

Conformal Safety Monitoring for Flight Testing: A Case Study in Data-Driven Safety Learning

Aaron O. Feldman¹, D. Isaiah Harp², Joseph Duncan³, Mac Schwager¹, *Senior Member, IEEE*

Abstract—We develop a data-driven approach for runtime safety monitoring in flight testing, where pilots perform maneuvers on aircraft with uncertain parameters. Because safety violations can arise unexpectedly as a result of these uncertainties, pilots need clear, preemptive criteria to abort the maneuver in advance of safety violation. To solve this problem, we use offline stochastic trajectory simulation to learn a calibrated statistical model of the short-term safety risk facing pilots. We use flight testing as a motivating example for data-driven learning/monitoring of safety due to its inherent safety risk, uncertainty, and human-interaction. However, our approach consists of three broadly-applicable components: a model to predict future state from recent observations, a nearest neighbor model to classify the safety of the predicted state, and classifier calibration via conformal prediction. We evaluate our method on a flight dynamics model with uncertain parameters, demonstrating its ability to reliably identify unsafe scenarios, match theoretical guarantees, and outperform baseline approaches in preemptive classification of risk.

I. INTRODUCTION

Safety in human-in-the-loop robotic systems often depends on uncertain dynamics, implicit constraints, and the need for preemptive intervention, factors that make formal specification of safety requirements difficult or incomplete. In this work, we develop a data-driven approach for runtime safety monitoring in flight testing, using conformal prediction to provide statistically calibrated, data-driven safety alerts to test pilots. Our goal is to learn, from prior flight data, a quantitative measure of short-term risk, enabling the system to preemptively signal when a pilot should abort a maneuver to ensure safety. Flight testing presents a compelling case study for reasoning about intangible safety constraints that emerge from uncertainty, human interaction, and implicit domain knowledge.

Our motivation is grounded in three key challenges. Firstly, flight testing is safety-critical, necessitating principled mechanisms for reasoning about safety. In flight testing, pilots perform aggressive maneuvers on aircraft with partially known dynamics to refine model parameters. These test flights are inherently risky due to the uncertain aircraft behavior, especially near performance limits. Secondly, due to human-machine interaction in flight testing, our monitor must anticipate future safety violation. To be actionable, alerts must allow the human pilot time to react i.e., to abort the maneuver. The need to preemptively alert makes even a well-defined safety

specification (e.g., a limit on lateral acceleration) potentially complex/ambiguous; we must reason about the potential of the current state to reach an unsafe future state based on the system dynamics. Thirdly, flight testing requires stochastic analysis of the implicit/future safety constraints. Because the aircraft parameters are uncertain and unknown during the maneuver, we cannot exactly propagate the system dynamics to predict the future state to reason about future safety. Instead, we must reason statistically.

To address these challenges, we propose using simulated flight rollouts to model and calibrate a statistical runtime safety monitor offline. Our method consists of three stages: future state prediction, safety classification for this prediction, and conformal calibration, discussed further in Section V.

At runtime, our safety monitor operates without requiring forward simulation or explicit evaluation of abstract safety specifications, making it computationally tractable. Using current observations, the predictive model and classifier are queried to obtain a conformal p -value, a calibrated scalar measure of risk, enabling both binary alerting and continuous safety scoring. By construction, our approach ensures that the probability of failing to preemptively alert before a true safety violation is at most a user-specified ϵ , offering a statistical guarantee that is easy to interpret for the human operator/pilot.

This work thus illustrates an approach to provide data-driven operational safety guarantees in settings like flight testing, where uncertain dynamics and human-machine interaction can make traditional safety constraint specification challenging.

II. LITERATURE REVIEW

Some related work in the field of flight testing includes [1] and [2]. [1] fits a model to characterize plane parameters (e.g., lift, drag, and moment coefficient) across a variety of aircraft. This work demonstrates a procedure for data-driven aircraft parameter estimation and uncertainty quantification rather than tackling the problem of runtime safety monitoring. In fact, the resulting parameter bounds and polynomial models from [1] could be used for stochastic simulation within our approach as a next step. [2] presents several approaches for real-time runtime safety monitoring of flight tests. One of their approaches is to determine a safe operating domain based on trajectories flown in simulation with some plane parameter randomization. They also consider an online approach for recursively estimating the aircraft stability/control derivatives and abort if these fall outside user-defined bounds. In contrast, our approach implicitly defines a safe operating domain using a nearest neighbor classifier and, by predicting future states

¹Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, USA, {aofeldma, schwager}@stanford.edu. ² Department of Aeronautics, United States Air Force Academy, Air Force Academy, CO 80840, USA, daniel.harp@afacademy.af.edu. ³ United States Air Force, Edwards AFB, CA 93523, USA, joseph.duncan.2@us.af.mil.

using a buffer of recent data, implicitly conditions on aircraft parameters without explicitly regressing them.

Outside of flight testing, conformal prediction has been widely used for trajectory prediction in robotics (e.g., for pedestrians) [3, 4, 5, 6, 7, 8], sensor anomaly detection [9, 10, 11, 12, 13, 14], and dynamics model error [15].

The work most closely related to this one is our own prior research [16] which used conformal prediction to learn a set of unsafe states from human feedback. This project builds on this work in two regards. Firstly, we apply the theory of [16] to the novel and practical use case of flight testing. Secondly, we combine the nearest neighbor model with a predictive model to predict future outputs. We show that this predictive model is key to the success of our approach and outperforms other transformations like PCA that were used in [16].

III. OVERVIEW OF CONFORMAL PREDICTION

Conformal prediction [17, 18] is a statistical approach for uncertainty quantification that operates without distributional assumptions. The core result is that for $N+1$ random variables (scores) s_1, \dots, s_N, s_{N+1} exchangeable, equally likely under re-ordering,

$$\Pr(s_{N+1} \leq s_{(k)}) \geq \frac{k}{N+1} \quad (1)$$

with equality assuming no ties. Thus, for a user-specified ϵ , taking $k(\epsilon) = \lceil (N+1)(1-\epsilon) \rceil$ will ensure that

$$\Pr(s_{N+1} \leq s_{(k)}) \geq 1 - \epsilon \quad (2)$$

i.e., we will fail to bound s_{N+1} with a miss rate of at most ϵ . Here, $s_{(k)}$ refers to the k -th order statistic of s_1, \dots, s_N . Broadly, there are two categories of conformal prediction using these results: split conformal prediction [19] and full conformal prediction [20]. We build on [16], which used full conformal prediction within a nearest neighbor model. Using conformal prediction, we can assign to the nearest neighbor model's output a statistical interpretation and determine when to alert to achieve a user-specified miss rate ϵ .

IV. PROBLEM SETTING

We assume a distribution over unknown system (plane) parameters $\theta \sim D_\theta$ from which we can repeatedly sample $\theta_i \sim D_\theta$ in simulation to produce resulting trajectories τ_i . The true, unknown, plane parameters are viewed as a new draw $\theta^* \sim D_\theta$ and induce a deterministic test trajectory τ^* . Our goal is to develop an early warning system such that if τ^* will become unsafe, we will alert the pilot with t_{early} advanced notice (we use $t_{\text{early}} = 0.25$ [sec]), leaving time for the human to take corrective action.

As an illustrative example for our experiments, we consider a continuous, linear time invariant flight dynamics model, provided to us by members of the USAF test pilot school.

$$\dot{x} = Ax + Bu, y = Cx + Du. \quad (3)$$

The three-dimensional plane state $x = (\beta, p, r)$ consists of the sideslip angle ([rad]), roll rate ([rad/s]), and yaw rate ([rad/s]) respectively. The control action $u = (\delta_a, \delta_r)$ consists of the aileron and rudder deflection ([rad]). The

output $y = (\beta, p, r, N_y, \delta_a, \delta_r)$ stores the state, action, and additionally N_y the lateral acceleration ([g]). The matrices $\theta = (A, B, C, D)$ are the unknown system parameters in our setting and are randomized over a known range (e.g., based on prior flight data). On top of the open-loop transfer function $G(s) = C(sI - A)^{-1}B + D$, a feedback controller $K(s)$ is modeled to translate pilot inputs into control actions, and we simulate rollouts using the closed-loop transfer function. We consider a simple flight maneuver referred to as a rudder doublet wherein the pilot applies as input to the closed-loop system $\delta_r = 1$ followed by $\delta_r = -1$ for one second each.

We consider that a “high-level” abstract safety specification can be queried to evaluate each rollout’s safety in hindsight offline. For instance, this might involve performing a computationally expensive structural/load analysis at different times during the maneuver or an evaluation of the changing gain/phase margin for a nonlinear dynamical system. This safety analysis could even come from simulated pilot intervention/interruption in hindsight [16]. For preliminary analysis in this work, a rollout is defined to become unsafe if the lateral acceleration grows too large: $|N_y| \geq 0.5$. Even in this case where the specification is well-defined at the current time, preemptively reasoning about violation remains challenging due to uncertainty in the true aircraft parameters.

V. APPROACH

A. Data Collection

Offline, the user specifies a number N of unsafe trajectories to collect, and we repeatedly sample system parameters θ_i and perform rollouts τ_i until we have obtained N unsafe trajectories D_u . In this process, we also obtain a variable number, say M , of safe trajectories D_s .

Each raw trajectory consists of a sequence of outputs $\tau = (y_1, \dots, y_T)$. To preemptively predict system failure, reasoning about the single, current output y_t is typically insufficient to anticipate future behavior. To form a better feature we concatenate a short buffer of the recent outputs $o_t = (y_{t-k}, \dots, y_t)$ as our observation at each time.

Additionally, for each unsafe trajectory $\tau_i \in D_u$, we record the time of failure t_i and go back in time t_{early} to obtain the observation $o_i := o_{t_i - t_{\text{early}}}$. We collect the observations o_i , $i \in \{1, \dots, N\}$, taken t_{early} in advance of system failure, into \mathcal{O}_u , which we refer to as the error observation set. For each safe trajectory $\tau_i \in D_s$, we extract observations, possibly randomly subselecting a few times from each trajectory, to construct a similar set \mathcal{O}_s of safe observations. We then offline fit and calibrate a classifier to distinguish between \mathcal{O}_u and \mathcal{O}_s .

Our approach consists of three components:

- State Prediction: A model is trained to forecast future system outputs from a short history of observed states.
- Safety Classification: A nearest-neighbor classifier identifies if the predicted future state is likely safe or unsafe.
- Conformal Calibration: Using conformal prediction, we calibrate the classifier’s alert threshold to guarantee that the miss rate does not exceed a user-defined bound.

B. State Prediction

We use a linear model to predict the state t_{early} into the future from the current observation:

$$\hat{y}_{t+t_{early}} = \phi(o_t) = Mo_t + \mu \quad (4)$$

where M, μ are learned parameters of the linear model ϕ . We fit this model using least squares to observations from the safe trajectories $D = \{(o_t, y_{t+t_{early}})\}$ using the future state $y_{t+t_{early}}$ as the regression target to predict i.e., we solve

$$\min_{M, \mu} \sum_{(o_t, y_{t+t_{early}}) \in D} \|y_{t+t_{early}} - (Mo_t + \mu)\|_2^2. \quad (5)$$

Using the resulting linear model, we transform the observations in $\mathcal{O}_u, \mathcal{O}_s$ to their predicted future states $\hat{y}_{t+t_{early}}$, obtaining corresponding sets $\mathcal{Y}_u = \phi(\mathcal{O}_u), \mathcal{Y}_s = \phi(\mathcal{O}_s)$.

We could instead directly apply nearest neighbor classification to distinguish between $\mathcal{O}_u, \mathcal{O}_s$. However, by first transforming to the predicted future states and classifying in the transformed space of $\mathcal{Y}_u, \mathcal{Y}_s$ we can improve the classifier performance, as shown in our experiments. The linear model can be viewed as learning a particularly useful representation for downstream classification.

C. Safety Classification

After transforming to the predicted future state $\hat{y}_{t+t_{early}}$ we could simply check the safety specification for it. However, this may be impossible at runtime if the safety specification requires significant computation, privileged information, or expert/pilot labeling. Even if we can check the safety specification at runtime, this would be imperfect as we have only a predicted, not true, future state.

Therefore, we further fit a nearest neighbor classifier to distinguish \mathcal{Y}_u from \mathcal{Y}_s . We use the conformal score from [16] which scores a test point y by its distance to the nearest unsafe versus safe point in the data:

$$s(y) = \min_{y_u \in \mathcal{Y}_u} \|y - y_u\|_2^2 - \min_{y_s \in \mathcal{Y}_s} \|y - y_s\|_2^2. \quad (6)$$

Thus, we expect new unsafe points to have low scores.

D. Conformal Prediction Calibration

Lastly, we calibrate our nearest neighbor classifier using conformal prediction. Given a new test point y , we would like to give a statistical interpretation to the score $s(y)$ output from the nearest neighbor model. Practically, we need to determine a threshold at which $s(y)$ is declared too low so that our warning system should issue an alert. We set this alert threshold to ensure a user-specified miss rate ϵ i.e., a requirement that the warning system miss preemptively flagging (t_{early} in advance) in only ϵ fraction of trajectories which became unsafe.

[16] showed that we could calibrate the nearest neighbor classifier by simply recording the intra-dataset nearest neighbor distances, which for unsafe observation $y_u^i \in \mathcal{Y}_u$ is given by

$$\alpha_i = \min_{y'_u \in \mathcal{Y}_u - \{y_u^i\}} - \min_{y_s \in \mathcal{Y}_s} \|y_u^i - y_s\|_2^2. \quad (7)$$

Notably, this allows us to calibrate offline, reusing \mathcal{Y}_u without need for a separate validation/calibration dataset. To achieve a miss rate at most ϵ , the conformal score threshold s^* is

$$s^* = \alpha_{(k)}, k = \lceil (N+1)(1-\epsilon) \rceil. \quad (8)$$

where $\alpha_{(k)}$ refers to the k -th order statistic of $\alpha_1, \dots, \alpha_N$. If at test time $s(y) \leq s^*$, we issue an alert.

Beyond a single threshold dictating when we should alert, we can obtain the conformal p -value associated with $s(y)$ as a preemptive measure of the risk facing the pilot. For given $s(y)$, the associated p -value ϵ^* is defined as the smallest ϵ for which we would not alert based on Eq. 8¹. Practically, the p -value ranges from $[0, 1]$ with lower values deemed safer, and alerting whenever $\epsilon^* \geq \epsilon$ is equivalent to alerting whenever $s(y) \leq s^*$. Thus, the pilot can simply monitor the p -value during flight and abort if it exceeds designated miss rate ϵ .

E. Runtime Safety Monitoring

During deployment, we perform the following at each time:

- 1) Form observation $o_t = (y_{t-k}, \dots, y_t)$ using latest outputs.
- 2) Predict future state $\hat{y}_{t+t_{early}} = \phi(o_t)$.
- 3) Compute nearest neighbor score $s(\hat{y}_{t+t_{early}})$ as in Eq. 6.
- 4) Convert this score to the associated conformal p -value.
- 5) Report the p -value to the pilot and alert if it exceeds user-specified miss rate ϵ .

VI. EXPERIMENTS

In our experiments, we used the system described in Section IV and collected $N = 50$ unsafe trajectory rollouts in simulation with randomized plane parameters and discretization $\Delta t = 0.05$ [sec]. We specify that our warning system should alert $t_{early} = 0.25$ [sec] in advance of failure. For state prediction, we use a delay of 3 states i.e., $o_t = (y_{t-2}, y_{t-1}, y_t)$. Using the process described in Section V, we subsampled 50 observations per safe trajectory to fit the linear predictive model and for the nearest neighbor classification.

In Figure 1 we show an unsafe and safe trajectory for two test rollouts of the rudder doublet maneuver. For the unsafe trajectory, we cut simulation at the time of system failure, marked with a dashed black line. We show the time t_{early} before then when the alert should trigger, as a blue dashed line. The safety cutoff of $|N_y| = 0.5$ is shown with horizontal red lines. We overlay the predicted outputs $\hat{y}_{t+t_{early}}$ with dashed lines of the same color as the true outputs. All predicted components match ground-truth well for both trajectories. Below each unsafe/safe trajectory, we show the associated p -value over time. Our runtime monitor performs well as the p -value rises near the time of system failure in the unsafe case, peaking around t_{early} before failure. In contrast, it remains at the lowest value during the safe trajectory.

To evaluate our method systematically, we varied the target miss rate ϵ and measured empirical miss rate and classification power. We trained the monitor on 10 different datasets, with $N = 50$ unsafe rollouts each, and tested on 500 trajectories.

¹This can be found by determining the index k at which $s(y)$ should be inserted into a sorted list of $\alpha_1, \dots, \alpha_N$ and then mapping to an ϵ via Eq. 8

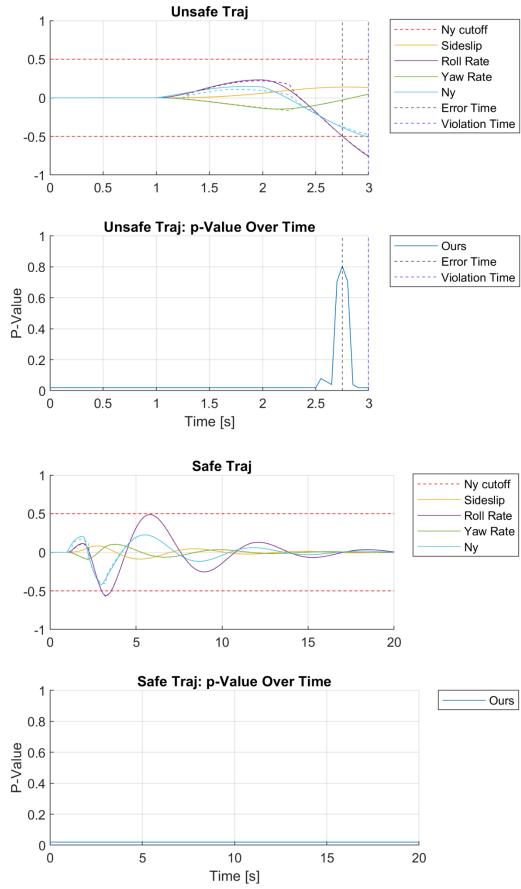


Fig. 1: True versus Predicted Outputs for Unsafe/Safe Trajectories with Associated p -Values

As alternatives to our approach we considered

- “No pred”: drops prediction model ϕ , applying conformal nearest neighbor directly to observation o_t .
- “PCA”: replaces ϕ with PCA to reduce o_t dimension (from 18 to 6).
- “Current $|N_y|$ ”: replaces the conformal score with $s(y) = -|N_y|$ which increases as the safety limit is approached.
- “Predicted $|N_y|$ ”: applies $s(y) = -|N_y|$ to the predicted future state $\hat{y}_{t+t_{early}}$.

The upper plot of Figure 2 shows the empirical miss rate, the fraction of unsafe test trajectories where no alert is issued by t_{early} seconds before failure. The empirical miss rate falls below the theoretical upper bound of ϵ , validating the conformal guarantee. For “Pred $|N_y|$ ” the bound appears slightly violated, due to taking an empirical average, but this would vanish with more fits. The bottom plot of Figure 2 shows the classification power i.e., the probability of no alert during a safe trajectory, which should ideally remain high across ϵ .

Our approach retains good classification power across the ϵ range and outperforms all alternatives except “Pred $|N_y|$ ”. Unlike our general-purpose nearest neighbor score, this baseline uses the known safety specification $|N_y| \leq 0.5$ to handcraft a conformal score. For more complex or implicit safety specifications, such tailored scores may be unavailable or impractical.

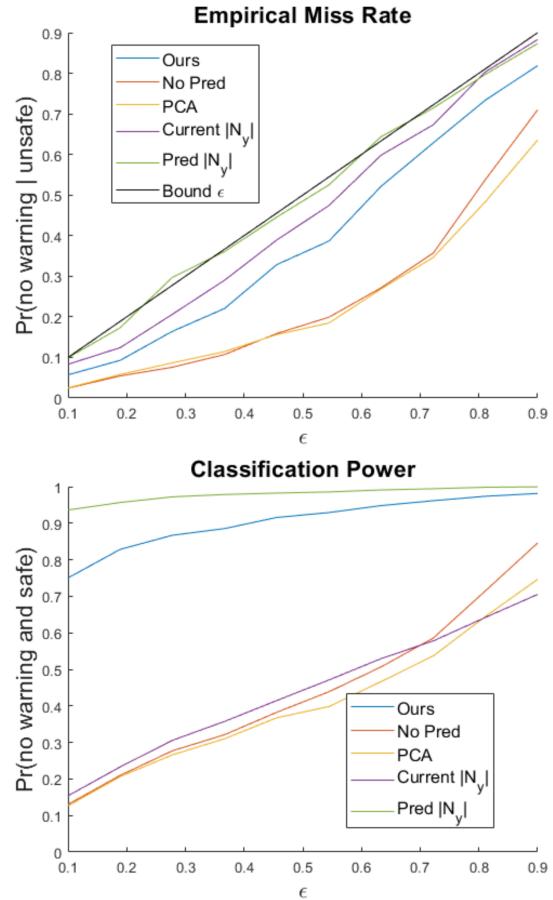


Fig. 2: Miss Rate and Classification Power for Varying ϵ

Still, the success of “Pred $|N_y|$ ” over “Current $|N_y|$ ” highlights the value of the prediction model. Similarly, we note that replacing the prediction model with PCA, designed for data reconstruction not prediction, performs poorly.

VII. CONCLUSION

This work serves as a proof-of-concept for providing test pilots with a statistics-based preemptive runtime safety monitor. Our approach, featuring a prediction model, nearest neighbor classification, and conformal prediction, could be used more generally for data-driven safety learning.

There are several exciting directions for future research. It would be valuable to use stochastic and reactive human models within the runtime monitor. The current prediction step could then also forecast the human’s actions, in turn changing the predicted future state. Secondly, it would be valuable to test with more abstract/complex safety specifications (e.g., based on structural/load analysis, gain/phase margin, or pilot intervention) where a custom conformal score cannot be easily defined. Thirdly, it would be interesting to consider other prediction models, which could be probabilistic or adapt online.

REFERENCES

- [1] J. A. Grauer and E. A. Morelli, “Generic global aerodynamic model for aircraft,” *Journal of Aircraft*, vol. 52, no. 1, pp. 13–20, 2015.

- [2] K. Nusrath TK, D. Kaliyari, J. Singh, and V. V. Patel, “Enhancing real time monitoring support for safe envelope expansion,” *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 254–259, 2016. 4th IFAC Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2016.
- [3] L. Lindemann, M. Cleaveland, G. Shim, and G. J. Pappas, “Safe planning in dynamic environments using conformal prediction,” *IEEE Robotics and Automation Letters*, 2023.
- [4] M. Cleaveland, I. Lee, G. J. Pappas, and L. Lindemann, “Conformal prediction regions for time series using linear complementarity programming,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 20984–20992, 2024.
- [5] K. J. Strawn, N. Ayanian, and L. Lindemann, “Conformal predictive safety filter for rl controllers in dynamic environments,” *IEEE Robotics and Automation Letters*, 2023.
- [6] A. Dixit, L. Lindemann, S. X. Wei, M. Cleaveland, G. J. Pappas, and J. W. Burdick, “Adaptive conformal prediction for motion planning among dynamic agents,” in *Learning for Dynamics and Control Conference*, pp. 300–314, PMLR, 2023.
- [7] A. Muthali, H. Shen, S. Deglurkar, M. H. Lim, R. Roelofs, A. Faust, and C. Tomlin, “Multi-agent reachability calibration with conformal prediction,” in *2023 62nd IEEE Conference on Decision and Control (CDC)*, pp. 6596–6603, IEEE, 2023.
- [8] R. Tumu, M. Cleaveland, R. Mangharam, G. J. Pappas, and L. Lindemann, “Multi-modal conformal prediction regions with simple structures by optimizing convex shape templates,” 2024.
- [9] R. Sinha, A. Sharma, S. Banerjee, T. Lew, R. Luo, S. M. Richards, Y. Sun, E. Schmerling, and M. Pavone, “A system-level view on out-of-distribution data in robotics,” *arXiv preprint arXiv:2212.14020*, 2022.
- [10] R. Laxhammar and G. Falkman, “Conformal prediction for distribution-independent anomaly detection in streaming vessel data,” in *Proceedings of the first international workshop on novel data stream pattern mining techniques*, pp. 47–55, 2010.
- [11] R. Laxhammar and G. Falkman, “Sequential conformal anomaly detection in trajectories based on hausdorff distance,” in *14th international conference on information fusion*, pp. 1–8, IEEE, 2011.
- [12] J. Smith, I. Nouretdinov, R. Craddock, C. Offer, and A. Gammerman, “Anomaly detection of trajectories with kernel density estimation by conformal prediction,” in *Artificial Intelligence Applications and Innovations: AIAI 2014 Workshops: CoPA, MHDW, IIVC, and MT4BD, Rhodes, Greece, September 19-21, 2014. Proceedings 10*, pp. 271–280, Springer, 2014.
- [13] P. Contreras, O. Shorinwa, and M. Schwager, “Out-of-distribution runtime adaptation with conformalized neural network ensembles,” *arXiv preprint arXiv:2406.02436*, 2024.
- [14] R. Sinha, E. Schmerling, and M. Pavone, “Closing the loop on runtime monitors with fallback-safe mpc,” in *2023 62nd IEEE Conference on Decision and Control (CDC)*, pp. 6533–6540, IEEE, 2023.
- [15] K. Y. Chee, T. C. Silva, M. A. Hsieh, and G. J. Pappas, “Uncertainty quantification and robustification of model-based controllers using conformal prediction,” in *Proceedings of the 6th Annual Learning for Dynamics and Control Conference* (A. Abate, M. Cannon, K. Margellos, and A. Papachristodoulou, eds.), vol. 242 of *Proceedings of Machine Learning Research*, pp. 528–540, PMLR, 15–17 Jul 2024.
- [16] A. O. Feldman, J. A. Vincent, M. Adang, J. E. Low, and M. Schwager, “Learning robot safety from sparse human feedback using conformal prediction,” 2025.
- [17] G. Shafer and V. Vovk, “A tutorial on conformal prediction.,” *Journal of Machine Learning Research*, vol. 9, no. 3, 2008.
- [18] A. N. Angelopoulos and S. Bates, “A gentle introduction to conformal prediction and distribution-free uncertainty quantification,” *arXiv preprint arXiv:2107.07511*, 2021.
- [19] H. Papadopoulos, K. Proedrou, V. Vovk, and A. Gammerman, “Inductive confidence machines for regression,” in *Machine learning: ECML 2002: 13th European conference on machine learning Helsinki, Finland, August 19–23, 2002 proceedings 13*, pp. 345–356, Springer, 2002.
- [20] A. Gammerman, V. Vovk, and V. Vapnik, “Learning by transduction,” in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI’98, (San Francisco, CA, USA), p. 148–155, Morgan Kaufmann Publishers Inc., 1998.