A Reference Architecture for Closed-Loop Predictive Quality Control under Limited Process Observability: A Centrifugal Casting Case Study

Peter Josef Haupts[1] and Mohieddine Jelali[1]

Abstract— **Industrial quality control often relies on end-of-line inspection, where defect labels become available only after most process decisions are irreversible. This paper proposes a closed-loop predictive quality control (CLPQC) reference architecture for multi-stage manufacturing under partial observability. It aggregates heterogeneous sensor data and stage history into a context representation, predicts defect risk together with decision-relevant uncertainty and out-of-distribution indicators, and maps these estimates to actionable interventions via a risk-bounded operating envelope. The resulting layer supports conservative, interpretable control recommendations and safe fall-back behavior under distribution shift. The centrifugal casting case study illustrates how the proposed architecture enables earlier, uncertainty-aware interventions compared to reactive inspection, while prospective closed-loop validation remains future work.**

I. INTRODUCTION

Centrifugal casting is a widely used process for producing axisymmetric components, offering high material yield and favorable mechanical properties when the process chain is well controlled. However, surface and near-surface defects may still arise from coupled thermo-fluid and metallurgical mechanisms and from variability accumulated across the multi-stage process chain. Casting process fundamentals and defect-relevant mechanisms, including centrifugal casting, are documented in established references [1], [2].

Recent progress in Machine Learning (ML) and Deep Learning (DL) has substantially improved automated visual inspection (AVI) for surface defect detection and segmentation in industrial settings. Systematic reviews report strong performance gains from DL-based detection architectures, while also highlighting persistent challenges such as limited labeled data, domain shift, and real-time constraints [3]. In centrifugal casting, these advances primarily enable reactive quality assessment after the part has been produced, i.e., after most process decisions are no longer reversible.

In parallel, predictive quality has emerged as a data-driven paradigm to estimate final quality outcomes from process and sensor data [4]. Representative implementations demonstrate that predictive models can reduce inspection load and support earlier decision-making by shifting quality assessment upstream in the process chain [5]. Yet, transferring predictive quality from offline estimation to closed-loop quality control remains difficult in many manufacturing domains.

[1]P. J. Haupts and M. Jelali are with the Institute of Product Development and Engineering Design (IPK) and with the Cologne Lab for Artificial Intelligence and Smart Automation (CAISA), TH Köln - University of Applied Sciences, Campus Deutz, Betzdorfer Straße 2, 50679 Cologne, Germany, (e-mail: peter_josef.haupts@th-koeln.de; mohieddine.jelali@th-koeln.de).

A key reason is that centrifugal casting exhibits limited and stage-dependent process observability: relevant latent states (e.g., melt cleanliness, local solidification dynamics, interfacial conditions) are only partially measurable, and quality feedback is typically delayed until post-process inspection. From a systems perspective, the process is naturally multi-stage, with error propagation and constrained intervention authority across stages [6]. From a decision-making perspective, this resembles planning and control under partial observability [7], where classical feedback control assumptions do not hold. Moreover, industrial monitoring and diagnosis frequently relies on data-driven methods precisely because first-principles observability is limited [8].

Contributions: This paper (i) formulates centrifugal casting quality control as a multi-stage, partially observable decision problem; (ii) proposes an implementation-oriented closed-loop predictive quality control (CLPQC) reference architecture linking stage-wise observability and intervention windows; and (iii) instantiates it in a centrifugal casting case study with defect-relevant process parameters reported in the literature [9].

II. PROCESS CHARACTERISTICS AND PROBLEM DEFINITION

Centrifugal casting quality is shaped by a sequence of process stages whose effects accumulate over time. Defect formation is therefore best viewed as the outcome of a multi-stage decision process with (i) heterogeneous observability, (ii) stage-dependent control authority, and (iii) delayed quality feedback [6]. This section formalizes these properties and states the closed-loop predictive quality control problem addressed in this work.

A. Multi-Stage Nature of Centrifugal Casting and Quality Formation

A typical centrifugal casting route can be decomposed into an ordered set of stages, e.g., melting/treatment, mould preparation, pouring, and spinning/solidification [1]. Decisions made early (e.g., melt treatment or mould condition) can influence downstream flow and solidification outcomes, while later stages often cannot fully compensate for earlier deviations. This aligns with general findings in multistage quality engineering, where variation propagates along a process chain and interacts across stages [6].

B. Stage-Wise Observability and Intervention Potential

Let $k \in \{1, \ldots, K\}$ index the process stage and t index time within a stage. We distinguish (latent) process states

$x_{k,t}$ from available measurements $z_{k,t}$. Let $u_{k,t}$ denote the applied control input / intervention at stage k and time t (e.g., setpoints, recipe selections, or operator-authorized adjustments).

In many industrial settings, monitoring and diagnosis must operate on partial, noisy, and heterogeneous measurements, motivating data-driven modeling for state estimation and fault detection [8]. In centrifugal casting, both observability and feasible intervention authority are strongly stage-dependent: melting and mould preparation are typically sparsely observed and largely irreversible, pouring offers only a short corrective window, and spinning provides the highest actuation authority with comparatively reliable sensing [1], [9].

C. Closed-Loop Predictive Quality Control Problem

Let y denote the final quality outcome (e.g., defect presence/type/severity or spatial defect indicators) obtained after inspection. Because inspection is typically downstream, the label y becomes available only after completion of the process chain. We express this delayed feedback abstractly as

$$y \sim p(y \mid x_{K,T}), \tag{1}$$

where $x_{K,T}$ denotes the (latent) terminal state after the final stage and T is the terminal time index of stage K.

The decision-making entity does not observe the latent state $x_{k,t}$ directly, but only the information (i.e., past measurements and applied actions across stages) available up to (k,t). We therefore define the information history as

$$H_{k,t} = \{z_{1,1:t_1}, u_{1,1:t_1}, \ldots, z_{k,1:t}, u_{k,1:t}\}, \tag{2}$$

where t_i denotes the terminal time index of stage $i < k$ and $z_{i,1:t_i}$ (analogously $u_{i,1:t_i}$) denotes the sequence $\{z_{i,1}, \ldots, z_{i,t_i}\}$. Under limited observability, control decisions can be interpreted as operating on a belief over latent states,

$$b_{k,t} = p(x_{k,t} \mid H_{k,t}), \tag{3}$$

or on a learned surrogate representation derived from $H_{k,t}$.

The objective of closed-loop predictive quality control is to use the available measurement and action history $H_{k,t}$ to (i) predict y early enough to be actionable, and (ii) select interventions $u_{k,t}$ within feasible decision windows of each stage. Formally, this corresponds to a partially observable sequential decision problem [7]:

$$x_{k,t+1} = f_k(x_{k,t}, u_{k,t}, w_{k,t}), \tag{4}$$

$$z_{k,t} = h_k(x_{k,t}, v_{k,t}). \tag{5}$$

where $f_k(\cdot)$ and $h_k(\cdot)$ denote (generally unknown) stage-dependent state-transition and measurement functions, with process noise $w_{k,t}$ and measurement noise $v_{k,t}$. The closed-loop goal is to find a policy π that maps available information to interventions, $u_{k,t} = \pi(H_{k,t})$, minimizing expected defect risk and intervention cost:

$$\pi^\star = \arg\min_{\pi} \; E\left[\ell(y) + \sum_{k,t} \kappa(u_{k,t}) \right]. \tag{6}$$

Here, $\ell(y)$ denotes a task-specific quality loss (e.g., $\ell(y) = I\{y \in \mathcal{Y}_d\}$ for binary defect labels, where $I\{\cdot\}$ is the indicator function and $\mathcal{Y}_d$ denotes undesirable defect outcomes, or $\ell(y) = y$ for continuous severity), and $\kappa(u_{k,t})$ penalizes interventions (e.g., deviation from a nominal recipe, energy, or productivity impact). This formulation explicitly captures the central difficulty in centrifugal casting: the most influential decisions may occur at stages with the poorest observability, while the most observable stages often have limited authority to correct upstream causes [6], [8], [9].

III. Reference Architecture for Closed-Loop Predictive Quality Control

This section translates the formulation of Section II into a reference architecture for CLPQC. The focus is architectural rather than algorithmic: predictive-quality models are embedded in decision support despite delayed labels, partial observability, and stage-dependent interventions. In contrast to offline predictive quality pipelines [4], [5], CLPQC requires online signal ingestion, uncertainty-aware risk estimation, and decision logic that respects intervention windows and safety constraints. For modularity and deployability, the architecture separates data acquisition, analytics, and control decision-making, consistent with industrial reference-architecture thinking [10], [11].

A. Architecture Overview

Fig. 1 depicts the proposed CLPQC architecture. It is structured into five interacting layers:

- **L1) Data acquisition and synchronization:** stage-tagged ingestion of process signals and metadata from the casting line (OT/SCADA/PLC) and IT sources (MES/ERP), including time alignment and quality label association.
- **L2) Stage-aware context construction:** formation of a compact context representation that summarizes the available process history across melting, mould preparation, pouring, and spinning.
- **L3) Predictive quality and uncertainty estimation:** defect risk prediction together with calibrated confidence and distribution-shift indicators.
- **L4) Risk-based decision and control policy:** translation of predicted risk into constrained recommendations or parameter envelopes, respecting stage-dependent feasibility.
- **L5) Execution, monitoring, and learning:** human-in-the-loop or automated actuation, traceability, drift monitoring, and model update using delayed inspection feedback.

A key design choice is the explicit separation between offline lifecycle activities (data curation, model training, calibration, validation) and online runtime activities (signal ingestion, inference, decision support, execution). This separation is common in deployed predictive-quality solutions and supports industrial constraints such as limited compute on the shop floor and strict change management [5].

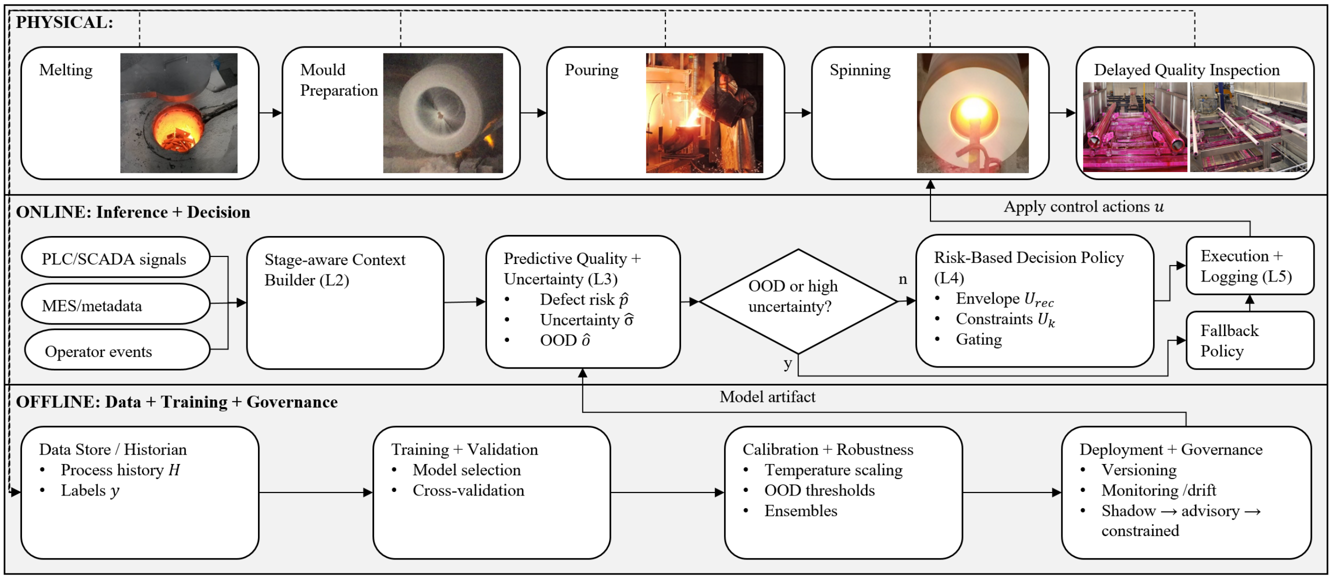

Fig. 1. CLPQC reference architecture with offline training/calibration/governance and online context building (L2), predictive quality with uncertainty/OOD (L3), and risk-based decision and execution (L4–L5). The spinning-stage control action u is exemplary; interventions $u_{k,t}$ can apply at any stage.

B. Stage-Aware Data Acquisition and Context Construction

1) Signal ingestion and stage tagging: Signals are ingested as stage-tagged sequences $(z_{k,t}, u_{k,t})$ and time-aligned across signal sources to form consistent trajectories for downstream context construction. Stage tagging partitions the process into K stages (cf. Section II) and enables consistent association of early-stage conditions with downstream outcomes. This is critical because influential decisions may occur in early stages with sparse measurements, while later stages are typically more observable but offer limited authority to correct upstream causes.

2) Context representation as a belief surrogate: Under limited observability, CLPQC cannot rely on direct access to the latent state $x_{k,t}$. Instead, the architecture constructs a stage-aware context vector $c_{k,t}$ as a compact surrogate for the belief state in (3) by aggregating the information history:

$$c_{k,t} = \phi(H_{k,t}), \tag{7}$$

where $H_{k,t}$ is defined in (2) and $\phi(\cdot)$ denotes a context encoder (e.g., feature engineering with temporal windows, sequence models, or learned embeddings). In industrial process monitoring, such latent representations are frequently used to support diagnosis and prediction when first-principles observability is limited [8]. In the centrifugal casting setting, $\phi(\cdot)$ must accommodate heterogeneous sampling rates, missing values, and discrete events (e.g., operator actions during mould preparation).

C. Predictive Quality and Uncertainty Estimation

Given $c_{k,t}$, the predictive quality module estimates defect-related outcomes to support decision-making at stage k. At runtime, for any candidate intervention $u \in \mathcal{U}_k$ (feasible under equipment, safety, and productivity constraints), the predictor outputs the estimated defect probability for the final inspected quality outcome y,

$$\hat{p}(u; c_{k,t}) = P(y \in \mathcal{Y}_d \mid c_{k,t}, u), \tag{8}$$

where $\mathcal{Y}_d$ denotes undesirable defect outcomes (binary, multi-class, or severity-thresholded events). Beyond point estimates, CLPQC requires decision-relevant uncertainty to support risk-aware intervention selection. We therefore introduce a learned prediction mapping Ψ that, for a given context $c_{k,t}$ and candidate intervention u, returns the estimated defect probability $\hat{p}(u; c_{k,t})$ together with an uncertainty indicator $\hat{\sigma}(u; c_{k,t})$ (e.g., predictive entropy or ensemble variance) and a distribution-shift / out-of-distribution (OOD) score $\hat{o}(u; c_{k,t})$:

$$(\hat{p}(u; c_{k,t}), \hat{\sigma}(u; c_{k,t}), \hat{o}(u; c_{k,t})) = \Psi(u; c_{k,t}). \tag{9}$$

Because Layer L4 relies on thresholds and envelopes, $\hat{p}$ should be calibrated (e.g., temperature scaling [12]). Practical robustness mechanisms include deep ensembles for uncertainty and simple OOD baselines for distribution-shift gating [13], [14].

D. Risk-Based Decision and Control Policy

1) From prediction to actionable interventions: Rather than attempting direct continuous control of defect formation (which is typically infeasible due to irreversibility and delayed inspection), the decision layer maps the action-conditional predictions in (9) to bounded interventions by evaluating candidate actions $u \in \mathcal{U}_k$.

We define the action-conditional expected-loss risk $\hat{r}(u; c_{k,t}) := E[\ell(y) \mid c_{k,t}, u]$, which reduces to $\hat{r} = \hat{p}$ for $\ell(y) = I\{y \in \mathcal{Y}_d\}$. To act conservatively under predictive uncertainty, we use

$$\tilde{r}(u; c_{k,t}) = \hat{r}(u; c_{k,t}) + \lambda\, \hat{\sigma}(u; c_{k,t}), \tag{10}$$

where $\lambda \geq 0$ tunes risk aversion.A practical output of the decision layer is a recommended operating envelope

$$\mathcal{U}_{k,t}^{\mathrm{rec}} = \{u \in \mathcal{U}_k \mid \tilde{r}(u; c_{k,t}) \leq \tau,\ \hat{o}(u; c_{k,t}) \leq \gamma\}, \quad (11)$$

where τ is a maximum acceptable conservative risk and γ is an OOD threshold that triggers fall-back behavior. In implementations where evaluating $\tilde{r}(u; c_{k,t})$ over u is not possible, (11) can be approximated via discrete candidate actions, local sensitivity analysis, or rule-based constraints informed by the predictive model and engineering knowledge.

2) Human-in-the-loop execution and auditability: Given the industrial safety implications, the architecture supports human-in-the-loop operation by default: the policy suggests parameter adjustments or feasible envelopes, while the operator (or supervisory logic) authorizes execution. To support trust and accountability, the decision module logs (i) the predicted risk, (ii) uncertainty/OOD indicators, and (iii) reason codes (e.g., top contributing features) alongside recommended actions.

E. Execution, Monitoring, and Learning Loop

1) Execution and fall-back: At runtime, recommended interventions are applied through existing interfaces (e.g., PLC setpoints or supervisory recipes). If $\hat{o}(u; c_{k,t}) > \gamma$ for the intended intervention u or key measurements are missing, the architecture falls back to conservative defaults or rule-based operation, ensuring safe degradation.

2) Delayed feedback assimilation: Inspection outcomes y become available after completion of casting and post-process quality assessment. The feedback layer closes the learning loop by aligning delayed labels with recorded histories $H_{k,t}$ and updating the training dataset. This enables periodic recalibration [12], drift detection, and controlled retraining cycles, which are standard in deployed predictive-quality systems [5], [4].

3) Model governance and change management: Model updates should follow standard governance practices (versioning, monitoring, controlled rollout) to ensure stability in production [10].

IV. CASE STUDY: CENTRIFUGAL CASTING IMPLEMENTATION

This section instantiates the CLPQC reference architecture for a centrifugal casting case. The goal is to set out a case-study blueprint that connects stage-wise process information, delayed inspection feedback, and constrained intervention windows into an actionable closed-loop pipeline without assuming full observability of defect formation. The case study is grounded in established centrifugal casting process knowledge and reported parameter sensitivities (e.g., rotation speed, pouring temperature, filling time) [1], [2], [15], [16], [17], [9].

A. Process Route and Data Sources

A typical centrifugal casting route is organized into melting/treatment, mould preparation, pouring, and spinning/solidification stages [1]. Each stage produces heterogeneous data with different sampling rates and reliability. In practice, online measurements are often best at later stages (e.g., rotational speed), whereas early-stage signals may be sparse or event-based (e.g., mould coating operations) [15].

Following Layer L1 of Section III, the runtime data acquisition ingests (i) time series from PLC/SCADA and sensors and (ii) batch/meta-information from production records (e.g., alloy grade, mould identifier, recipe). Table I provides an example of stage-tagged signals that are typical for centrifugal casting installations and sufficient to instantiate the proposed context and prediction modules.

B. Dataset Construction and Quality Labeling

To close the loop (Layer L5), each casting is associated with a unique identifier that links process histories to inspection outcomes. We consider post-process surface inspection as the primary source of quality labels, motivated by the strong recent progress of DL-based defect detection/segmentation in industrial AVI [3]. Concretely, the inspection module provides defect candidates (e.g., masks or polygons) together with severity attributes (e.g., area, length, confidence).

Depending on the objective, y may encode defect presence, type, or severity. For an initial, governance-friendly deployment, we use a binary label

$$y = I\{A_{\mathrm{def}} \geq A_{\min}\}, \quad (12)$$

and note that multi-class and continuous variants follow analogously. While the architecture supports all options, binary or coarse multi-class labels typically ease initial deployment and model governance in production [4], [5].

C. Context Construction at Decision Points

Closed-loop actuation is only possible at specific decision points (e.g., just before pouring or at the start of spinning). Therefore, the context encoder $\phi(\cdot)$ (Layer L2) is evaluated at stage-dependent times (k, t) to prevent label leakage and to ensure actionability:

$$c_{k,t} = \phi(H_{k,t}), \quad (13)$$

with $H_{k,t}$ as in (2). A practical $\phi(\cdot)$ combines stage aggregates (statistics/durations), event/categorical features (recipes, mould IDs, treatments), trajectory summaries (e.g., rpm peak/ramp/time-above-threshold), and explicit missingness indicators. Such feature-based context construction is consistent with data-driven monitoring practice in industrial settings [8] and enables a transparent initial instantiation before introducing more complex sequence models.

D. Predictive Model Instantiation and Calibration

Given a context vector $c_{k,t}$, the predictive module $\Psi(\cdot)$ (Layer L3) estimates defect risk for the final inspected outcome y. Following the generic definition in (9), the instantiated predictor provides $(\hat{p}(u; c_{k,t}), \hat{\sigma}(u; c_{k,t}), \hat{o}(u; c_{k,t}))$ for candidate actions u. In an engineering deployment, two complementary model families are commonly practical:

- **Interpretable baseline:** regularized logistic regression or gradient-boosted decision trees as the primary risk

TABLE I

EXAMPLE OF STAGE-TAGGED SIGNALS AND DECISION VARIABLES USED TO INSTANTIATE CLPQC FOR CENTRIFUGAL CASTING.

Stage	**Observations $z_{k,t}$ (examples)**	**Controls $u_{k,t}$ (examples)**	**Notes on actionability / relevance**
Melting / treatment	Furnace temperature profile, holding time, chemical analysis (batch), degassing events	Superheat target, treatment selection, holding time	Affects cleanliness and gas content; mostly irreversible downstream [1], [2]
Mould preparation	Mould preheat temperature, coating event logs, mould ID, downtime since last use	Preheat setpoint, coating thickness/recipe	Influences interfacial heat transfer; sparse sensing common [15]
Pouring	Pouring temperature, fill/start timestamps, estimated filling time/flow regime proxies	Pouring temperature setpoint, filling time target, gating/sequence parameters	Short decision window; affects entrainment and initial solidification [16], [17]
Spinning / solidification	Rotational speed (rpm) trace, acceleration profile, vibration/torque proxies (if available)	Speed profile, duration, acceleration limits	High actuation authority; affects filling stability and segregation [16], [17]
Inspection (delayed)	Surface images, defect masks/areas, severity scores	-	Used as delayed label y; may be automated via DL-based inspection [3]

estimator (robust with mixed numerical/categorical features).

- **Uncertainty-aware variant:** ensembles of predictors to estimate epistemic uncertainty and improve robustness under dataset shift [13]. Simple OOD baselines can be used as a safeguard for out-of-support operation [14].

Since CLPQC decisions depend on thresholds and risk envelopes, probability calibration is critical. Post-hoc temperature scaling provides a lightweight and effective calibration mechanism for modern neural predictors [12]. For tree models, isotonic regression or Platt scaling can be used analogously.

E. Closed-Loop Decision Example: Envelope-Based Spinning Control

To illustrate Layer L4, consider a decision at the start of spinning, i.e., stage $k = k_{\text{spin}}$ with $k_{\text{spin}} \in \{1, \dots, K\}$. In this stage, the controller selects among a discrete library of admissible speed profiles $\mathcal{U}_k^{\text{cand}} \subset \mathcal{U}_k$ (e.g., nominal, conservative, high-throughput). For each candidate $u \in \mathcal{U}_k^{\text{cand}}$, the predictor in (9) is evaluated under context $c_{k,t}$ to obtain the quantities required by the conservative risk score $\tilde{r}(u; c_{k,t})$ in (10). The decision layer then approximates the recommended operating envelope (11) by restricting evaluation to $\mathcal{U}_k^{\text{cand}}$ and selecting

$$u^\star = \arg \min_{u \in \mathcal{U}_k^{\text{cand}}} \tilde{r}(u; c_{k,t}) \quad \text{s.t.} \quad \hat{o}(u; c_{k,t}) \le \gamma. \qquad (14)$$

If the OOD constraint cannot be satisfied (e.g., $\hat{o}(u; c_{k,t}) > \gamma$ for all $u \in \mathcal{U}_k^{\text{cand}}$), the system falls back to a nominal recipe or operator-only decision making. Beyond discrete libraries, the same mechanism yields operating envelopes over continuous spinning parameters (e.g., peak rpm vs. ramp rate), where $\mathcal{U}_{k,t}^{\text{rec}}$ forms a safe region under the conservative risk and OOD constraints in (11). This is particularly relevant in centrifugal casting, where rotation speed and filling time strongly affect flow stability and defect susceptibility and a "critical range" may exist [16], [17]. Fig. 2 provides a schematic example of such an envelope output.

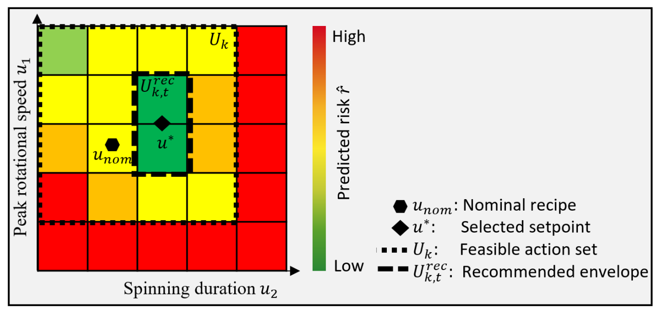

Fig. 2. Schematic example of an envelope-based CLPQC output at the spinning stage. The background shows the predicted defect risk $\hat{r}$ over two controllable spinning parameters (peak rotational speed u_1 and spinning duration u_2). The dotted boundary indicates the feasible action set $\mathcal{U}_k$, within which the dashed box denotes the recommended operating envelope $\mathcal{U}_{k,t}^{\text{rec}}$. The nominal recipe u_{nom} and the selected setpoint $u^\star$ are highlighted.

F. Evaluation Protocol and Deployment Readiness

The case-study instantiation is evaluated in an offline-to-online manner consistent with deployed predictive-quality systems [4], [5]:

- **Offline predictive performance:** AUROC / AUPRC (binary), macro-F1 (multi-class), and proper scoring rules such as the Brier score for probabilistic outputs.
- **Calibration and uncertainty:** expected calibration error (ECE) and reliability diagrams; uncertainty separation for in-/out-of-distribution samples; effectiveness of temperature scaling [12].
- **Decision quality:** fraction of castings for which $\mathcal{U}_{k,t}^{\text{rec}}$ is non-empty, recommendation stability across runs, and estimated risk reduction relative to nominal operation (counterfactual evaluation).

Finally, to support safe industrial adoption, deployment proceeds through staged modes (shadow $\rightarrow$ advisory $\rightarrow$ constrained actuation), with continuous monitoring for drift and periodic retraining using newly available inspection labels [5].

V. Discussion

Although instantiated for centrifugal casting, the proposed CLPQC architecture targets multi-stage manufacturing processes in which quality formation spans stages, relevant states are partially observable, inspection feedback is delayed, and intervention authority changes over the chain. Transfer to other domains requires redefining stages, context variables, feasible actions, labels, and safety limits, but not the logic of stage-aware prediction, uncertainty gating, and risk-bounded decision support. These characteristics also constrain what "closed-loop" can mean: rather than instantaneous regulation of defect formation, the loop is closed through feedback across production runs [4], [5].

Actionability under irreversibility and limited observability: A central implication of multi-stage casting is that early-stage decisions are largely irreversible, while later stages provide higher actuation authority. Consequently, downstream control often compensates for accumulated risk rather than correcting upstream causes. The architecture operationalizes this by mapping predictions to bounded recommendations (candidate actions or parameter envelopes) instead of unconstrained continuous control. This aligns with recipe-based operation, supports human-in-the-loop authorization, and enables fallback under missing inputs or distribution shift [14].

Uncertainty, calibration, and decision-centric evaluation: Because predictions drive interventions, calibration and uncertainty estimation are key enablers of safe CLPQC. Post-hoc calibration improves the interpretability of predicted probabilities as decision signals [12], while ensemble-based uncertainty provides a practical robustness mechanism against sparse sensing and dataset shift [13]. Evaluation should therefore include calibration quality, intervention coverage, envelope stability, and estimated risk reduction relative to nominal operation, while acknowledging that counterfactual evaluation from observational data remains limited.

Limitations and scope: The present work provides a reference architecture and case-study instantiation, but not yet a prospective closed-loop plant trial. Limitations include sparse early-stage sensing, delayed/noisy inspection labels, drift, latent confounding, and the need to validate thresholds and OOD gates before constrained actuation.

VI. Conclusion and Outlook

This paper addressed closed-loop predictive quality control for centrifugal casting under limited observability and multi-stage irreversibility. Starting from a stage-wise problem definition with delayed inspection feedback, we proposed a deployment-oriented reference architecture integrating (i) stage-tagged data acquisition and context construction, (ii) uncertainty-aware defect-risk prediction with calibration and OOD safeguards, and (iii) risk-based decision logic that produces bounded recommendations or operating envelopes consistent with stage-dependent constraints. A centrifugal casting case study illustrated how the architecture can be instantiated from realistic signals across melting, mould preparation, pouring, and spinning, and how predictions become decision support at feasible intervention points.

Future work will prioritize (i) prospective validation through controlled trials, (ii) comparison against predictive and decision-level baselines, (iii) improved early-stage sensing and robust proxy variables to reduce latent confounding, and (iv) integration of physics-based constraints or reduced-order models to enhance extrapolation. Model governance will be essential to transition from advisory operation to constrained actuation and to assess transferability across multi-stage manufacturing settings.

References

[1] ASM Handbook Committee, *ASM Handbook, Volume 15: Casting*, 10th ed., S. Lampman, C. Moosbrugger, and E. DeGuire, Eds. ASM International, 2008.

[2] J. Campbell, *Complete Casting Handbook: Metal Casting Processes, Metallurgy, Techniques and Design*, 2nd ed. Butterworth-Heinemann, 2015.

[3] Y. Ma, J. Yin, F. Huang, and Q. Li, "Surface defect inspection of industrial products with object detection deep networks: a systematic review," *Artificial Intelligence Review*, vol. 57, p. 333, 2024.

[4] H. Tercan and T. Meisen, "Machine learning and deep learning based predictive quality in manufacturing: a systematic review," *Journal of Intelligent Manufacturing*, vol. 33, pp. 1879–1905, 2022.

[5] J. Schmitt, J. Bönig, T. Borggräfe, G. Beitinger, and J. Deuse, "Predictive model-based quality inspection using machine learning and edge cloud computing," *Advanced Engineering Informatics*, vol. 45, p. 101101, 2020.

[6] J. Shi and S. Zhou, "Quality control and improvement for multistage systems: A survey," *IIE Transactions*, vol. 41, pp. 744–753, 2009.

[7] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[8] S. J. Qin, "Survey on data-driven industrial process monitoring and diagnosis," *Annual Reviews in Control*, vol. 36, no. 2, pp. 220–234, 2012.

[9] K. Mazloum, A. Al Njjar, and A. Sata, "Fea-assisted minimization of shrinkage porosity and air entrainment defects in a356 vertical centrifugal castings using a statistical predictive model," *International Journal on Interactive Design and Manufacturing*, vol. 19, pp. 7009–7025, 2025.

[10] Industry IoT Consortium, "The industrial internet reference architecture," Industry IoT Consortium, Tech. Rep. Version 1.10, Nov. 2022, foundational document. [Online]. Available: https://www.iiconsortium.org/wp-content/uploads/sites/2/2022/11/IIRA-v1.10.pdf

[11] Plattform Industrie 4.0, "Reference architectural model industrie 4.0 (rami4.0) – an introduction," Plattform publication, Aug. 2018. [Online]. Available: https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/rami40-an-introduction.html

[12] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70, 2017, pp. 1321–1330.

[13] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 6402–6413.

[14] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *International Conference on Learning Representations*, 2017.

[15] S. Mohapatra, H. Sarangi, and U. K. Mohanty, "Effect of processing factors on the characteristics of centrifugal casting," *Manufacturing Review*, vol. 7, p. 26, 2020.

[16] X. Wang, R. Chen, Q. Wang, S. Wang, Y. Li, Y. Xia, G. Zhou, G. Li, and Y. Qu, "Influence of rotation speed and filling time on centrifugal casting through numerical simulation," *International Journal of Metalcasting*, vol. 17, pp. 1326–1339, 2023.

[17] M. Xin, Z. Wang, B. Lu, and Y. Li, "Effects of different process parameters on microstructure evolution and mechanical properties of 2060 Al–Li alloy during vacuum centrifugal casting," *Journal of Materials Research and Technology*, vol. 21, pp. 54–68, 2022.