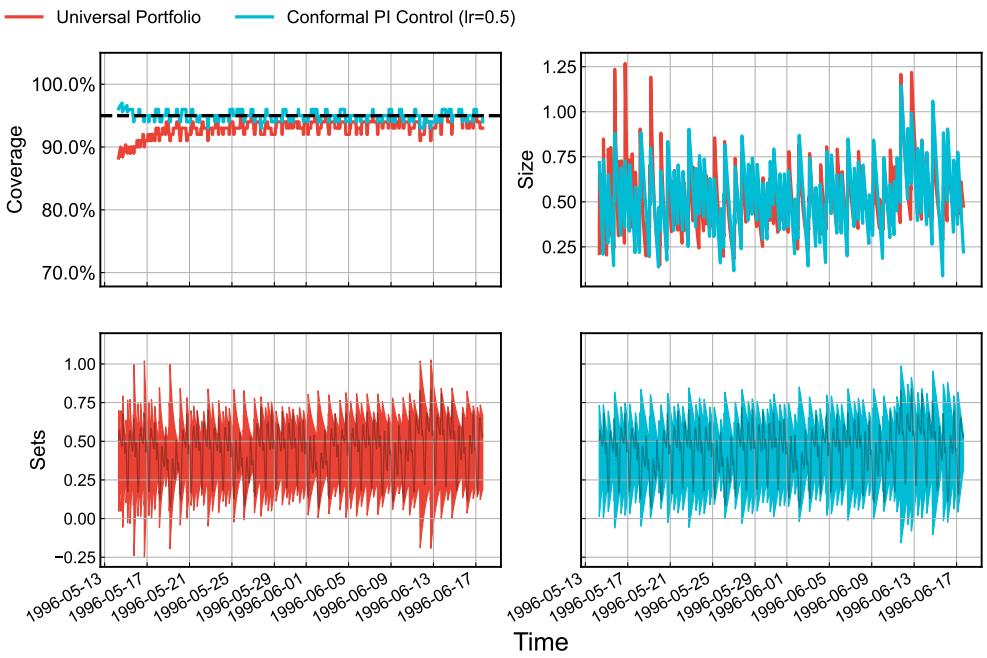

Figure 46: As in Figure 42, UP-OCP vs. PI Ctrl (lr=0.5).

Table 6: Quantitative Comparison on the electricity demand dataset.

	UP	KT	DtACI	SFOGD (lr=1.0)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.5)
Marginal coverage	0.933	0.927	0.939	0.95	0.95	0.95
Longest err sequence	**3**	7	6	**2**	**1**	**2**
Average set size	**0.507**	0.518	∞	0.561	0.528	0.533
Median set size	**0.487**	0.506	0.53	0.552	0.524	0.525
75% quantile set size	0.593	0.572	0.563	0.664	0.637	0.65
90% quantile set size	0.715	0.635	0.619	0.772	0.742	0.748
95% quantile set size	0.793	0.664	0.671	0.862	0.806	0.821

More Pareto Frontiers and Target-level Tracking.

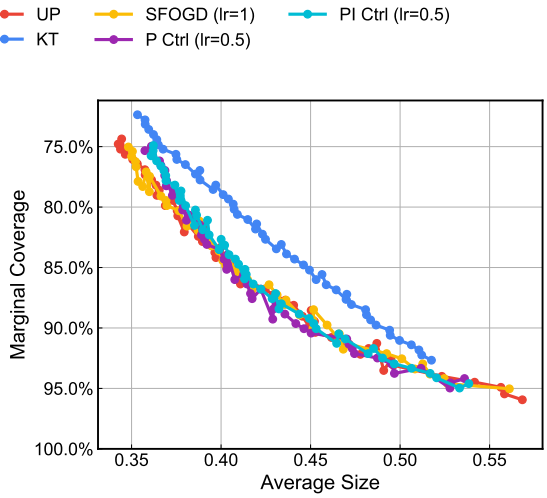

Figure 47: Mean prediction set sizes on electricity demand.

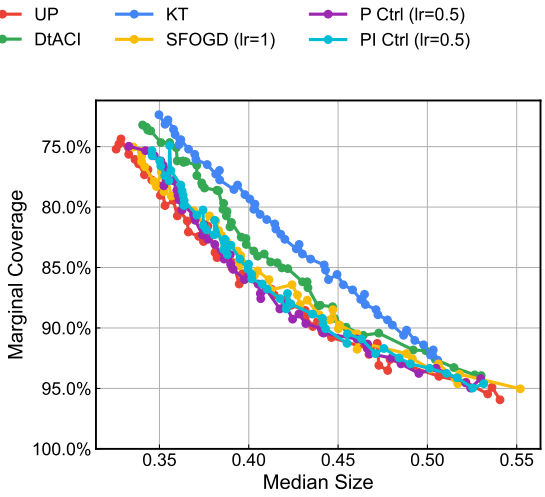

Figure 48: Median prediction set sizes on electricity demand.

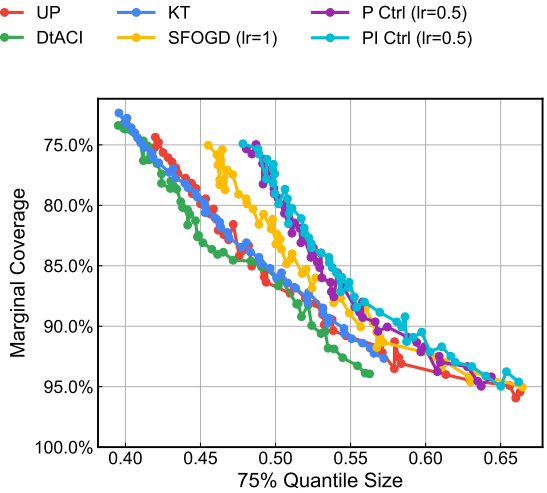

Figure 49: 75% quantile prediction set sizes on electricity demand.

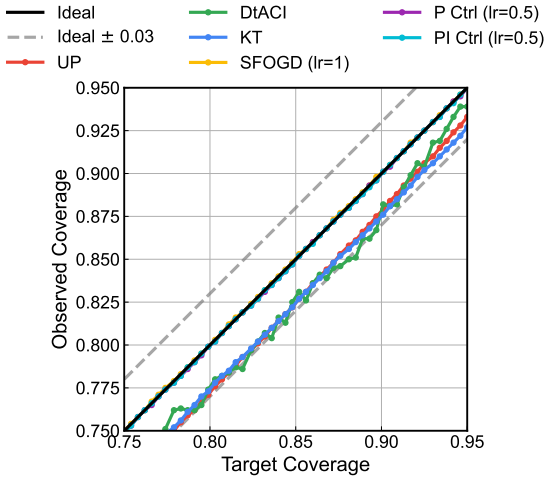

Figure 50: Realized vs. target coverage on electricity demand. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

P.5 Synthetic Sinusoid

We generate the nonconformity scores S_t as a sinusoidal wave distorted by Gaussian noise. Formally, for $t = 1, \ldots, T$,

$$S_t = \max\left(0, \left[\sin\left(\frac{2\pi t}{P}\right) + 0.5\right] S_{\text{mag}} + S_{\min} + \epsilon_t\right),$$

where $\epsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$. We fix the period $P = 200$, the magnitude scaler $S_{\text{mag}} = 10$, and the minimum offset $S_{\min} = 2$. The noise scale is set to $\sigma = 0.3$. The total sequence length is $T = 3000$.

Local Adaptivity.

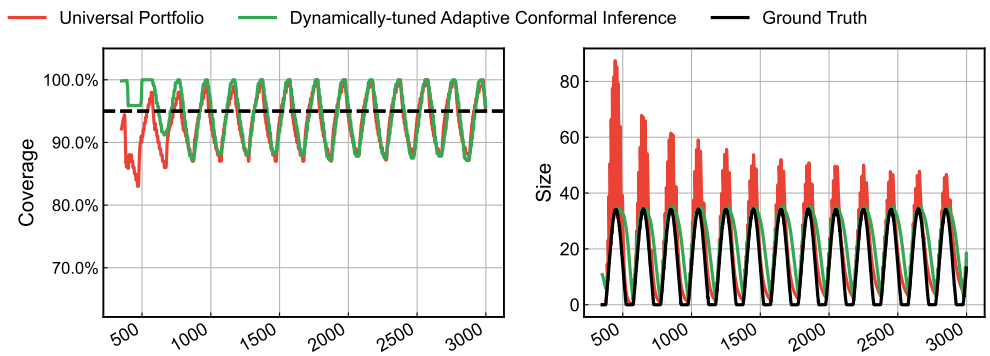

Figure 51: UP-OCP vs. DtACI for forecasting synthetic sinusoid data.

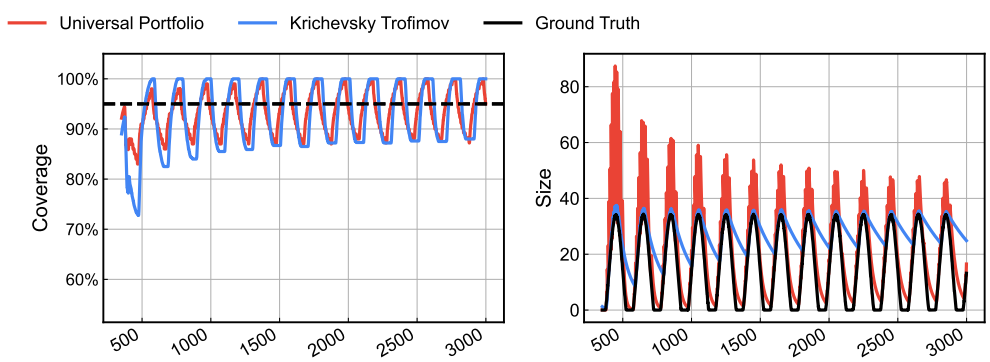

Figure 52: As in Figure 51, UP-OCP vs. KT.

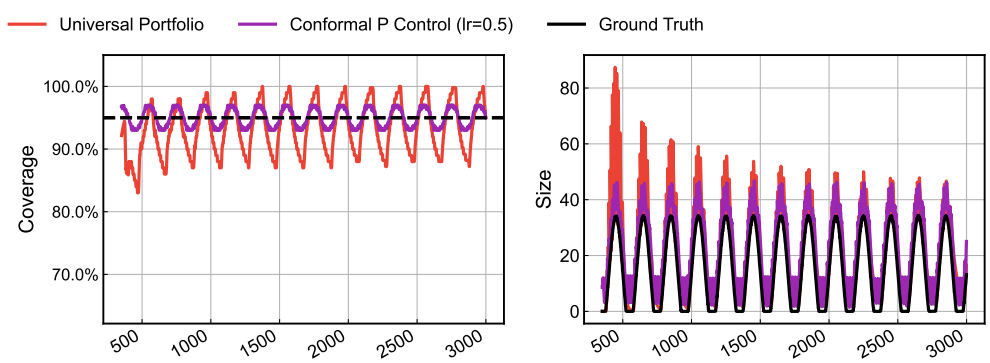

Figure 53: As in Figure 51, UP-OCP vs. P Ctrl (lr=0.5).

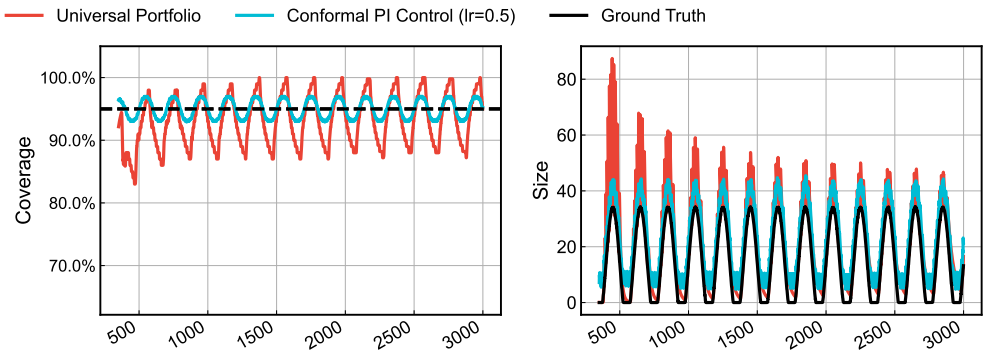

Figure 54: As in Figure 51, UP-OCP vs. PI Ctrl (lr=0.5).

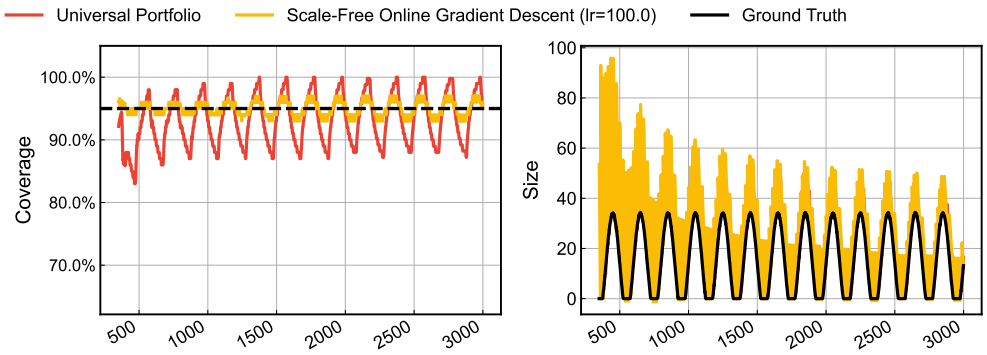

Figure 55: As in Figure 51, UP-OCP vs. SFOGD (lr=100).

Table 7: Quantitative Comparison on the Sinusoid Dataset (synthetic).

	UP	KT	DtACI	SFOGD (lr=100)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.5)
Marginal coverage	0.931	0.928	0.942	0.95	0.95	0.95
Average set size	**21.1**	26.9	∞	28.6	22.8	22.8
Median set size	**17.9**	27.6	26.4	27.7	21.9	21.5
75% quantile set size	34.1	32.1	33.2	40	34.9	34.9
90% quantile set size	43.2	34.5	∞	50.4	41.8	41.9
95% quantile set size	48.3	35.3	∞	57.8	45	45.1

More Pareto Frontiers and Target-level Tracking.

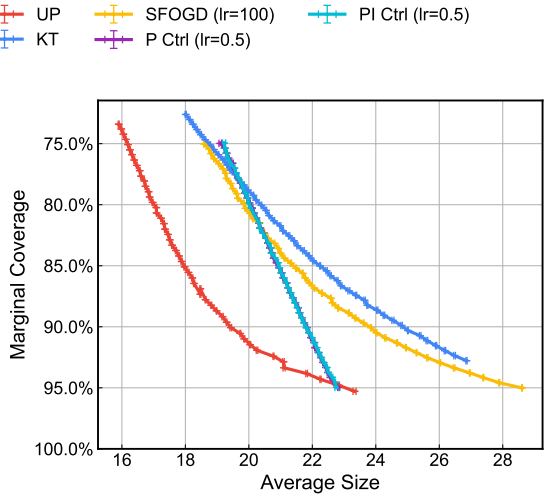

Figure 56: Mean prediction set sizes on synthetic sinusoid data.

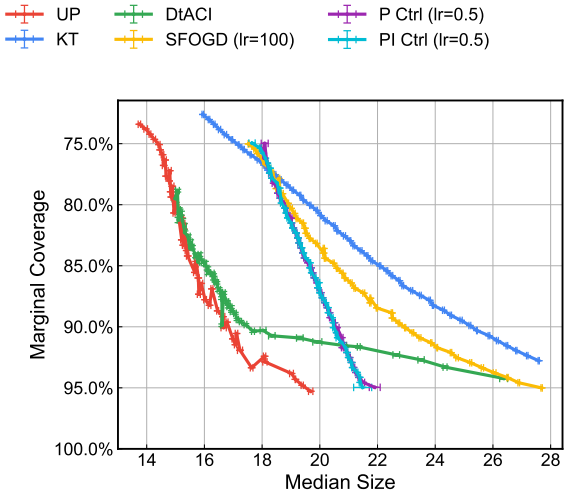

Figure 57: Median prediction set sizes on synthetic sinusoid data.

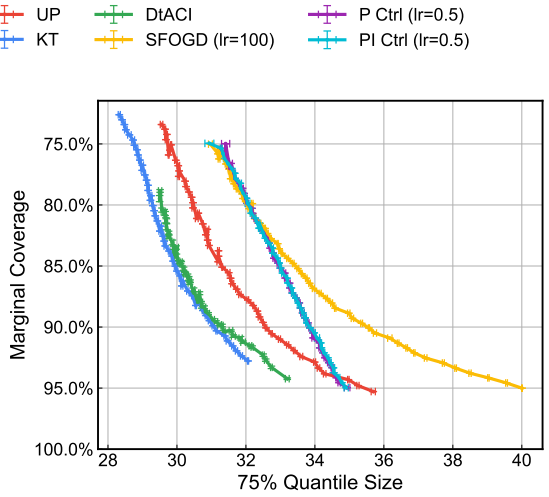

Figure 58: 75% quantile prediction set sizes on synthetic sinusoid data.

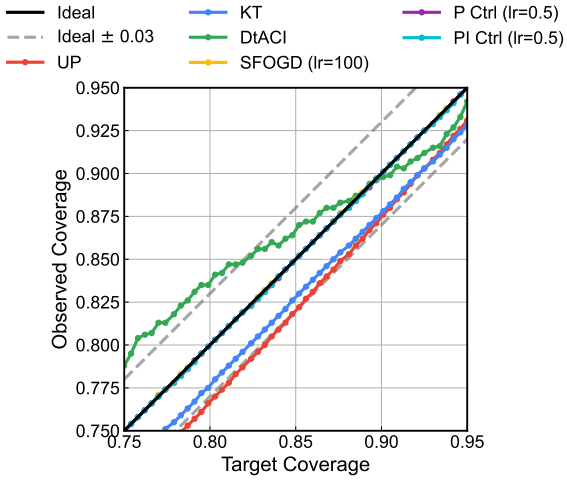

Figure 59: Realized vs. target coverage on synthetic sinusoid data. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

P.6 Stationary Trend with Random waves

Here we try to examine robustness against sparse, heavy-tailed random waves, in comparison with fixed wave positions with periodicity for sinusoid. The nonconformity scores S_t are generated via a process involving a constant baseline, sparse exponential noise, and a rolling window to create wavelet structures.

Formally, for $t = 1, \ldots, T$, we define a constant baseline $C = 10$. We sample a sparsity mask B_t and a noise magnitude E_t as:

$$B_t \sim \text{Bernoulli}(p),$$
$$E_t \sim \text{Exponential}(1/\sigma),$$

where $p = 0.1$ represents the spike probability and $\sigma = 10$ is the scale parameter. We first compute an intermediate score $\tilde{S}_t$ by applying multiplicative noise only when the mask is active:

$$\tilde{S}_t = C(1 + B_t E_t).$$

To simulate locally correlated volatility rather than isolated point outliers, the final score S_t is obtained by applying a centered rolling max-filter of window size $W = 25$:

$$S_t = \max_{\tau \in [t - \lfloor W/2 \rfloor, t + \lfloor W/2 \rfloor]} \tilde{S}_\tau.$$

The total sequence length is $T = 3000$.

Local Adaptivity.

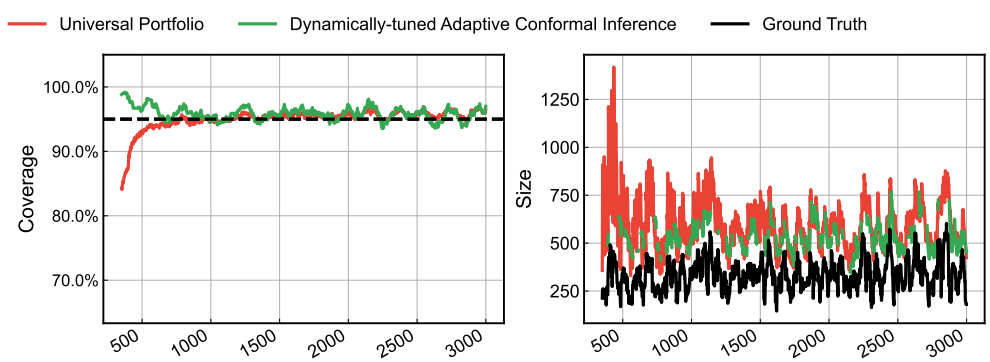

Figure 60: UP-OCP vs. DtACI for forecasting stationary synthetic data with random waves.

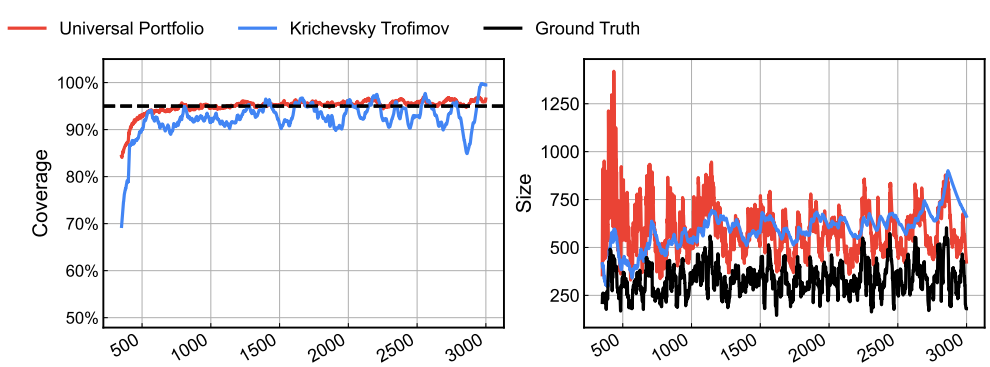

Figure 61: As in Figure 60, UP-OCP vs. KT.

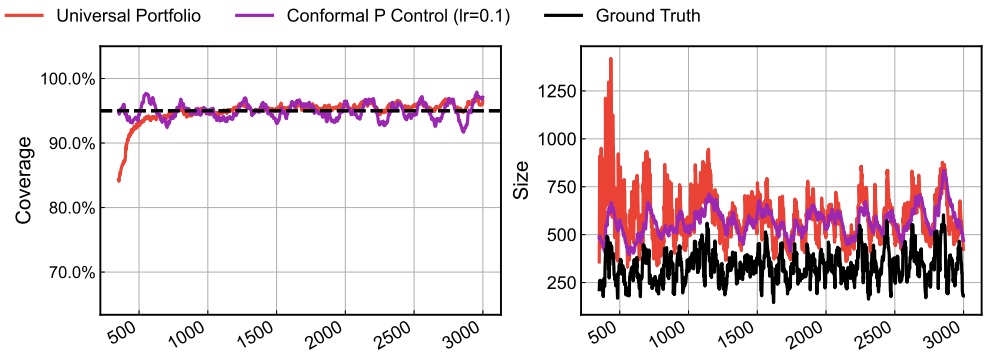

Figure 62: As in Figure 60, UP-OCP vs. P Ctrl (lr=0.1).

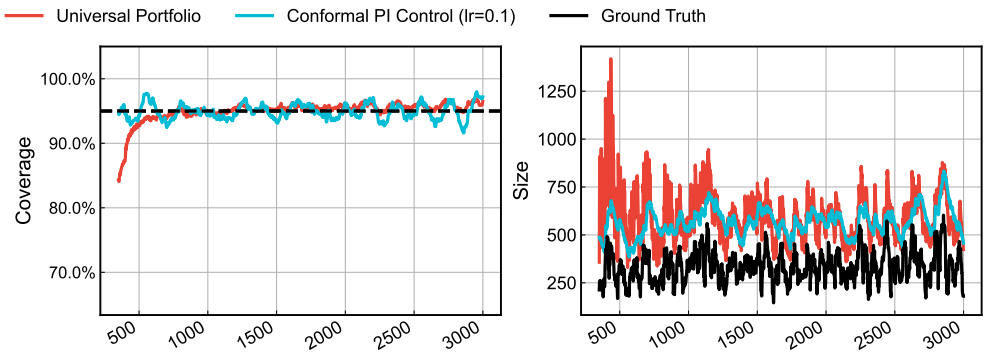

Figure 63: As in Figure 60, UP-OCP vs. PI Ctrl (lr=0.1).

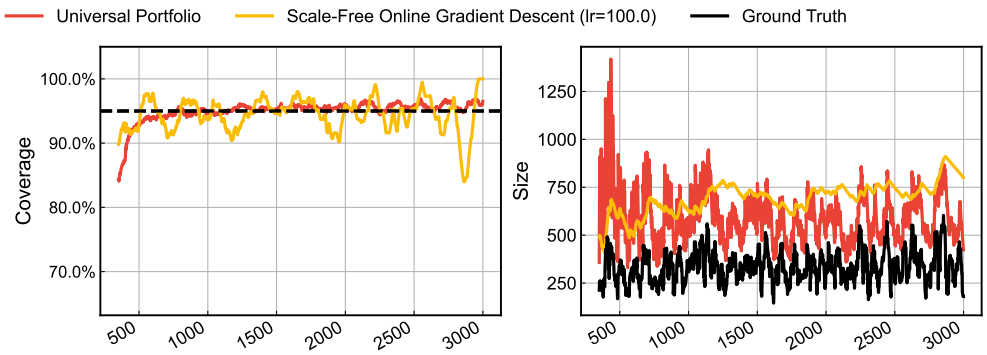

Figure 64: As in Figure 60, UP-OCP vs. SFOGD (lr=100).

Table 8: Quantitative Comparison on the Stationary Dataset (synthetic).

	UP	KT	DtACI	SFOGD (lr=100)	P Ctrl (lr=0.1)	PI Ctrl (lr=0.1)
Marginal coverage	**0.952**	0.93	0.958	0.946	0.949	0.949
Average set size	592	596	∞	694	566	567
Median set size	**500**	574	494	691	531	534
75% quantile set size	746	724	665	841	692	694
90% quantile set size	1070	892	∞	962	882	880
95% quantile set size	1320	994	∞	1050	1000	1010

More Pareto Frontiers and Target-level Tracking.

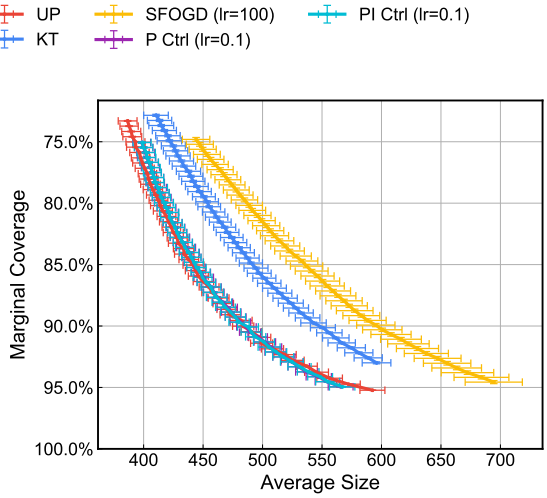

Figure 65: Mean prediction set sizes on synthetic stationary data with random waves.

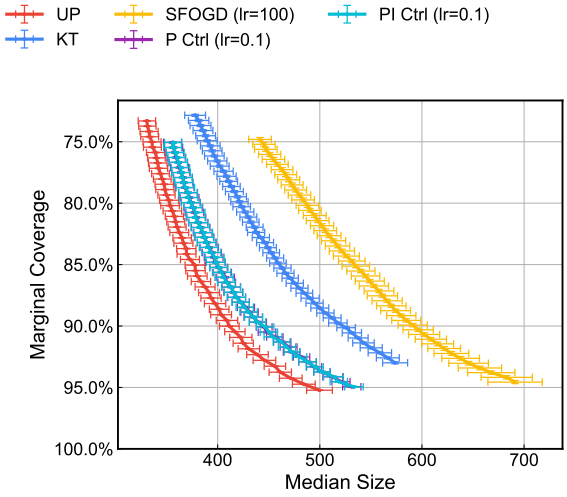

Figure 66: Median prediction set sizes on synthetic stationary data with random waves.

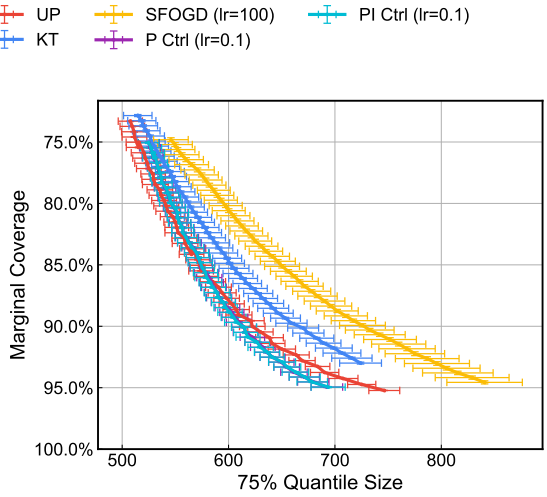

Figure 67: 75% quantile prediction set sizes on synthetic stationary data with random waves.

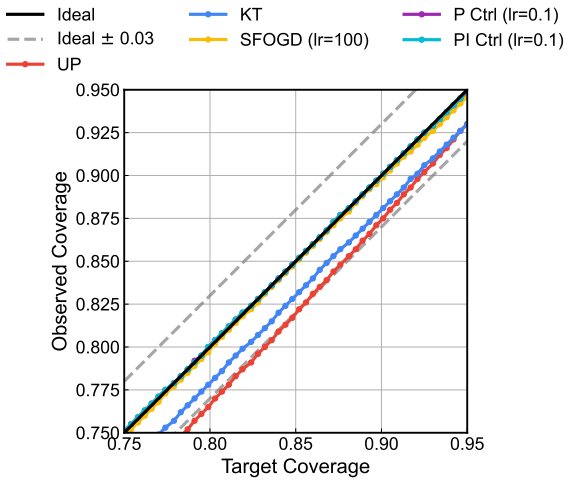

Figure 68: Realized vs. target coverage on synthetic stationary data with random waves. Most methods track the diagonal within a small tolerance (± 0.03).

P.7 Quadratic Trend with Random Waves

At last, we evaluate the algorithms on quadratic drift, which combines the sparse noise structure of the stationary regime with a non-stationary, monotonically increasing trend. This tests the ability of the algorithms to track a drifting baseline while remaining robust to random waves.

Formally, the underlying trend T_t follows a quadratic trajectory starting at 0 and ending at 20 over $T = 3000$ steps:

$$T_t = \frac{20}{T^2} t^2. \tag{45}$$

The noise generation follows the same multiplicative, sparse structure as the stationary wavelet experiment. We sample a Bernoulli mask $B_t \sim \text{Bernoulli}(0.1)$ and exponential noise $E_t \sim \text{Exponential}(1/10)$. The raw score $\tilde{S}_t$ applies this noise to the drifting baseline:

$$\tilde{S}_t = T_t(1 + B_t E_t). \tag{46}$$

Finally, the observed score S_t is the result of a rolling max-pooling operation with window size $W = 25$, creating locally correlated volatility structures around the drift.

Local Adaptivity.

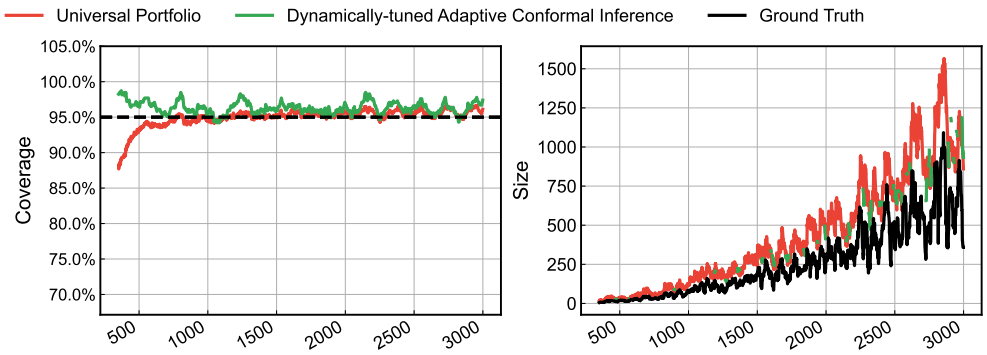

Figure 69: UP-OCP vs. DtACI for forecasting synthetic data with quadratic trend and random waves.

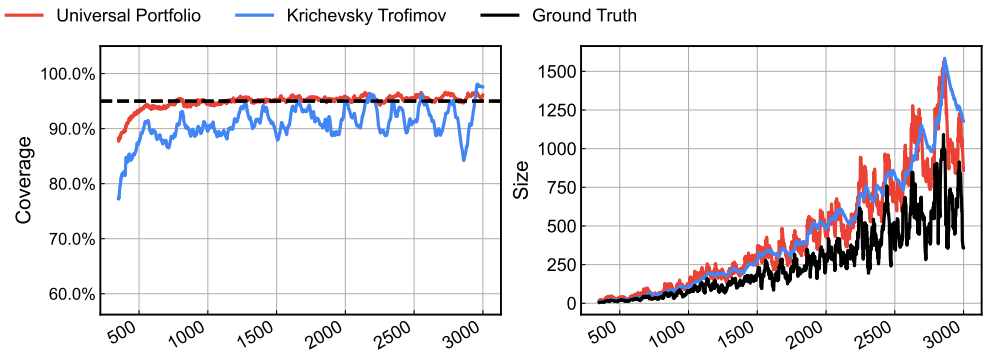

Figure 70: As in Figure 69, UP-OCP vs. KT.

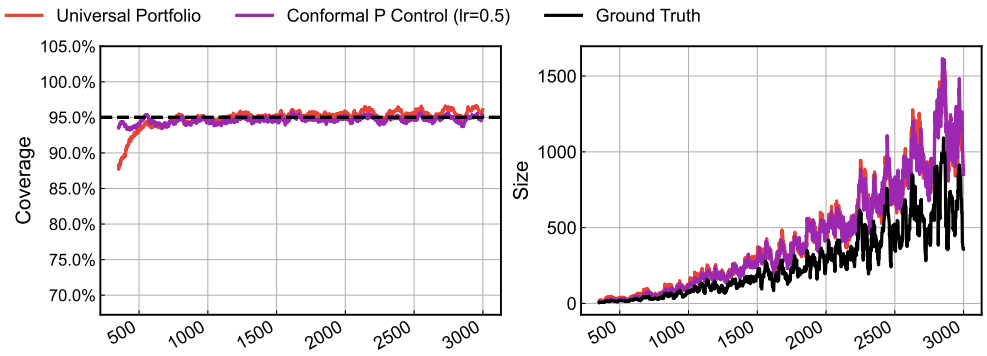

Figure 71: As in Figure 69, UP-OCP vs. P Ctrl (lr=0.5).

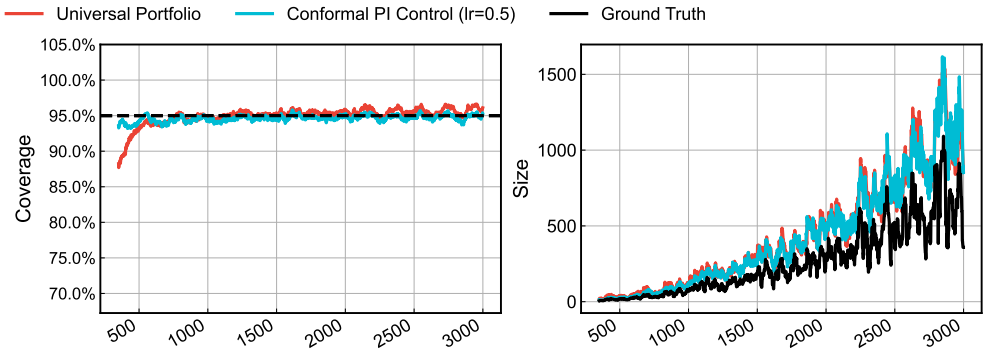

Figure 72: As in Figure 69, UP-OCP vs. PI Ctrl (lr=0.5).

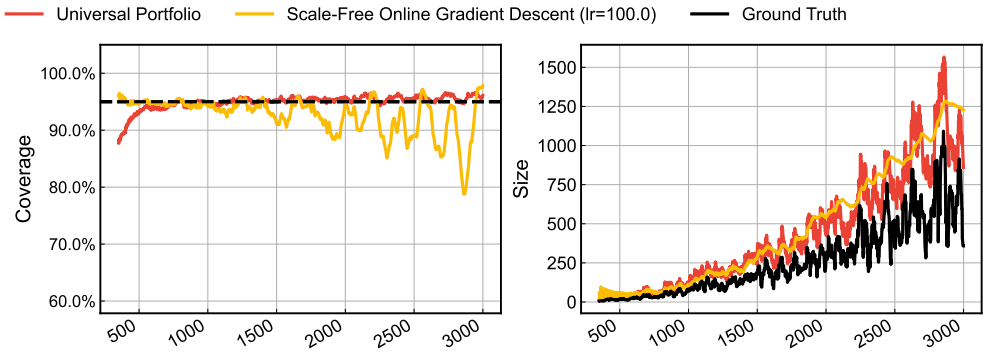

Figure 73: As in Figure 69, UP-OCP vs. SFOGD (lr=100).

Table 9: Quantitative Comparison on the Mix Dataset (s synthetic).

	UP	KT	DtACI	SFOGD (lr=100)	P Ctrl (lr=0.5)	PI Ctrl (lr=0.5)
Marginal coverage	**0.951**	0.913	0.963	0.927	0.947	0.947
Average set size	**433**	451	∞	467	**429**	**429**
Median set size	**292**	308	406	324	**285**	**285**
75% quantile set size	**624**	692	∞	790	**626**	**626**
90% quantile set size	**1020**	1040	∞	1090	**1020**	**1020**
95% quantile set size	1320	1360	∞	1260	1310	1310

More Pareto Frontiers and Target-level Tracking.

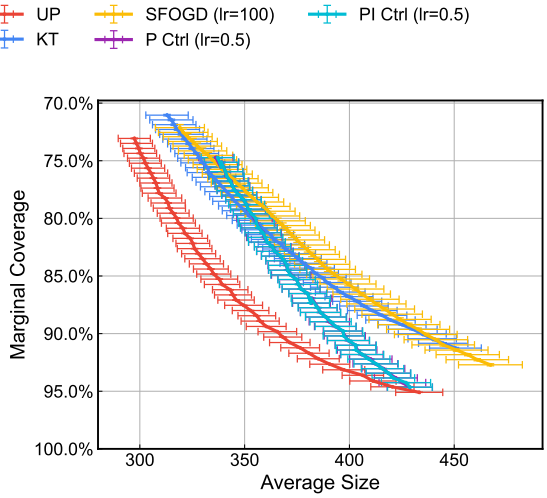

Figure 74: Mean prediction set sizes on synthetic data with quadratic trend and random waves.

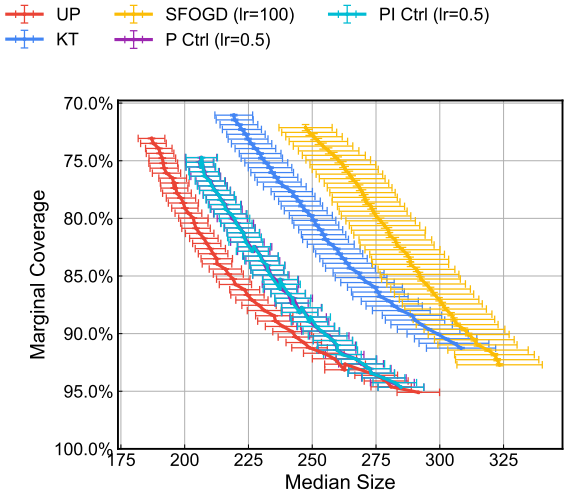

Figure 75: Median prediction set sizes on synthetic data with quadratic trend and random waves.

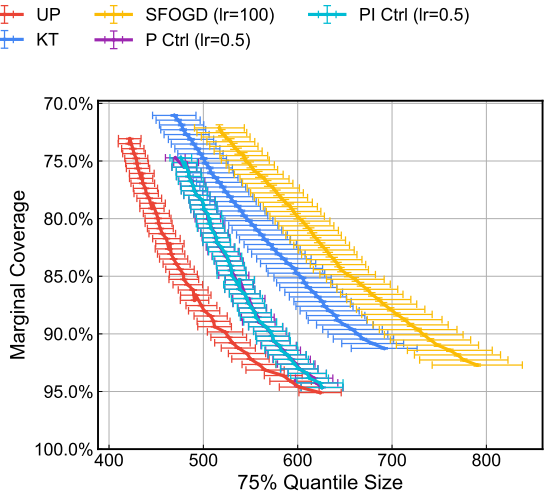

Figure 76: 75% quantile prediction set sizes on synthetic data with quadratic trend and random waves.

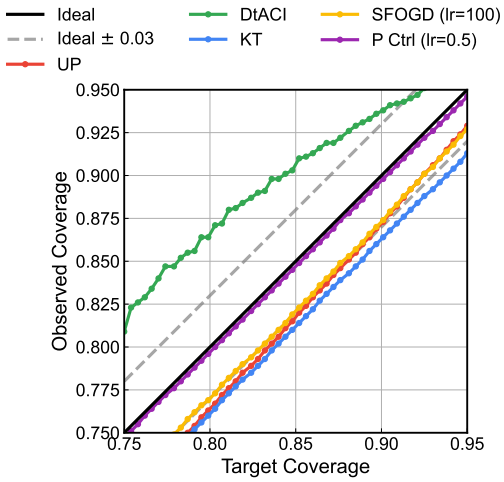

Figure 77: Realized vs. target coverage on synthetic data with quadratic trend and random waves. Most methods track the diagonal within a small tolerance ($\pm$ 0.03).

Q Sensitivity of Parameterized Baselines to Hyperparameters

In this section, we empirically demonstrate the sensitivity of parameterized OCP methods (SF-OGD, P/PI Control) to hyperparameter choices. While tuned baselines can achieve competitive performance (as shown in Section 5), selecting these parameters requires an oracle or grid search that is not feasible in a true online setting. We give three such examples of failure below. These examples underscore that parameterized methods can not be naively plugged in; they require careful tuning. UP-OCP avoids these divergence modes by design without requiring manual tuning.

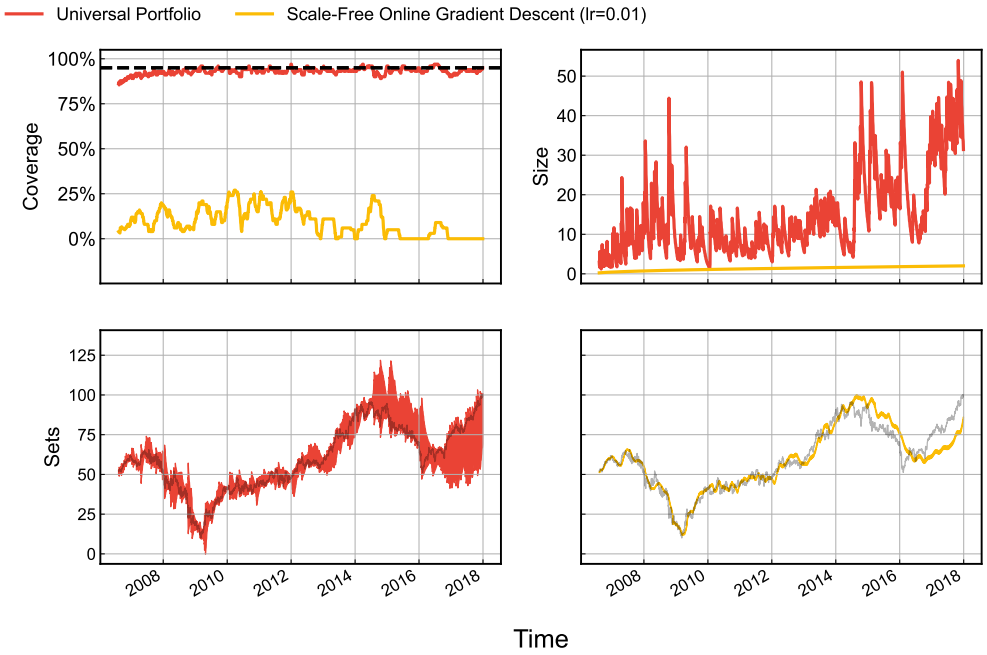

Figure 78: UP-OCP vs. SF-OGD (lr=0.01) on AXP dataset; SF-OGD (lr=0.01)fails to expand the prediction sets sufficiently, resulting in intervals that are consistently too narrow (right panel). The marginal coverage (yellow, left panel) collapses to nearly 0%, far below the 95% target.

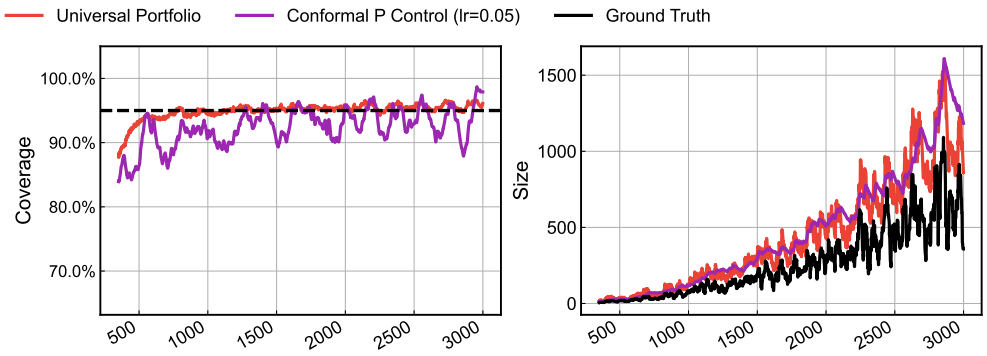

Figure 79: UP-OCP vs. P Ctrl (lr=0.05) on synthetic data with quadratic trend and random waves; While the parameter-free UP-OCP (red) maintains stable coverage near the target, the P Controller (purple) exhibits significant oscillation. Intuitively this indicates that the controller is over-reacting to single data points.

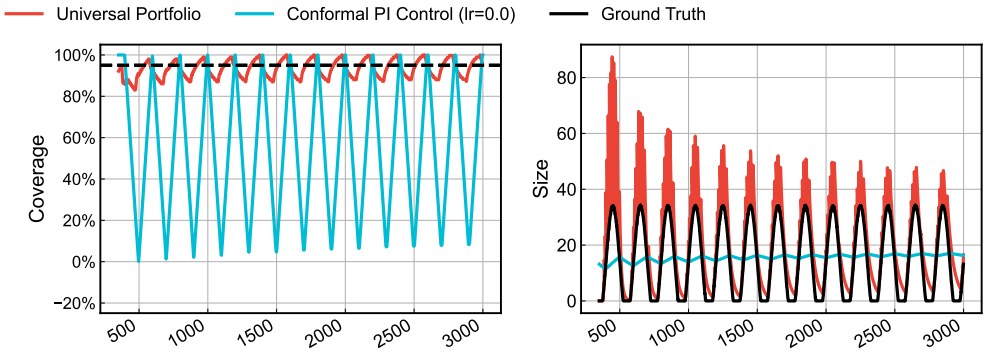

Figure 80: UP-OCP vs. PI Ctrl (lr=0.0) on synthetic sinusoid data; PI Controller fails to adapt to periodicity of the ground truth (black). The prediction set sizes (cyan, right panel) remain effectively constant. The coverage (left panel) oscillates deterministically between 0% and 100% as the ground truth noise wave passes in and out. In contrast, UP-OCP (red) correctly modulates the interval width to track the sinusoidal pattern, maintaining valid coverage.