machines

MDPI

*Systematic Review*

# Edge–Cloud Collaboration for Machine Condition Monitoring: A Comprehensive Review of Mechanisms, Models, and Applications

**Liyuan Yu [1], Jitao Fang [2,3,*], Qiuyan Wang [2,3], Fajia Li [2,3] and Haining Liu [2,3]**

[1] Graduate School of Shandong Sport University, Shandong Sport University, Jinan 250102, China; yuliyuan@sdpei.edu.cn
[2] School of Mechanical Engineering, University of Jinan, Jinan 250022, China; wangqy0@stu.ujn.edu.cn (Q.W.); me_lifj@ujn.edu.cn (F.L.); me_liuhn@ujn.edu.cn (H.L.)
[3] Shandong Provincial Key Laboratory of Sensor Technology and High Precision Weighing Instruments, Jinan 250102, China
* Correspondence: me_fangjt@ujn.edu.cn

## Abstract

Machine condition monitoring increasingly depends on distributed sensing, edge intelligence, and cloud analytics, yet timely and trustworthy health assessment remains constrained by latency, bandwidth, privacy, and reliability requirements. Cloud-only architectures provide scalable computation and historical data integration but often fail to satisfy real-time industrial needs, whereas edge-only deployments are limited by restricted computing resources and fragmented local knowledge. Edge–cloud collaboration has, therefore, emerged as a practical architecture for distributing perception, inference, learning, and coordination across hierarchical industrial systems. This review examines 147 publications on edge–cloud collaboration for machine condition monitoring published between 2019 and February 2026. A four-dimensional taxonomy is developed to organize the literature into model-centric, data-centric, resource and task-centric, and architecture and trust-centric mechanisms, while 13 survey and review papers are considered separately for contextual comparison. On this basis, the review analyzes representative collaboration mechanisms and enabling technologies, with particular attention to federated learning, transfer learning, knowledge distillation, digital twins, and deep reinforcement learning, and surveys their deployment in manufacturing, energy, transportation, and infrastructure monitoring scenarios. The literature remains dominated by model-centric collaboration, while architecture and trust-centric studies increasingly provide the system foundations required for practical deployment. The review further identifies major open challenges, including robust generalization under changing operating conditions, efficient data transmission, real-time resource coordination, interoperability, and trustworthy large-scale deployment, and outlines future directions in foundation-model-based edge–cloud collaboration, continual learning, dual digital twins, trustworthy collaboration, and privacy-preserving industrial ecosystems.

**Keywords:** edge computing; cloud computing; edge–cloud collaboration; machine condition monitoring; prognostics and health management; federated learning; digital twin; industrial internet of things; intelligent manufacturing

Check for updates

Academic Editors: Xuefang Xu, Zijian Qiao, Mengdi Li and Peng Mei

Received: 18 March 2026
Revised: 14 April 2026
Accepted: 20 April 2026
Published: 24 April 2026

**Copyright:** © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

## 1. Introduction

The reliable operation of machinery underpins productivity, safety, and sustainability across virtually every industrial sector, because unplanned failures in manufacturing plants,

power generation facilities, transportation systems, and critical infrastructure can cause severe production losses, higher maintenance costs, and significant safety risks. As industry advances toward the smart manufacturing paradigm of Industry 4.0 and the more human-centered vision of Industry 5.0 [1], production assets are becoming increasingly complex, interconnected, and instrumented, which makes continuous machine-health assessment both more necessary and more feasible. In this context, machine condition monitoring (MCM) and prognostics and health management (PHM) provide the methodological foundation for shifting maintenance from reactive or schedule-based practice toward condition-aware and predictive decision making [2,3]. By analyzing operational signals such as vibration, acoustic emissions, temperature, and electrical signatures, MCM supports anomaly detection, fault diagnosis, and remaining useful life (RUL) prediction, thereby helping maintain production continuity, maintenance efficiency, product quality, and operational safety.

Within smart manufacturing, the rise of the Industrial Internet of Things (IIoT) has greatly expanded the scope and resolution of MCM. Modern production lines deploy high-frequency vibration, current, thermal, and acoustic sensors, producing continuous data streams that create both new opportunities for intelligent diagnosis and prognosis and new pressure on communication and storage resources [4,5]. As a result, industrial monitoring systems face a fundamental dual requirement: they must support low-latency responses for time-sensitive maintenance decisions while also exploiting large and diverse datasets for more accurate diagnosis, prognosis, and fleet-level learning.

Cloud platforms address the analytical side of this requirement effectively by providing scalable computation, storage, and cross-site integration [6]. In practice, centralized cloud infrastructures are attractive because they can consolidate historical data from distributed assets and support large-scale diagnostic and prognostic model development. However, cloud-only pipelines remain problematic for industrial operation: communication latency can violate real-time requirements [7], continuous uploading of raw streams can impose substantial transmission burden, and centralized data transfer raises privacy and compliance concerns [8].

Edge computing addresses these constraints by placing computation near the data source, such as gateways, programmable logic controllers (PLCs), and embedded controllers, thereby enabling lower-latency inference and more localized data handling. Recent reviews of IoT-oriented edge diagnosis emphasize that signal acquisition, preprocessing, feature extraction, and pattern recognition increasingly need to be executed close to machines to satisfy real-time maintenance requirements [5]. Yet edge-only deployments are also limited: constrained hardware budgets often require compression-oriented model design [9], lightweight deployment remains difficult when models are developed mainly for high-performance platforms rather than embedded hardware [10], and distributed nodes can become isolated knowledge silos [11].

As illustrated in Figure 1, neither tier alone can satisfy the full requirements of industrial MCM. The left panel shows cloud-centric processing with centralized analysis, the middle panel shows edge-centric processing with local autonomy, and the right panel shows edge–cloud collaboration, in which fast local response and global model coordination are jointly optimized. Edge–cloud collaboration (ECC), therefore, emerges as a natural intelligent computing paradigm for MCM: latency-sensitive and data-local tasks can be handled at the edge, while large-scale coordination, model updating, and long-horizon analytics remain in the cloud. This division is also consistent with the broader transition from "cloud manufacturing" to "cloud–edge collaborative manufacturing" discussed in recent manufacturing-architecture literature [12]. Across these configurations, the common

objective is to transform high-dimensional sensor streams into reliable and actionable machine-health assessment.

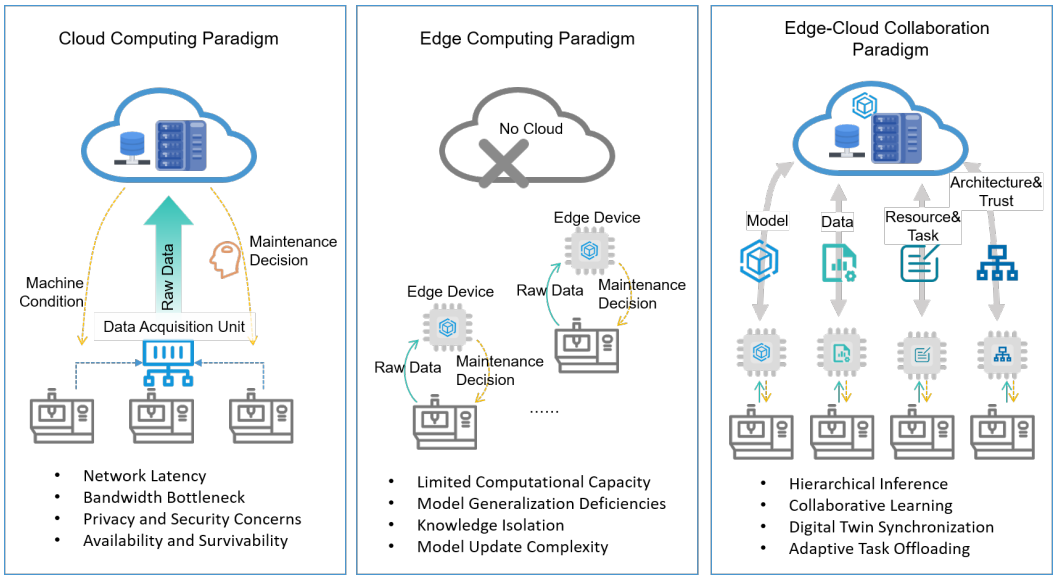



**Figure 1.** Comparison of cloud computing, edge computing, and edge–cloud collaboration architectures for machine condition monitoring.

Operationally, existing MCM studies instantiate ECC through four recurring collaboration patterns rather than a single architecture. Hierarchical inference uses edge nodes for fast preliminary screening and cloud nodes for deeper analysis [13,14]. Collaborative model training (especially federated learning (FL)) coordinates distributed edge learning through cloud-side aggregation [15,16]. Digital twin (DT) synchronization combines richer cloud twins with lighter edge twins for real-time support [17,18]. Adaptive task offloading shifts workloads between tiers according to runtime conditions and service objectives [19,20]. This diversity motivates a systematic comparison under a common analytical framework.

However, the literature still lacks a unified treatment of ECC for MCM. Existing surveys address edge artificial intelligence (AI) and predictive maintenance architectures [21,22], edge computing for Internet of Things (IoT)-based machine signal processing and fault diagnosis [5], anomaly detection and failure root cause analysis in service-based cloud applications [23], deep learning for bearing fault diagnosis [24], or federated learning in cloud-edge systems [6] in relative isolation. None provides a taxonomy-driven, mechanism-level synthesis that simultaneously covers model-centric collaboration, data transmission strategies, resource orchestration, and system architecture and trust–the four dimensions that jointly determine how an ECC system for MCM is designed, deployed, and governed.

To address this gap more explicitly and to align the review with mechanisms, models, and applications, this paper is organized around four research questions:

RQ1: What core collaboration mechanisms define edge–cloud collaboration for machine condition monitoring, and how can the literature be organized into a coherent taxonomy?

RQ2: What computational models, algorithms, and enabling strategies support these mechanisms, and what trade-offs do they introduce in accuracy, latency, communication, privacy, and resource usage?

RQ3: How are these mechanisms and models applied across industrial domains, and how do sensing conditions, operational objectives, and quality-of-service requirements shape deployment choices?

RQ4: What methodological, technical, and deployment gaps still limit scalable and trustworthy adoption, and which future research directions are most critical?

The main contributions of this review are summarized as follows:

(1) A four-dimensional taxonomy is developed for 147 reviewed publications, providing a structured map of the major edge–cloud collaboration mechanisms used in machine condition monitoring.
(2) Representative computational models and enabling strategies—including federated learning, transfer learning, knowledge distillation, digital twin architectures, and deep reinforcement learning—are comparatively analyzed with respect to operating principles, communication cost, computational burden, and deployment suitability.
(3) A cross-domain synthesis is provided across manufacturing, energy systems, transportation, and infrastructure, highlighting how sensing characteristics, service objectives, and QoS constraints shape mechanism selection and deployment design.
(4) A consolidated discussion of methodological limitations, unresolved deployment challenges, and future research directions is provided, covering taxonomy reproducibility, distribution-shift robustness, privacy-preserving exchange, real-time orchestration, interoperability, adaptive lifelong learning, industrial foundation models, and trustworthy collaboration ecosystems.

The remainder of this paper is organized as follows. Section 2 describes the literature search methodology and presents the four-dimensional taxonomy. Section 3 provides a detailed analysis of each collaboration mechanism with representative works. Section 4 examines key enabling technologies and their computational properties. Section 5 surveys application domains and deployment patterns. Section 6 identifies critical challenges and open issues. Section 7 outlines future research directions, and Section 8 concludes the review.

To keep terminology consistent, "edge–cloud collaboration" is used as the default term throughout this manuscript, and "ECC" is used as its acronym. When cited studies explicitly use the term "edge–cloud cooperation", that full wording is retained for consistency with the original papers.

## 2. Review Methodology and Taxonomy

### *2.1. Literature Search and Selection Criteria*

A systematic literature search was conducted across five major databases: Web of Science, ScienceDirect, IEEE Xplore, SpringerLink, and MDPI. The query structure was defined as follows: (*"edge-cloud*" OR "cloud-edge*" OR "edge cloud*" OR "edge intelligence" OR "cloud intelligence"*) AND (*"machine condition monitoring" OR "fault diagnos*" OR "prognostic*" OR "predictive maintenance" OR "health management" OR "anomaly detection" OR "PHM" OR "RUL" OR "bearing fault" OR "rotating machinery"*). A total of 1372 records were retrieved, including 346 from Web of Science, 40 from ScienceDirect, 486 from IEEE Xplore, 280 from SpringerLink, and 220 from MDPI.

The query intentionally included terminology variants and wildcard forms for retrieval completeness; after screening, the synthesis uses "edge–cloud collaboration" as the default term, while "edge–cloud cooperation" is retained only when reflecting the wording of cited studies.

The search was restricted to peer-reviewed journal articles published between January 2019 and February 2026, capturing the rapid evolution of edge–cloud collaboration during the Industry 4.0 period. As it is illustrated in Figure 2, an explicit PRISMA-style protocol was followed, consisting of the following: (i) database retrieval, (ii) duplicate removal,

(iii) title/abstract screening, (iv) manual full-text eligibility assessment, and (v) final inclusion for coding and synthesis. Among the 1372 retrieved records, 216 duplicated papers were removed, leaving 1156 records for screening. Title and abstract screening excluded 805 records. The remaining 351 records then underwent manual full-text eligibility assessment using the inclusion and exclusion criteria listed in Table 1, after which 204 studies were excluded. This process resulted in a final corpus of 147 included studies, including 13 survey/review papers retained for contextual comparison.

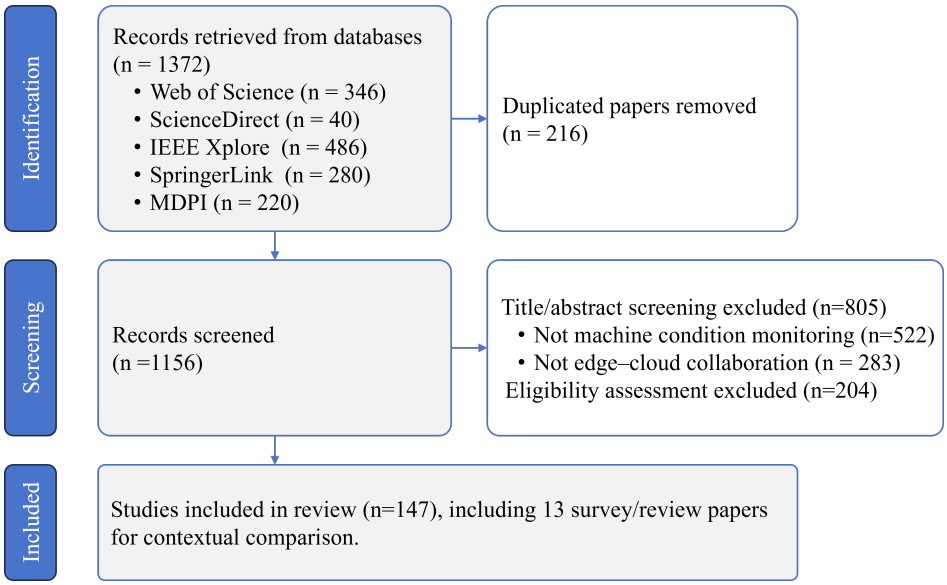



**Figure 2.** PRISMA-style literature screening workflow and study counts.

**Table 1.** Inclusion and Exclusion Criteria Used for Study Selection.

| No. | Include If | Exclude If |
|---|---|---|
| 1 | Peer-reviewed journal/conference publication (2019–February 2026). | Non-peer-reviewed source. |
| 2 | Explicit edge–cloud collaboration architecture or algorithmic mechanism. | Cloud-only, edge-only, or generic IoT work without explicit edge–cloud collaboration. |
| 3 | Direct relevance to machine condition monitoring, fault diagnosis, prognostics, or predictive maintenance. | Outside machine condition monitoring scope. |
| 4 | Sufficient technical detail for mechanism attribution and comparative analysis. | Insufficient methodological detail or no clear technical contribution. |

Study selection was conducted by three authors. Two reviewers independently screened each record at the title/abstract and full-text stages, while a third reviewer resolved disagreements through consensus meetings. Agreement statistics were tracked during pilot and full screening rounds; Cohen's $\kappa$ exceeded 0.80 at both stages, indicating substantial consistency.

To improve review rigor, a structured quality appraisal was applied to all included studies using four dimensions: methodological clarity, reproducibility of the pipeline, validity of evaluation design, and practical deployment relevance. Each dimension was scored on a three-level scale (0: insufficient, 1: partial, 2: clear), and quality notes were used to weight narrative synthesis emphasis when conflicting claims appeared.

After quality appraisal, all 147 included studies were coded for structured analysis. For each paper, the coding recorded its primary taxonomy label, any secondary mechanism

labels, and key comparative attributes such as application domain and enabling technologies. Primary-label coding assigned each paper to one dominant mechanism based on its principal collaboration object and optimization objective, while secondary labels captured hybrid designs (e.g., federated learning combined with task orchestration or digital twins). Geographic and institutional diversity was observed across contributions from China (42%), the United States (18%), Europe (23%), and other regions (17%), reflecting broad international interest.

### *2.2. Four-Dimensional Taxonomy of Edge–Cloud Collaboration*

Edge–cloud collaboration manifests through diverse mechanisms reflecting the heterogeneity of MCM requirements. To systematically organize this body of literature, a four-dimensional taxonomy was proposed, categorizing collaboration based on the *primary interaction object* and *optimization objective* (Figure 3):

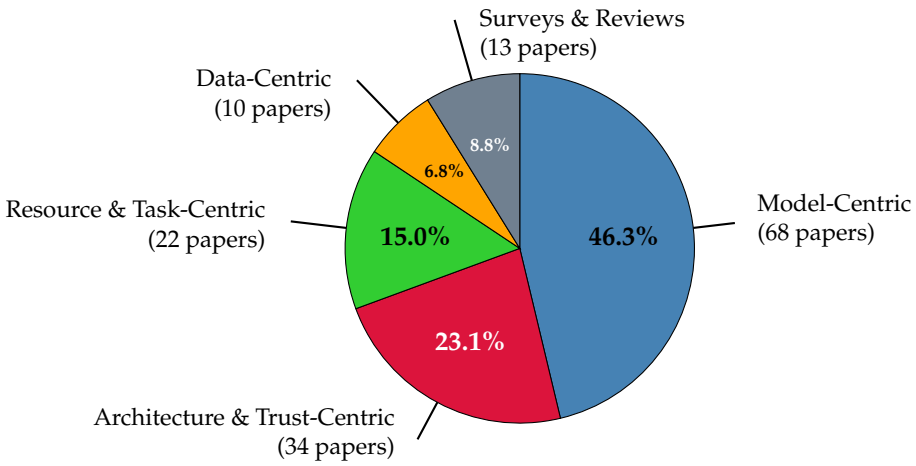



**Figure 3.** Primary-label distribution of 147 reviewed studies across five taxonomy groups.

Figure 3 reports primary-label proportions for transparent one-to-one categorization. Because many studies combine mechanisms (for example, federated learning with dynamic task orchestration), overlap is explicitly captured through secondary labels and summarized in Figure 4. To avoid false numerical precision, the heat map is presented as a qualitative co-occurrence summary rather than a statistical correlation matrix. The diagonal label "Primary" marks the dominant taxonomy assignment used in the main coding of each paper, while the off-diagonal labels are derived from the relative frequency of secondary-label overlap across the reviewed corpus: "High" denotes frequently recurring hybrid combinations, and "Medium" denotes moderate but still clearly recurrent cross-category overlap.

The four collaboration mechanisms can be summarized as follows:

- **Model-centric collaboration** centers on the transmission of model parameters, gradients, or knowledge representations between edge and cloud tiers. Its main objectives are joint model training, knowledge transfer, model compression, and distributed inference. Representative technologies include federated learning (FL), transfer learning (TL), knowledge distillation (KD), split computing, continual learning, and domain adaptation. This category contains 68 papers (46.3% of the reviewed corpus), confirming that collaborative model development and updating remain the dominant line of research.
- **Data-centric collaboration** focuses on the transmission of raw sensor readings, extracted features, or compressed representations between edge and cloud tiers. The principal objectives are data acquisition optimization, multimodal fusion, preprocessing,

and bandwidth-efficient transmission. Typical techniques include compressed sensing, adaptive sampling, data compression, multimodal fusion, semantic communication, and predictive transmission protocols. This category contains 10 papers (6.8%), reflecting the comparatively smaller but still important body of work on industrial data-flow management.

- **Resource and task-centric collaboration** concentrates on computational task assignments and resource-allocation decisions exchanged across tiers. The main concerns are offloading strategy, workload balancing, energy optimization, and quality-of-service guarantees. Key technologies include task-offloading algorithms, resource scheduling, deep reinforcement learning for dynamic allocation, energy-aware optimization, and coded distributed computing. This category contains 22 papers (15.0%), indicating sustained attention to computational efficiency and service-level control.
- **Architecture and trust-centric collaboration** emphasizes system organization, platform infrastructure, and reliability mechanisms. Here, the objectives include platform design, digital twin integration, microservice orchestration, security, and standardization. Common technologies include digital twin architectures, microservices, containerization, blockchain-based trust schemes, industrial gateways, and unified interfaces. This category contains 34 papers (23.1%), underscoring the importance of system architecture in moving ECC from laboratory demonstrations to industrial deployment.

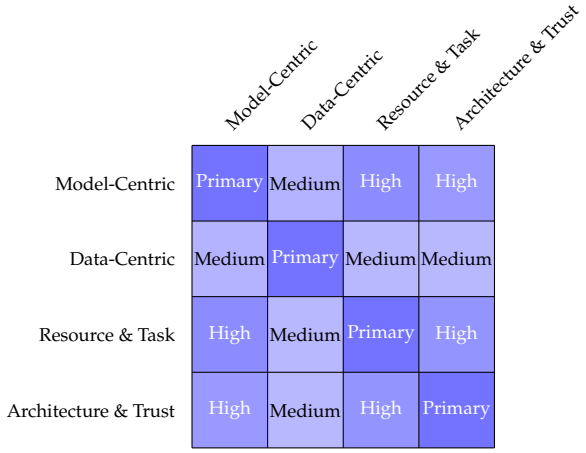



**Figure 4.** Qualitative cross-dimension co-occurrence heat map derived from secondary-label coding.

### *2.3. Relation to Existing Surveys and Reviews*

To contextualize this review within the broader literature landscape, this section systematically examines 13 survey and review papers that intersect with edge–cloud collaboration for machine condition monitoring, identifying their respective scopes, contributions, and the gaps that motivate the present work. Table 2 provides a comparative summary.

Existing surveys provide useful background, but they stop short of a mechanism-level synthesis of ECC for machine condition monitoring. Edge AI reviews by Bala et al. [22], Hemmati et al. [21], Hafeez et al. [4], and Lu et al. [5] clarify where data processing, model training, and deployment occur across the edge–cloud continuum, yet they do not explain collaboration in terms of the exchanged object, coordination logic, or system-level trade-offs. As a result, they are informative for architectural orientation but less effective for comparing concrete ECC design patterns in industrial monitoring.

Reviews centered on condition monitoring and prognostics show a similar limitation. Garcia et al. [25] and Saeed et al. [26] provide broad assessments of signal processing, deep learning, and deployment challenges for intelligent maintenance, while Bao et al. [24] gives a detailed component-specific survey for bearing diagnosis. These studies are valuable within their intended scope, but edge–cloud collaboration is treated either as one imple-

mentation choice among many or within a narrowly bounded application domain. They, therefore, do not provide a cross-mechanism view of how edge and cloud share model, data, scheduling, and infrastructure responsibilities in MCM.

**Table 2.** Comparison of Existing Surveys and Reviews with This Work.

| Reference | Primary Focus | Key Contributions | Limitations for MCM | This Work's Complement |
| --- | --- | --- | --- | --- |
| Bala et al. 2024 [22] | Edge AI for predictive maintenance (PdM) | Edge deployment locations; data reduction techniques; MCM applications | Limited edge–cloud collaboration mechanisms; no taxonomy | Four-mechanism taxonomy; collaboration patterns |
| Hemmati et al. 2024 [21] | Edge AI for Big Data | Systematic review of Edge AI; measurement environment analysis; big data taxonomies | General-purpose; not MCM-specific; no collaboration mechanisms | MCM-specific analysis; collaboration mechanisms |
| Garcia et al. 2025 [25] | Condition monitoring and predictive maintenance (CM-PdM) methods | Natural language processing (NLP)-assisted review; signal processing tools; hybrid models | Edge-cloud architecture not central focus; limited collaboration mechanisms | Architecture-centric collaboration; mechanism-level analysis |
| Liu et al. 2025 [27] | Edge–cloud collaborative computing general survey | Model optimization; AI-driven resource management; security architectures | General computing focus; not MCM-specific; limited MCM applications | MCM domain-specific; 147-paper corpus |
| Hafeez et al. 2021 [4] | Edge Intelligence | Data handling; three processing patterns; novel architecture proposal | Pre-2021; no FL advances; limited DT integration | Comprehensive FL taxonomy; DT collaboration |
| Lu et al. 2023 [5] | Edge Diagnosis for IoT MCM | Edge computing for machine signal processing; IoT fault diagnosis pipeline; deployment constraints | Edge-focused; cloud collaboration not central; no ECC taxonomy | Explicit edge-cloud mechanism taxonomy for MCM |
| Saeed et al. 2025 [26] | Intelligent machinery health monitoring (IMHM) survey | Deep learning (DL) for intelligent fault diagnosis (IFD) and remaining useful life (RUL); edge deployment challenges | No edge–cloud collaboration mechanisms; single-tier focus | Four collaboration mechanisms; hierarchical analysis |
| Kok et al. 2024 [28] | Large language model (LLM)-IoT integration | LLM-IoT system integration; communication protocols | Emerging topic; limited MCM applications; not edge-cloud focused | Traditional ECC mechanisms; comprehensive MCM coverage |
| Bao et al. 2026 [24] | Bearing fault diagnosis | Deep learning for bearing fault diagnosis (FD); DT integration; edge-cloud architectures | Single component focus; limited collaboration mechanisms | Multi-component MCM; mechanism taxonomy |
| Xiao et al. 2024 [29] | Digital twin-driven prognostics and health management (DT-PHM) | DT reference architectures; tiered management; PHM applications | DT-centric only; no comprehensive ECC mechanism coverage | DT as one of four mechanisms; holistic view |
| Xu et al. 2025 [30] | Edge deep learning | Edge DL for computer vision (CV) and medical applications; hardware platforms | Domain focus outside MCM; no collaboration mechanisms | MCM-specific; collaboration mechanisms |
| Xia et al. 2026 [31] | Embodied intelligence | Embodied AI in manufacturing; large multimodal models (LMMs); cloud-edge-end collaboration | Emerging paradigm; limited MCM applications; not collaboration-focused | Traditional ECC mechanisms; MCM-specific analysis |
| Soldani and Brogi 2022 [23] | Cloud Fault Diagnosis Survey | Anomaly detection and root-cause analysis in service-based cloud applications | Cloud service reliability focus; not MCM-specific; no edge-cloud mechanism taxonomy | Industrial asset monitoring focus; ECC mechanisms for PHM |

Broader ECC surveys and digital-twin-oriented reviews extend the perspective without fully resolving this gap. Liu et al. [27] survey general edge–cloud collaborative computing,

but their discussion is not tailored to the sensing, latency, and reliability requirements of industrial condition monitoring. Xiao et al. [29] offer a strong review of digital twin-driven PHM, yet digital twins represent only one collaboration mechanism. Soldani and Brogi [23] provide a useful adjacent survey of anomaly detection and root-cause analysis in service-based cloud applications, but their focus remains cloud-service fault diagnosis rather than machine condition monitoring across edge–cloud industrial hierarchies. Emerging surveys on large language model (LLM)-IoT integration, edge deep learning outside MCM, and embodied intelligence in manufacturing [28,30,31] are useful for understanding future directions, but they do not yet provide a stable basis for structuring the mature ECC literature for machine monitoring applications.

Against this background, the present review is positioned as an MCM-specific, mechanism-centered synthesis. It examines 147 publications, develops a four-dimensional taxonomy for the full review corpus, and integrates application evidence across multiple industrial sectors. In contrast to prior surveys, the contribution here lies not only in breadth but also in the explicit comparison of what is exchanged between tiers, why that exchange is needed, and how it shapes performance, deployability, and trustworthiness in practical ECC systems. Building on this taxonomy and comparative baseline, the next section examines each collaboration mechanism in technical detail.

## 3. Edge–Cloud Collaboration Mechanisms

Building on Section 2, this section analyzes the four collaboration mechanisms through representative works, technical approaches, and performance characteristics.

### *3.1. Model-Centric Collaboration*

Model-centric collaboration is the most intensively studied ECC paradigm, accounting for 68 papers (46.3% of the reviewed corpus). These studies focus on the exchange of model parameters, gradients, or knowledge representations between edge and cloud tiers in order to balance diagnostic accuracy, inference latency, and resource efficiency. Four recurrent patterns appear in the literature: cloud training with edge inference, collaborative training through federated learning, knowledge transfer through distillation and transfer learning, and split execution across the edge–cloud hierarchy. Figure 5 visualizes the common workflow logic behind the dominant model-centric mechanisms. Table 3 summarizes the main variants.

#### 3.1.1. Cloud Training with Edge Inference

The most fundamental collaboration pattern involves training models on cloud servers where computational resources are abundant, followed by deploying compressed or lightweight variants to edge devices for low-latency inference. This pattern addresses scenarios where edge devices possess sufficient computational capability for inference but lack the resources for large-scale model training, and where model updates can be delivered during maintenance windows or opportunistically over persistent connections.

Foundational work by Calabrese et al. [32] established SOPHIA, an event-based IoT architecture employing tree-based ensemble models (Gradient Boosting, Random Forest, XGBoost) for remaining useful life prediction in woodworking machinery, demonstrating that cloud-side feature engineering from terabytes of log data could support edge-side real-time screening. Their subsequent work [2] operationalized this paradigm through an edge-cloud PHM infrastructure in which edge nodes perform streaming analysis for novelty detection while cloud servers handle comprehensive model training using accumulated historical data, thereby addressing the common industrial scenario in which labeled fault data are initially unavailable.

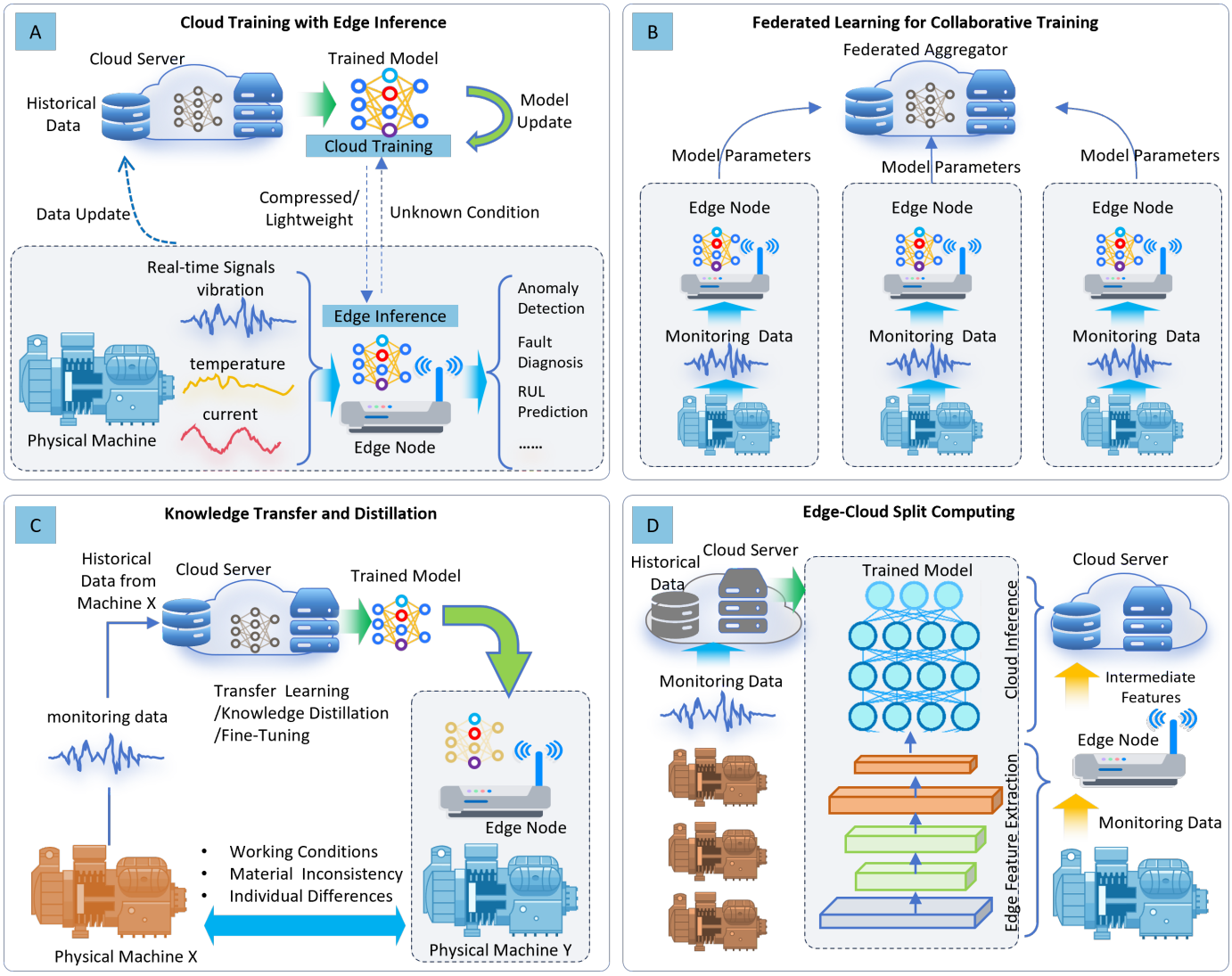



**Figure 5.** Main model-centric edge–cloud collaboration patterns for machine condition monitoring: (**A**) cloud training with edge inference; (**B**) federated learning for collaborative training; (**C**) knowledge transfer and distillation; (**D**) edge–cloud split computing.

A parallel research line has focused on lightweight neural architectures optimized for edge inference under resource constraints. Wang et al. [33] deployed an improved You Only Look Once (YOLO) v3-tiny for photovoltaic defect detection, incorporating residual modules to achieve 99.78% accuracy while reducing training time by 66%. You et al. [34] proposed AH_TCNet, a lightweight tool condition monitoring network featuring adaptive activation functions and hard swish operators optimized through cloud-side architecture search. Li et al. [35] developed LiteDiagNet, employing multi-scale convolutions with only 12.9K parameters to achieve 99.8% accuracy under noisy industrial conditions. Wu et al. [9] introduced a quantized subtraction-convolution network that combines a learnable sparse pulse kernel with sparse pulse quantization and quantization-aware training, achieving 99.88% accuracy with a 9.5× compression ratio.

Alternative designs based on dictionary learning offer more interpretable and incrementally updatable models for process monitoring. Huang et al. [36] proposed an error-triggered dictionary learning framework in which the edge monitors process states locally and use label-error-based mismatch detection to trigger cloud-side model retraining when time-varying conditions degrade model consistency. Their subsequent work [8] extended this idea through self-updating dictionary learning with distillation compres-

sion and hybrid encryption, enabling secure collaborative monitoring in privacy-sensitive industrial environments.

More recent studies have begun to explore foundation-model-style strategies for industrial diagnosis. Lai et al. [37] pioneered BearingFM, a bearing-specific foundation model pre-trained through contrastive learning on large-scale unlabeled vibration data, achieving 98% accuracy with only 1.2% labeled samples via cloud-edge-end semi-supervised fine-tuning and illustrating the potential for knowledge transfer across bearing types and operating conditions.

Researchers have also examined model selection and adaptive deployment rather than fixed architecture design alone. Wang and Li [38] developed a cloud-side model pre-training pool that enables edge nodes to retrain and select suitable variants while reducing local training difficulty and shortening training time by 66%. Sui et al. [39] proposed ISDMM, an interpretable switching deep learning model that avoids unnecessary retraining by selectively activating appropriate model components based on process-state recognition. Ren et al. [40] introduced LM-CNN with label sampling space enlarging for adaptive fault diagnosis under data imbalance, where cloud-side label expansion and weighted sampling support the deployment of scene-specific edge diagnostic models for heterogeneous equipment. Chen et al. [14] likewise used cloud-assisted model remodeling for infrared electrical-equipment diagnosis: Jetson-based edge devices perform on-site detection, upload only unknown or misdetected cases for cloud-side updating, and then receive refreshed models, yielding about 20% improvement in mean average precision and about 83% reduction in backbone-network bandwidth.

These design patterns have been validated in domain-specific deployments as well. Song et al. [41] developed a cloud-edge collaborative insulator detection method for power infrastructure in which unmanned aerial vehicle (UAV)-side edge computation performs direction pre-estimation to reduce computational load, enabling real-time defect detection during aerial inspections. Tang et al. [42] implemented linear defect detection for large-scale photovoltaic plants using convolutional neural network (CNN)-based models trained in the cloud and deployed for real-time inference at distributed edge nodes. Chen et al. [43] further demonstrated this training-deployment pattern in smart surveillance using an edge-triggered cloud update loop for unknown objects, reporting about 94% backbone bandwidth reduction while maintaining real-time edge inference.

**Table 3.** Summary of Model-Centric Collaboration Approaches in Edge-Cloud MCM (68 papers).

| Main Approach | Subcategory | Ref. | Key Contribution |
|---|---|---|---|
| Cloud Training, Edge Inference (Pattern 1.1) | Secure Dictionary Learning | [8] | Self-updating dictionary learning with distillation compression for edge deployment and hybrid encryption. |
| | UAV Defect Detection | [41] | Cloud-edge insulator detection with edge-based direction pre-estimation to reduce UAV computational load. |
| | LM-CNN Deployment | [40] | Label sampling space enlarging for adaptive fault diagnosis under data imbalance. |
| | Error-Triggered Dictionary | [36] | Dictionary learning with model simplification and label-error-triggered cloud updating for time-varying process monitoring. |
| | LiteDiagNet | [35] | Multi-scale convolution achieving 99.8% accuracy with only 12.9K parameters for edge deployment. |
| | CNN Defect Detection | [42] | CNN-based defect detection where cloud trains models and edge runs inference. |
| | Quantized Network | [9] | Quantized subtraction-convolution network with 99.88% accuracy and 9.5× compression ratio. |
| | PV Object Detection | [33] | Improved YOLO v3-tiny with cloud-based training and edge deployment via transfer learning. |
| | Lightweight AH_TCNet | [34] | Lightweight network with adaptive activation functions for tool condition monitoring. |
| | ISDMM | [39] | Switching variables and model update strategy to avoid unnecessary retraining. |
| | Foundation Model | [37] | BearingFM using contrastive learning; 98% accuracy with 1.2% labels. |
| | Event-Based IoT Predictive Maintenance | [32] | SOPHIA event-based IoT architecture with tree-based machine learning (ML) ensemble for predictive maintenance. |
| | Semi-Supervised PHM | [2] | Semi-supervised PHM framework using edge-cloud infrastructure with streaming analysis. |
| | Model Pool + Edge Retraining | [38] | Cloud-side model pre-training pool with edge-side retraining and selection, reducing training time by 66%. |
| | Active Cloud Update | [14] | Edge-side infrared diagnosis with cloud-side remodeling of unknown cases; about 20% mAP gain and about 83% bandwidth reduction. |
| | Real-Time Object Detection | [43] | Distributed real-time object detection with edge–cloud collaboration for smart video surveillance. |

**Table 3.** *Cont.*

| Main Approach | Subcategory | Ref. | Key Contribution |
|---|---|---|---|
| Collaborative Training—Federated Learning (Pattern 1.2) | Standard FL | [6] | Comprehensive review of FL deployment in cloud-edge architecture. |
| | Standard FL | [44] | First FL-based RUL prediction using CAE and quality-aware weighted aggregation. |
| | Standard FL | [45] | Adaptive weighted selection for global CAE feature extraction. |
| | Standard FL | [15] | Branch model allowing edge nodes to select partial models, reducing communication by 40%. |
| | Standard FL | [46] | Secure FL for smart grid with DRL-based incentive mechanisms. |
| | Dynamic Aggregation | [47] | Rademacher complexity-guided dynamic aggregation for non-IID data. |
| | Hierarchical FL | [48] | Joint optimization of aggregation frequency and resource allocation. |
| | Hierarchical FL | [11] | Multi-level FL with level-by-level parameter sharing and Kalman filtering aggregation. |
| | Hierarchical FL | [49] | Asynchronous quasi-cloud-edge-client FL with Sequential Kalman Filter fusion. |
| | Event-Triggered FL | [50] | Event-triggered FL reducing communication by 63% for wind turbine diagnosis. |
| | Two-Phase FL | [16] | Cloud-edge collaborative GAN with mixed focal loss function. |
| | FL + Continual Learning | [51] | MeCo for federated continual learning with meta task-invariant knowledge consolidation. |
| | FL + Domain Generalization | [52] | FDG-CM framework for privacy-preserving condition monitoring. |
| | FL + Domain Generalization | [53] | Progressive training with iterative weight updates for federated fault diagnosis. |
| | FL + Domain Generalization | [54] | Intra/inter-domain gradient alignment for domain generalization. |
| | FL + Transfer Learning | [55] | Adaptive fine-tuning with Gumbel-softmax and double encryption. |
| | FL + Transfer Learning | [56] | Transfer learning combined with FL for small-sample imbalanced faults. |
| | FL + Transfer Learning | [57] | Federated fine-tuning and target self-adaptation for cross-domain fault diagnosis. |
| | FL + Zero-Shot Learning | [58] | Federated zero-shot diagnosis with cloud-shared semantic knowledge base. |
| | FL + Episodic Learning | [59] | Dual-level normality bank for extrapolating unseen working conditions. |
| | FL + Knowledge Amalgamation | [60] | Fault attributes as unbiased knowledge for federated fault diagnosis. |
| | FL + Knowledge Amalgamation | [61] | Uploading diagnostic knowledge instead of model parameters. |
| | FL + Model-Agnostic | [62] | Architecture-independent knowledge carriers for heterogeneous FL. |
| | FL + Compressive Learning | [63] | Compressive sampling for dimensionality reduction in FL. |
| | FL + Split Learning | [64] | Cloud-edge-client model splitting with DT-assisted DRL resource scheduling. |
| | FL + RUL Prediction | [65] | Two-stage FL where edge extracts features and cloud trains RUL model. |
| | FL + Knowledge Distillation | [66] | Three-layer FRL with dual-RL optimization and bi-distillation. |
| | FL + Resource Optimization | [67] | Elastic local update and n-softsync aggregation for personalized models. |
| Knowledge Transfer—Distillation & TL (Pattern 1.3) | DRL Distillation | [68] | OD3 method for on-device DRL with distillation to compress policy knowledge. |
| | KD for Training | [13] | Cloud-edge training using KD to transfer inference from Cloud-NN to Edge-NN. |
| | Spatial Alignment KD | [69] | Feature-aligned incremental learning with spatial alignment-based KD. |
| | Multidimensional KD | [70] | Measurement point optimization and feature compression via KD. |
| | TL with VAE | [71] | Multiscale convolution and VAE-based domain fusion. |
| | TL with Adversarial | [72] | Adversarial transfer learning (DANN-DCGAN) for cross-domain diagnosis. |
| | Graph Domain Adaptation | [73] | Adaptive multihop graph adaptation with pseudolabel-guided edge-cloud orchestration for cross-condition monitoring. |
| | TL + Knowledge Graph | [74] | SAES-DNN with transfer learning and knowledge graph visualization. |
| | TL for Rapid Diagnosis | [75] | DSCNN-GAP and DS evidence theory for rapid fault diagnosis. |
| | Parameter Transfer for RUL | [76] | RVM-based battery RUL prediction with horizontal parameter transfer reducing training samples to 30%. |
| | TL for HIF Detection | [77] | Transfer learning-based high impedance fault detection. |
| | Offline-Transfer-Online | [78] | OTO framework accelerating edge DRL training via KD. |
| | Imbalance-Aware CET | [79] | Monitor-based device-free sensing under class imbalance with gradient-aware loss correction. |
| | Deep Mutual Learning | [80] | DML-based fault diagnosis with mutual learning between ResNet models. |
| Edge-Cloud Split Computing (Pattern 1.4) | Feature Extraction at Edge | [81] | Edge: EMD + feature extraction; cloud: 1D-ResNet classification. |
| | Edge-Cloud Split CNN | [82] | Cloud-PE with deep model; edge-PE with shallow model for RUL prediction. |
| | Transformer-CNN | [83] | Swin Transformer + multiscale CNN with 99.55% precision. |
| | GA-Att-LSTM | [84] | GA-optimized attention LSTM for real-time IIoT fault detection. |
| | Contrastive Learning | [85] | Self-correcting-guided contrastive learning for PV fault diagnosis. |
| | Multi-model Ensemble | [86] | IOWA operator for multi-model weight allocation. |
| | Knowledge Adaptation | [87] | Multi-teacher distillation for continuous learning without forgetting. |
| | Feature-Guided | [88] | Feature performance-guided distributed model for surface roughness monitoring. |
| | Hierarchical KPI | [89] | Hierarchical fault detection with KPI monitoring. |
| | Dual-Scale SFA | [90] | Dual-scale slow feature analysis for KPI-related diagnosis. |

#### 3.1.2. Federated Learning for Collaborative Training

Federated learning has emerged as the predominant paradigm for privacy-preserving collaborative training across edge-cloud hierarchies, with 28 papers addressing various aspects of this approach. The fundamental appeal of FL in industrial settings stems from its ability to aggregate knowledge from geographically distributed machines without requiring raw data transmission, thereby addressing both privacy concerns and network bandwidth constraints. The reviewed literature reveals a rich taxonomy of FL variants spanning standard implementations, hierarchical architectures, communication-efficient schemes, and advanced integration with complementary learning paradigms.

Standard FL implementations established the baseline for collaborative training in edge–cloud MCM. Bao and Guo [6] outlined the key challenges and application scenarios, and subsequent domain-specific studies demonstrated how those principles translate into monitoring systems. Guo et al. [44] proposed FedRUL, integrating convolutional

autoencoders for remaining useful life prediction with quality-aware weighted aggregation to handle data heterogeneity across clients. Yu et al. [45] developed FedCAE with adaptive weighted selection for global convolutional autoencoder feature extraction, improving representation quality across non-independent and identically distributed (non-IID) client distributions. Wang et al. [15] introduced a branch neural network architecture that enables edge nodes to select partial model components based on local computational resources, achieving 40% reduction in communication overhead while maintaining diagnostic performance.

As deployments grew in scale, researchers proposed hierarchical and asynchronous FL architectures to reduce straggler effects and central bottlenecks. Su et al. [48] jointly optimized aggregation frequency and resource allocation, showing that adaptive scheduling can balance convergence speed against communication and computational cost. Ma et al. [11] developed a multi-level FL framework with level-by-level parameter sharing and Kalman filtering aggregation, while Ma and Wen [49] proposed an asynchronous quasi-cloud-edge-client architecture that allows edge nodes to update without waiting for slow clients.

Communication-efficient FL remains essential when devices operate over bandwidth-constrained links. Lu et al. [50] reduced wind-turbine communication by transmitting model updates only when parameter variation or accuracy gain exceeds adaptive thresholds, achieving 63% communication reduction without degrading diagnostic performance. Xiao et al. [47] developed FedRad, which uses Rademacher-complexity-guided aggregation to prioritize informative updates. Gao et al. [63] reduced per-round payload through compressive sampling, and Gao et al. [67] proposed elastic local updates with n-softsync aggregation for personalized models. Zhang et al. [16] addressed class imbalance and communication jointly through a two-phase federated framework with collaborative generative augmentation.

Another active line of work extends FL toward domain generalization and low-data recognition. Eslaminia et al. [52], He and Shen [53], and Ye et al. [54] all target cross-condition robustness by learning domain-invariant or gradient-aligned representations under distributed training. Liang et al. [55], Sima et al. [56], Wang et al. [57], and Li and Zhao [58,59] further integrate transfer, zero-shot, and episodic-learning ideas so that clients can recognize rare or previously unseen operating conditions without access to full centralized data. In particular, Wang et al. [57] combined federated fine-tuning with target self-adaptation to mitigate cross-client distribution shift, and reported a 0.45% gain over a centralized deep transfer-learning baseline.

Heterogeneous models and complementary learning paradigms have also widened the FL design space. FedKA and FedMAKE replace simple parameter averaging with knowledge carriers that can be exchanged across different architectures [60,62], and Jiang et al. [61] showed that uploading diagnostic prototypes or decision knowledge can outperform raw-parameter exchange in wind-turbine diagnosis. Other studies combine FL with RUL prediction, digital twins, split learning, continual learning, or incentive design [46,51,64–66], indicating that FL is increasingly used as a coordination substrate rather than a standalone training algorithm.

#### 3.1.3. Knowledge Transfer via Distillation and Transfer Learning

Knowledge transfer techniques enable the migration of learned representations, decision boundaries, or policy knowledge between models operating at different tiers of the edge-cloud hierarchy. This category encompasses 14 papers that leverage knowledge distillation (KD) and transfer learning (TL) to address the computational asymmetry between cloud training infrastructure and edge deployment constraints, as well as the domain shift between historical training data and operational deployment conditions.

Knowledge distillation remains a practical route to edge deployment because it transfers knowledge from large cloud-hosted teacher models to compact student models without requiring a full redesign of the diagnostic pipeline. Jang et al. [68] developed OD3 for on-device deep reinforcement learning, Ji et al. [69] proposed spatially aligned distillation for incremental learning, Shi et al. [13] established a confidence-aware cloud-edge distillation framework, and Wang et al. [70] combined measurement-point optimization with multidimensional distillation. He et al. [80] further showed that deep mutual learning can make this exchange bidirectional, improving both cloud and edge models.

Transfer learning addresses a different but related problem: adapting cloud-trained knowledge to data-scarce target domains at the edge. Li et al. [71] used multiscale variational domain fusion for real-time edge diagnosis, Zou et al. [72] employed adversarial transfer learning for unlabeled target domains, and Zhang et al. [74,77] demonstrated interpretable or task-specific transfer in railway and power-distribution settings. Tang et al. [75] showed that cloud pretraining followed by light edge fine-tuning can significantly reduce adaptation time while maintaining diagnostic accuracy. Zhou et al. [76] extended transfer to battery prognostics by combining relevance vector machine (RVM)-based RUL prediction with horizontal parameter transfer, reducing the sample requirement for model learning to 30% while preserving probabilistic prediction quality. Liu and Chen [73] addressed cross-condition condition monitoring through a graph adaptation framework in which adaptive multihop feature aggregation, metric alignment, and pseudolabel-guided edge–cloud orchestration reduce distribution mismatch between source and target operating states.

Offline-transfer-online strategies connect these ideas to operational deployment. Zeng et al. [78] proposed the OTO framework, which accelerates edge deep-reinforcement-learning adaptation by distilling knowledge from cloud-side offline training. Zhou et al. [79], by contrast, addressed class-imbalance in cloud-edge-terminal device-free sensing through a monitor mechanism that tracks gradient changes, infers class proportions, and applies loss correction during collaborative training.

#### 3.1.4. Edge-Cloud Split Computing

Edge-cloud split computing partitions neural networks across the computational hierarchy, executing early layers at edge nodes for feature extraction and later layers at cloud servers for classification or prediction. This paradigm addresses scenarios where complete model inference at the edge is computationally prohibitive, while uploading raw sensor data to the cloud introduces unacceptable latency or bandwidth consumption. The split computing approach represents a middle ground: edge devices perform computationally lightweight feature extraction and preliminary analysis, transmitting compact intermediate representations to the cloud for resource-intensive processing.

The canonical split-computing pattern assigns signal preprocessing and feature extraction to the edge while reserving deep inference or long-horizon prediction for the cloud. Fan et al. [81] used this strategy for rotor-bearing diagnosis by performing empirical mode decomposition and time-frequency feature extraction locally before cloud-side 1D-ResNet classification. Jing et al. [82] paired a shallow edge predictor with a deeper cloud model for RUL estimation, and Sun et al. [83] partitioned a Swin-Transformer-CNN pipeline across the hierarchy for shearer rocker gear diagnosis.

More elaborate split-computing variants rely on coordinated predictors or adaptive feedback across the partitioned pipeline rather than a single fixed cut point. Sun et al. [86] used multi-model adaptive estimation for battery prognosis, and Jia and Ren [87] showed that multi-teacher distillation can refresh edge-side decision modules while preserving responsiveness under changing conditions. Real-time and key performance indicator (KPI)-guided variants follow the same principle, using cloud feedback to refine edge-side

feature extraction and decision logic in milling, quality monitoring, and photovoltaic diagnosis [84,85,88–90].

### *3.2. Data-Centric Collaboration*

Data-centric collaboration addresses the fundamental challenge of managing massive industrial sensor data streams across edge-cloud hierarchies under bandwidth, latency, and storage constraints. This category encompasses 10 papers (6.8% of the corpus) focusing on the transmission of data streams—including raw sensor readings, extracted features, or compressed representations—between edge and cloud tiers. The primary collaboration objectives encompass data acquisition optimization, multimodal fusion, bandwidth-efficient transmission, and intelligent data selection. Unlike model-centric approaches that transmit parameters or gradients, data-centric collaboration centers on what data to transmit, when to transmit, and how to compress or transform data for efficient uplink. Table 4 provides a comprehensive summary.

**Table 4.** Summary of Data-Centric Collaboration Approaches in Edge-Cloud MCM (10 papers).

| Main Approach | Subcategory | Ref. | Key Contribution |
|---|---|---|---|
| Data Compression & Selective Transmission (Pattern 2.1) | QDCS-Net Compression | [91] | Quantized layers and dual-path structure for 1/128 compression ratio. |
| | Novelty Detection | [92] | ANN novelty detector reducing data transmission by 94% for power electronics. |
| | Joint Optimization | [93] | Joint optimization of sampling rate and preprocessing with Lyapunov optimization. |
| | DPS-CE | [94] | Synchronized prediction models reducing transmission by 88.66%. |
| | Variable Frequency Upload | [95] | Variable frequency data upload with Apriori data mining. |
| Multimodal Data Fusion (Pattern 2.2) | Self-Adaptive Hybrid | [96] | Self-adaptive hybrid model using vibration and noise multimodal fusion. |
| | MMDT-IAD | [97] | Multi-modal DT integrating geometric, visual, physical, and semantic modalities. |
| | Spatio-Temporal Analytics | [98] | TKCCA and SCCA for spatio-temporal feature extraction in industrial monitoring. |
| | CNN-Based Data Fusion | [99] | CNN-based real-time data fusion with genetic algorithm routing optimization. |
| Data Pipeline & Architecture (Pattern 2.3) | IntelliPdM Framework | [100] | Heterogeneous data streams; synthetic data generation; 25–30% cost reduction. |

#### 3.2.1. Data Compression and Selective Transmission

The proliferation of industrial sensors has created unprecedented data volumes that overwhelm conventional network infrastructure. Raw sensor data transmitted from edge to cloud for centralized processing imposes substantial bandwidth costs, introduces transmission latency, and creates storage burdens at cloud data lakes. Data compression and selective transmission strategies address these challenges by intelligently reducing data volume while preserving diagnostically relevant information.

End-to-end compression frameworks are a major theme in data-centric ECC. Zhang et al. [91] proposed QDCS-Net, a quantized deep compressed-sensing framework that learns sensing, compression, and reconstruction jointly and achieves a 1/128 compression ratio while preserving useful signal structure for cloud-side reconstruction.

Selective transmission is equally important. Bhoi et al. [92] developed a novelty detector for power-electronics monitoring that transmits only diagnostically informative deviations from normal operation, reducing transmission by 94%. Wu et al. [94] proposed a dual-prediction scheme in which synchronized edge and cloud predictors suppress communication until prediction error exceeds an adaptive threshold, reducing transmission by 88.66% while maintaining bounded reconstruction accuracy. Shi et al. [93] moved one level higher by jointly optimizing sampling rate, preprocessing mode, and resource allocation through Lyapunov optimization, thereby reducing both energy and delay.

Traditional fixed-frequency upload schedules also ignore the uneven information density of industrial signals. Erbao et al. [95], therefore, proposed variable-frequency upload guided by Apriori data mining to identify high-value segments, illustrating that

selective transmission can be improved not only by signal compression but also by event-aware upload policies that better align communication effort with diagnostic value.

#### 3.2.2. Multimodal Data Fusion and Alignment

Industrial monitoring increasingly employs multiple sensor modalities—vibration, acoustic, thermal, visual—to capture complementary aspects of machine health. Multimodal fusion at the edge enables compact cross-modal representations that preserve diagnostic information while reducing transmission volume, and supports more robust diagnosis when individual modalities experience noise or degradation.

Digital twin architectures provide a natural basis for multimodal integration. Ren et al. [97] proposed MMDT-IAD, which combines geometric, visual, physical, and semantic modalities through cross-modal fusion so that aligned multimodal features can be transmitted compactly from the edge to cloud twins. Sheng et al. [96] developed a self-adaptive fusion scheme for machining quality prediction that weights vibration and acoustic signals according to signal quality, while Zhang et al. [98] used canonical-correlation-based methods to capture spatio-temporal structure across sensor arrays. At the infrastructure level, Zhou [99] showed that protocol-aware real-time fusion pipelines can substantially reduce end-to-end latency by handling heterogeneous industrial protocols before cloud transmission.

#### 3.2.3. Data Pipeline and Processing Architecture

Comprehensive data management frameworks address the full lifecycle from acquisition through transmission to cloud storage and analysis. Somu and Dasappa [100] proposed IntelliPdM, an edge-cloud IIoT framework handling heterogeneous data streams with synthetic data generation capabilities, achieving 25–30% cost reduction through intelligent data orchestration. The framework integrates edge-side data validation, anomaly-triggered transmission, and cloud-side synthetic sample generation to address data scarcity in rare fault categories.

### *3.3. Resource and Task-Centric Collaboration*

Resource and task-centric collaboration addresses the fundamental challenge of optimally distributing computational workloads, allocating limited resources, and scheduling tasks across edge-cloud hierarchies under strict latency, energy, and reliability constraints. This category encompasses 22 papers (15.0% of the corpus) focusing on computational task assignments and resource allocation instructions transmitted between tiers. Unlike data-centric approaches that optimize what to transmit, resource and task-centric collaboration determines where to compute and when to offload. The primary collaboration objectives include task offloading decisions, workload balancing, energy optimization, and quality-of-service (QoS) guarantees. Table 5 provides a comprehensive summary.

**Table 5.** Summary of Resource and Task-Centric Collaboration Approaches in Edge-Cloud MCM (22 papers).

| Main Approach | Subcategory | Ref. | Key Contribution |
| --- | --- | --- | --- |
| Dynamic Task Offloading (Pattern 3.1) | DRL-Based | [101] | DQN + convex optimization for multi-task offloading, 7–73% delay reduction. |
| | DRL-Based | [102] | Attention mechanism for high-dimensional multi-objective offloading. |
| | DRL-Based | [103] | Multi-agent DRL with caching system for Internet of Vehicles. |
| | Approximate Layer Allocation | [104] | Greedy and recursion-based approximate allocation of DNN layers to minimize edge-cloud inference latency. |
| | Heuristic-Based | [19] | Two task offloading algorithms targeting fastest response and load balancing in digital twin-based machine tools. |
| | Game-Theoretic | [105] | Stackelberg game for DT-driven computation offloading. |
| | Container Update | [106] | RL-based update decisions with layer sharing. |
| | Heuristic + DL | [107] | Computation offloading for metro integrated supervision system. |
| | MADDPG | [108] | DT-assisted joint optimization of device association, offloading, and bandwidth. |
| | DRL-Based | [109] | Collaborative clustering parallel RL for DT resource scheduling in manufacturing. |
| | Hybrid Framework | [110] | Hybrid edge-cloud framework with KNN (edge) and LSTM (cloud) for predictive maintenance. |
| | Edge-Based | [20] | Edge-based framework for real-time prognosis and opportunistic maintenance. |

**Table 5.** *Cont.*

| Main Approach | Subcategory | Ref. | Key Contribution |
|---|---|---|---|
| Resource Scheduling & Management (Pattern 3.2) | DRL-Based | [111] | Multi-dimensional resource management for DT-enabled IIoT. |
| | Workload Forecasting | [112] | ARMA-ENN forecasting with adaptive threshold-based migration. |
| | Two-Timescale | [113] | Two-timescale optimization for HDT deployment under mobility. |
| | HGNN + CDC | [114] | Hypergraph neural networks with coded distributed computing for resource scheduling. |
| | Risk-Sensitive RL | [115] | Risk-sensitive RL for collaborative computing in train-edge-cloud systems. |
| Energy-Aware Optimization (Pattern 3.3) | Genetic Algorithm | [116] | PoE-based energy and delay-aware offloading for DT edge networks. |
| | Multi-Objective | [117] | Fractal NSGA-II for latency-energy-aware task offloading. |
| Task Partitioning (Pattern 3.4) | Bio-Inspired | [118] | Edge: anomaly detection; cloud: CAE-based fault classification. |
| | Optical Network | [119] | Control-layer AI and on-board AI collaboration for optical networks. |
| | MILP + RL | [120] | MILP task partitioning with RL-based offloading. |

#### 3.3.1. Dynamic Task Offloading

Dynamic task offloading represents the central problem in resource and task-centric collaboration: deciding which functions should remain local, which should move to nearby edge servers, and which should be escalated to the cloud under latency, bandwidth, energy, and reliability constraints. In industrial MCM, this decision rarely concerns inference alone. It often spans prognosis, maintenance scheduling, and service deployment across heterogeneous assets. Zhang et al. [20], for example, moved real-time prognosis to edge nodes while reserving opportunistic maintenance scheduling for the cloud, showing that workflow-level partitioning can outperform cloud-only coordination. Sathupadi et al. [110] adopted a related hybrid strategy for predictive maintenance by assigning anomaly screening to an edge-side K-nearest neighbors model and cloud-side failure forecasting to a long short-term memory (LSTM) model under dynamic workload management. Related studies formulated offloading more explicitly for shared industrial infrastructure. Wang et al. [19] proposed two digital-twin-enabled offloading strategies aimed at fastest response and load balancing under edge resource constraints, whereas Li et al. [105] formulated computation offloading as a Stackelberg game in a cloud-edge-end manufacturing system. Hu et al. [104] instead formulated a latency-minimum DNN layer-allocation problem and solved it with an approximate resource-aware scheduling algorithm across the edge–cloud hierarchy.

Because operating conditions, workloads, and network states vary continuously, adaptive learning has become a dominant design choice. Tao et al. [101] combined deep Q-networks with convex optimization for multi-task offloading, Wang and Yang [102] introduced attention-based recurrent policies for high-dimensional decision spaces, and Zhang et al. [103] extended the problem to vehicle–road cooperation through multi-agent deep reinforcement learning. Cooperative formulations remain important when infrastructure is shared across multiple agents: Wang et al. [108] and Yang et al. [109] used multi-agent coordination for end-edge-cloud allocation, whereas Cui et al. [106] optimized layer-aware container updates and Wang et al. [107] tailored offloading to the stringent timing requirements of metro supervision systems. Taken together, these studies show that dynamic offloading in ECC is evolving from isolated task placement toward context-aware orchestration of prediction, scheduling, and service execution.

#### 3.3.2. Resource Scheduling and Management

Beyond individual task offloading decisions, system-level resource scheduling coordinates the allocation of multiple resource types—computational, communication, and storage—across the edge-cloud hierarchy over extended time horizons. Resource scheduling work extends beyond one-shot offloading to the continuous management of compute, communication, and storage. Hu et al. [111] proposed D4PG, a DDPG variant with double actors and double critics, to jointly manage CPU cycles, DT-model resources, and communication bandwidth in DT-enabled IIoT systems, while introducing a synchronization threshold to reduce straggler-induced delay during distributed learning. Liu et al. [112] introduced proactive scheduling through workload forecasting and

threshold-based migration, while Wang and Zhu [114] combined hypergraph neural networks with coded distributed computing to improve fault tolerance. Yang et al. [113] addressed the different time horizons of deployment and scheduling through a two-timescale optimization framework. Zhu et al. [115] further incorporated risk-sensitive reinforcement learning in train-edge-cloud systems, explicitly penalizing edge overload and delay violations during task scheduling for safety-critical transport scenarios.

#### 3.3.3. Energy-Aware Optimization

Mobile and remote edge devices often operate on limited energy budgets, making energy efficiency a critical optimization objective alongside latency and throughput. Energy-aware optimization is increasingly important in remote or battery-powered deployments. Lin et al. [117] explored latency–energy trade-offs with a hierarchical NSGA-II framework, while Bozkaya et al. [116] coupled energy-aware offloading with blockchain-based auditability in digital twin edge networks. Together, these studies show that energy-aware ECC design increasingly combines classical multi-objective optimization with trust and accountability constraints.

#### 3.3.4. Task Partitioning Strategies

Task partitioning determines how to decompose complex computational tasks into subtasks that can be distributed across edge-cloud infrastructure, considering both computational requirements and data dependencies. Task partitioning studies make the edge–cloud split explicit at the algorithmic level. Miao et al. [118] separated rapid anomaly screening from cloud-side CAE-based fault classification, while Li et al. [119] and Zhou et al. [120] showed that domain-specific partitioning rules remain necessary when optical-network control loops or computer numerical control (CNC) deployment constraints impose strict timing and coordination requirements.

### *3.4. Architecture and Trust-Centric Collaboration*

Architecture and trust-centric collaboration address system-level concerns, including platform design, digital twin frameworks, microservices orchestration, security protocols, and standardization across edge-cloud deployments. This category encompasses 34 papers (23.1% of the corpus) emphasizing the transmission of infrastructure configuration, service organization instructions, and reliability mechanisms between tiers. The primary collaboration objectives encompass platform architecture design, digital twin synchronization, service orchestration, trust establishment, and interoperability assurance. Table 6 provides a comprehensive summary of the reviewed works in this category.

#### 3.4.1. Digital Twin Enabled Collaboration

Digital twin (DT) architectures represent the predominant pattern within architecture and trust-centric collaboration, with 18 papers addressing various aspects of DT deployment across edge-cloud hierarchies. Digital twins create virtual representations of physical assets that synchronize state, behavior, and context in real time, enabling simulation, prediction, and optimization capabilities that would be computationally prohibitive at the edge alone. The core architectural logic is the asymmetric distribution of twin fidelity: cloud-side twins retain high-fidelity models for system-level reasoning, whereas edge-side twins support time-critical estimation, control, and adaptation close to the monitored asset.

**Table 6.** Summary of Architecture and Trust-Centric Collaboration Approaches in Edge-Cloud MCM (34 papers).

| Main Approach | Subcategory | Ref. | Key Contribution |
|---|---|---|---|
| Digital Twin Enabled Collaboration (Pattern 4.1) | DT Clock Sync | [121] | DT-enabled clock synchronization reducing communication overhead. |
| | DT Bionics | [122] | Biological evolution-based DT for product-development-production co-evolution. |
| | Process DT | [123] | Three collaboration patterns for remote process tracking. |
| | Battery DT | [18] | Long short-term memory (LSTM)-based state of health (SOH) prediction with SP + electrochemical model. |
| | Dual DT | [17] | Dual digital twin with Lyapunov-based incremental learning for EV batteries. |
| | Multi-Agent DT | [124] | Multi-agent and cloud-edge orchestration for distributed production. |
| | Federated DT | [125] | Federated DT construction using hierarchical game theory and DRL for resource optimization. |
| | Cloud-Edge DT Fusion | [126] | 9.8× processing speedup through multi-level data fusion. |
| | DT for Content Delivery | [127] | DT-driven content delivery with popularity prediction and caching. |
| | DT for Defect Control | [128] | Multidimensional modeling for end-face defect control. |
| | DT for Fault Detection | [129] | End-to-end DT with 99.71% accuracy for real-time fault detection. |
| | DT for 6G | [130] | Hybrid deployment for 6G network DT with GenAI integration. |
| | DT for battery management system (BMS) | [131] | AI and DT technologies review for intelligent battery management. |
| | DT for Wind Turbine | [132] | DT with edge–cloud collaboration for wind turbine operations and maintenance (O&M). |
| | DT for Coal Platform | [133] | Real-time data warehouse and AI-driven analytics. |
| | DT + Anti-Latency Control | [134] | DT and RL for anti-latency intelligent control in 5G networks. |
| | DT + Cloud Optimization | [135] | Edge-cloud DT framework with energy-minimizing task offloading. |
| | DT + multi-agent reinforcement learning (MARL) | [136] | Cloud-edge model with LSTM (edge) and MARL (cloud) for distributed networks. |
| Blockchain for Trust & Security (Pattern 4.2) | Blockchain-DT Verification | [137] | Blockchain verification with DT for secure task offloading. |
| | Consensus Protocol | [138] | pDPoSt + sPBFT consensus for secure parallel RL in edge-cloud networks. |
| Microservices & Service Orchestration (Pattern 4.3) | Microservices CM | [139] | Microservices-based platform achieving 90% diagnostic accuracy. |
| | CeCM | [12] | Five-layer architecture integrating microservices and DT. |
| Industrial Infrastructure & Platforms (Pattern 4.4) | Industrial Gateway | [140] | Service-oriented information model with message middleware. |
| | Framework/Platform | [141] | Arrowhead framework for interoperable service orchestration. |
| | Framework/Platform | [142] | Low-code capabilities with DT engine and knowledge graph. |
| | Software-Defined | [143] | Software-defined edge intelligent controller. |
| | Edge Framework | [144] | Edge-enabled framework for condition monitoring and predictive maintenance (PdM). |
| | Control Architecture | [145] | Hybrid first-principles and data-driven control. |
| | Exception Control | [146] | Three-level self-adaptive collaboration with fuzzy Bayesian. |
| | Fault Injection | [147] | Knowledge graph-based fault injection with 42.76% time reduction. |
| | Smart Grid | [7] | Comprehensive review of edge–cloud collaborative computing (EC-CC) architecture for smart grid. |
| | AI-Driven Manufacturing | [148] | AIaCM framework for customized manufacturing with edge–cloud collaboration. |
| | Intelligent Machine Tool | [149] | IMT-ECC architecture with data, information, and knowledge collaboration. |
| | AI Agent PdM | [150] | AI agent-based PdM framework integrating retrieval-augmented generation (RAG), large language models (LLMs), and digital twins. |

Foundational studies established the synchronization and architectural principles required for reliable cloud–edge twin deployment. Jia et al. [121] addressed clock synchronization in industrial Internet of Things (IIoT) systems through virtual clock modeling, and Shen et al. [123] developed a cloud-edge framework for process digital twin generation through real-virtual collaborative process tracking. Subsequent work moved toward multi-level or domain-specific twins with differentiated responsibilities. Xie et al. [17] and Liu et al. [18] partitioned battery-management functions between cloud and edge twins, Li et al. [122] proposed a biologically inspired DT framework for product-development-production co-evolution, and Li et al. [132] developed a wind-turbine operation-and-maintenance framework that combines digital twins with collaborative cloud–edge computing.

The DT literature has since expanded from monitoring to optimization, control, and large-scale orchestration. Chen et al. [125] studied federated DT construction across distributed sensing nodes, whereas Nie et al. [124] and Lei et al. [126] emphasized multi-agent orchestration and multi-level cloud-edge fusion. Chai et al. [134] integrated DTs with reinforcement learning for anti-latency control in 5G industrial networks, Feng et al. [135] used DT-enabled cloud optimization to reduce delay and energy consumption in edge–cloud IoT, and Han et al. [136] combined edge-side long short-term memory (LSTM), cloud-side multi-agent reinforcement learning (MARL), and DT feedback for scalable distributed-network optimization. Application-specific studies further extended DT ideas to content delivery [127], defect-control and manufacturing quality scenarios [128,133], IIoT fault detection [129], intelligent battery systems [131], and generative-artificial-intelligence-enabled (GenAI-enabled) 6G architectures [130]. Together, these works show that digital twins are

evolving from monitoring surrogates into a general system layer for edge–cloud coordination, optimization, and trustworthy industrial control.

#### 3.4.2. Blockchain for Trust and Security

Blockchain technology provides decentralized trust mechanisms for edge–cloud collaboration scenarios where multiple stakeholders with potentially competing interests share infrastructure and data. Blockchain-based trust mechanisms are explored less frequently than digital twins, but they address a distinct need in shared or decentralized infrastructures. Yang et al. [138] proposed a blockchain-assisted scheduling architecture for secure parallel reinforcement learning, and Li et al. [137] verified DT-driven coordination instructions through blockchain-backed analysis. These studies suggest that blockchain is most relevant when auditability and tamper resistance are more important than minimal overhead.

#### 3.4.3. Microservices and Service Orchestration

Microservices architectures enable modular, scalable, and maintainable service deployment across edge-cloud infrastructure, addressing the complexity of managing diverse condition monitoring applications with heterogeneous requirements. It has been proposed as a way to make ECC platforms more modular and maintainable. Yang et al. [139] developed a Kubernetes-managed microservices platform for collaborative condition monitoring, and Guo et al. [12] integrated microservices with digital twins in a five-layer cloud-edge collaborative manufacturing architecture. Together, these studies show that service decomposition is useful not only for scaling but also for managing heterogeneous monitoring functions across the lifecycle of industrial applications.

#### 3.4.4. Industrial Infrastructure and Platforms

Industrial infrastructure and platform papers address the foundational hardware, software, and architectural components enabling edge–cloud collaboration for machine condition monitoring. This category encompasses 12 papers spanning industrial gateways, software-defined infrastructure, control architectures, and application frameworks. These papers focus on the enabling substrate rather than on a single algorithm. Zhang et al. [140] and Hastbacka et al. [141] addressed gateways and heterogeneous service orchestration, while Hu et al. [143], Byrtus [144], and Lou et al. [149] emphasized software-defined, hierarchical, or machine-tool-centered infrastructure design for Industry 4.0 deployments. Control-oriented studies by Wang et al. [145,146] combined first-principles reasoning, data-driven models, and exception handling. Broader platform studies then moved toward low-code deployment, collaborative observability, smart-grid architecture, AI-driven customized manufacturing, and agent-enabled predictive-maintenance workflows [7,142,147,148,150]. Collectively, these works indicate that industrial uptake of ECC depends as much on middleware, orchestration, maintainability, and system integration as on the learning algorithm itself.

## 4. Key Enabling Technologies and Computational Approaches

The efficacy of edge–cloud collaboration for machine condition monitoring rests upon a constellation of enabling technologies that collectively address the computational asymmetry between cloud resources and edge constraints, the privacy imperative in industrial data sharing, and the latency requirements of real-time monitoring. This section provides a systematic characterization of the predominant computational paradigms—federated learning, transfer learning, knowledge distillation, deep reinforcement learning, and digital twin architectures—elucidating their theoretical foundations, algorithmic variants adapted for edge–cloud hierarchies, and computational properties that govern deployment decisions.

### *4.1. Federated Learning: Privacy-Preserving Distributed Training*

#### 4.1.1. Theoretical Foundations

Federated learning (FL) establishes a distributed optimization framework wherein multiple edge clients collaboratively train a shared model under the coordination of a central cloud server while preserving data locality [151]. The canonical formulation addresses the following optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(\mathbf{w}), \quad F_k(\mathbf{w}) = \frac{1}{n_k} \sum_{i\in\mathcal{D}_k} \ell(\mathbf{w}; \mathbf{x}_i, y_i) \tag{1}$$

where $K$ denotes the number of edge clients, $n_k$ represents the sample count at client $k$, and $F_k(\mathbf{w})$ is the local objective function computed over the private dataset $\mathcal{D}_k$. The Federated Averaging (FedAvg) algorithm iteratively performs the following: (i) local stochastic gradient descent on each client for $E$ epochs, yielding updated parameters $\mathbf{w}_k^{t+1}$; (ii) server aggregation through weighted averaging $\mathbf{w}^{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} \mathbf{w}_k^{t+1}$ and (iii) global model broadcast to all clients. This paradigm achieves communication complexity $O(K \cdot d \cdot R)$ for $R$ communication rounds, contrasting with $O(K \cdot n_k \cdot d)$ for centralized approaches that require raw data upload [6].

The theoretical appeal of FL for industrial MCM stems from several properties: (i) data locality preserves proprietary process knowledge at manufacturing facilities; (ii) communication efficiency transmits model parameters rather than high-frequency sensor streams and (iii) collaborative generalization enables models trained on distributed equipment to generalize across operational conditions unobserved at any single site.

#### 4.1.2. Algorithmic Variants for Edge–Cloud MCM

The reviewed literature reveals a rich taxonomy of FL variants addressing challenges specific to industrial condition monitoring.

Quality-aware aggregation is one response to the limitations of standard FedAvg under heterogeneous industrial data. Guo et al. [44] weighted client updates according to reconstruction error on a validation set, thereby suppressing poor-quality contributions and improving RUL prediction. FedMAKE addressed a different limitation by replacing architecture-dependent parameter averaging with attention- and prototype-based knowledge carriers that can be exchanged across heterogeneous models [62]. Hierarchical and asynchronous aggregation methods then reduced the delays introduced by strict synchronization, as shown by Ma et al. [11,49] and Su et al. [48].

Bandwidth-aware FL variants target industrial wireless constraints directly. Lu et al. [50] used event-triggered communication, Xiao et al. [47] prioritized updates by generalization potential, and Gao et al. [63] reduced payload through compressive sensing. Domain-shift-aware FL extends the same logic to generalization: Eslaminia et al. [52], He and Shen [53], and Ye et al. [54] all learned distributed representations that remain useful under previously unseen operating conditions. Transfer- and zero-shot-enhanced FL, including CECFTL and semantic-knowledge-based diagnosis, further reduce dependence on complete fault coverage at individual clients [55,58].

#### 4.1.3. Computational Properties and Deployment Considerations

FL deployment in edge–cloud MCM exhibits distinct computational profiles. Client-side computation scales as $O(E \cdot n_k \cdot d)$ per round, where $d$ denotes model dimensionality—suitable for edge devices when $E$ and $d$ are constrained. Server-side aggregation complexity is $O(K \cdot d)$, trivial compared with training. Memory requirements at edge devices scale linearly with model size, motivating compression techniques discussed in Section 4.3.

Privacy guarantees, while implicit in data locality, can be strengthened through differential privacy [46], secure aggregation, and homomorphic encryption—though at computational cost.

### *4.2. Transfer Learning: Cross-Domain Knowledge Adaptation*

#### 4.2.1. Theoretical Foundations

Transfer learning addresses the fundamental challenge of domain shift—the discrepancy between source domain $\mathcal{D}_S = \{(\mathbf{x}_i^s, y_i^s)\}$ used for training and target domain $\mathcal{D}_T = \{(\mathbf{x}_j^t, y_j^t)\}$ encountered at deployment [152]. In edge–cloud MCM, domain shift arises from equipment aging, operational condition variations, and cross-machine deployment. The theoretical objective minimizes target domain generalization error $\epsilon_T(h)$ by leveraging source domain knowledge:

$$\epsilon_T(h) \leq \epsilon_S(h) + d_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) + \lambda^* \tag{2}$$

where $\epsilon_S(h)$ is source error, $d_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T)$ quantifies distribution discrepancy under hypothesis space $\mathcal{H}$, and $\lambda^*$ is the optimal joint error. This bound motivates two transfer strategies: (i) minimizing distribution discrepancy through domain adaptation and (ii) learning domain-invariant representations that generalize across domains.

In edge–cloud architectures, transfer learning decomposes asymmetrically: the cloud, with abundant computational resources and historical datasets, trains source models; the edge, with limited data and compute, fine-tunes models for local deployment. This decomposition reduces edge-side labeling burden while preserving diagnostic accuracy under target domain shift.

#### 4.2.2. Algorithmic Variants for Edge–Cloud MCM

Adversarial domain adaptation remains one of the main transfer strategies. Zou et al. [72] combined ensemble empirical mode decomposition (EEMD) signal decomposition with Domain Adversarial Neural Networks, using cloud-side training to learn domain-invariant features that can be deployed at the edge without labeled target-domain data.

Other studies focus on multiscale or parameter-level transfer. Li et al. [71] developed MSDF-VAE, which combines multiscale convolution with variational autoencoder (VAE)-based domain fusion and assigns a lightweight autoencoder to the edge for real-time diagnosis. Tang et al. [75] and Zhou et al. [76] showed that transferring cloud-trained parameters can substantially reduce the amount of target-domain data required for rapid adaptation in fault diagnosis and battery RUL prediction.

Bridging offline model development and online deployment, Zeng et al. [78] proposed the offline-transfer-online (OTO) framework for deep reinforcement learning, in which offline cloud training is followed by transfer and online adaptation at the edge. Federated transfer learning pushes this idea further by combining privacy preservation with cross-domain adaptation, as demonstrated by Wang et al. [57], Sima et al. [56], and Liang et al. [55].

#### 4.2.3. Computational Properties

Transfer learning exhibits favorable computational asymmetry for edge–cloud deployment. Cloud-side source pretraining requires $O(|\mathcal{D}_S| \cdot d^2)$ operations—expensive but infrequent. Edge-side fine-tuning requires $O(|\mathcal{D}_T| \cdot d)$ operations with $|\mathcal{D}_T| \ll |\mathcal{D}_S|$, suitable for resource-constrained devices. Memory footprint scales with the number of transferable parameters; selective transfer of final layers reduces memory overhead while preserving task-relevant knowledge.

### *4.3. Knowledge Distillation: Model Compression for Edge Deployment*

#### 4.3.1. Theoretical Foundations

Knowledge distillation (KD) transfers knowledge from a large teacher network $f_T$ to a compact student network $f_S$ through a supervised learning process that mimics teacher outputs [153]. The student network minimizes a combined loss:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{hard}}(y, \hat{y}_S) + (1 - \alpha) \cdot \mathcal{L}_{\text{soft}}(\hat{y}_T, \hat{y}_S) \quad (3)$$

where $\mathcal{L}_{\text{hard}}$ denotes cross-entropy with ground truth labels $y$, and $\mathcal{L}_{\text{soft}}$ measures divergence between softened teacher outputs $\hat{y}_T$ and student outputs $\hat{y}_S$. Soft targets encode richer information than hard labels, capturing inter-class relationships that improve student generalization. The temperature parameter $T$ controls soft target smoothness: $\hat{y}_T^{(i)} = \text{softmax}(z_T^{(i)}/T)$ where $z_T$ are teacher logits.

In edge–cloud MCM, KD addresses the computational asymmetry between cloud-hosted teacher models and edge-deployed student models. The cloud, with abundant resources, trains high-capacity models on comprehensive datasets; distillation produces compressed models achieving comparable accuracy with significantly reduced inference latency and memory footprint, enabling real-time edge diagnosis.

#### 4.3.2. Algorithmic Variants for Edge–Cloud MCM

Multidimensional distillation extends beyond output logits to intermediate features, fusion representations, and compressed student architectures. Wang et al. [70] developed MKDC for edge deployment by combining measurement-point optimization with multidimensional distillation-based compression of feature-extraction and fusion units. Across their friction-plate and gearbox case studies, the method reduced model size by more than half while limiting the average accuracy loss to 2.2–3.5%.

Distillation has also been adapted to incremental and bidirectional learning. Ji et al. [69] proposed spatial-alignment-based KD for class-aware incremental learning in partial discharge monitoring, enabling edge models to learn new fault patterns while retaining earlier knowledge. He et al. [80] used deep mutual learning between cloud-hosted ResNet models and compact edge networks so that cloud and edge improve one another rather than participating in one-way transfer only.

For control-oriented applications, Jang et al. [68] showed that policy distillation can compress cloud-trained reinforcement-learning policies into edge-deployable control agents. Distillation has also been integrated with FL, as in the three-layer federated reinforcement learning architecture of Zhou et al. [66] and the model-agnostic knowledge exchange strategy in FedMAKE [62].

#### 4.3.3. Computational Properties and Compression Efficiency

KD can provide favorable compression–accuracy trade-offs for edge deployment, but the practical gains depend strongly on architecture, dataset, and hardware. In the MKDC study, for example, model size was reduced by more than half and runtime was shortened while diagnostic accuracy remained above 96–98% on the reported test conditions [70]. More generally, compression reduces memory footprint and inference cost, which is why distillation remains attractive for deployment on embedded edge devices. The distillation process itself still requires teacher inference on training data, incurring a one-time computational cost that is amortized over edge deployment.

### *4.4. Deep Reinforcement Learning: Adaptive Resource Orchestration*

#### 4.4.1. Theoretical Foundations

Deep reinforcement learning (DRL) provides a framework for sequential decision-making under uncertainty, wherein an agent learns a policy $\pi(a|s)$ that maps system states $s \in \mathcal{S}$ to actions $a \in \mathcal{A}$ to maximize cumulative reward [154]. The optimization objective maximizes expected discounted return:

$$\max_{\pi} J(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \tag{4}$$

where $\tau = (s_0, a_0, r_0, s_1, \ldots)$ denotes a trajectory, $\gamma \in [0, 1)$ is the discount factor, and $r(s_t, a_t)$ is the immediate reward. Deep neural networks approximate the policy (policy gradient methods) or value function (value-based methods), enabling learning in high-dimensional state spaces characteristic of edge–cloud systems.

In edge–cloud MCM, DRL addresses dynamic resource orchestration: task offloading decisions (local vs. edge server vs. cloud), resource allocation (CPU, memory, bandwidth), and workload scheduling under time-varying conditions. The Markov decision process (MDP) formulation captures system dynamics, where state $s_t$ encodes workload, network conditions, and resource availability; action $a_t$ specifies offloading targets and resource allocations; reward $r_t$ balances latency, energy consumption, and diagnostic accuracy.

#### 4.4.2. Algorithmic Variants for Edge–Cloud MCM

Value-based methods such as deep Q-networks (DQN) remain common when offloading decisions are discrete. Tao et al. [101] proposed C3O, which combines DQN with convex optimization for multi-task offloading and reports 7–73% delay reduction across varying workload conditions. Wang and Yang [102] extended this line with attention mechanisms that help the agent focus on the most relevant dimensions of large system states.

When decisions are continuous, policy-gradient methods offer finer control. Hu et al. [111] proposed a DDPG variant with double actors and double critics, termed D4PG, for digital-twin-enabled IIoT resource management. Their framework jointly allocates CPU cycles, DT-model resources, and communication bandwidth, and further introduces a synchronization threshold to reduce straggler-induced delay during distributed learning. Multi-agent reinforcement learning becomes necessary when multiple users, devices, or servers interact, as shown by the multi-agent deep deterministic policy gradient (MADDPG)-based framework of Wang et al. [108], the collaborative Q-learning system of Yang et al. [109], and the vehicle-road orchestration strategy of Zhang et al. [103].

Safety-critical deployments have also motivated risk-sensitive formulations. Zhu et al. [115] incorporated explicit penalties for edge overload and deadline violations in train–edge–cloud collaborative computing, while offline–online transfer frameworks such as that of Zeng et al. [78] reduce the prohibitive online training cost of DRL by shifting most learning to cloud simulators before edge-side fine-tuning.

#### 4.4.3. Computational Properties

DRL exhibits pronounced computational asymmetry favorable for edge–cloud deployment. Training complexity scales as $O(T \cdot |\mathcal{S}| \cdot |\mathcal{A}|)$ for $T$ timesteps—computationally prohibitive for edge devices but tractable on cloud graphics processing units (GPUs). Inference complexity is $O(|\mathcal{S}| \cdot h)$ for $h$ hidden units—enabling microsecond-scale real-time decisions on edge hardware. Memory footprint during inference is minimal (network parameters only), contrasting with training, which requires experience replay buffers. The learned policy network is a compact neural network, typically 100 KB–10 MB, suitable for edge deployment.

### *4.5. Digital Twin Architectures: Physics–Data Hybrid Virtualization*

#### 4.5.1. Theoretical Foundations

Digital twin (DT) architectures integrate physics-based simulation models with data-driven inference to create high-fidelity virtual representations of physical assets [155]. The theoretical framework comprises the following: (i) a physical entity $\mathcal{P}$ with state $\mathbf{x}(t)$ evolving under dynamics $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t)$; (ii) a virtual counterpart $\mathcal{V}$ with state $\hat{\mathbf{x}}(t)$ simulated from a model $\hat{f}$ and (iii) a bidirectional synchronization mechanism maintaining $\|\mathbf{x}(t) - \hat{\mathbf{x}}(t)\| \leq \epsilon$ through data assimilation and model updating.

The hybrid nature of DT—combining physics-based partial differential equations with data-driven neural networks—addresses limitations of purely physics-based models (computational expense, parameter uncertainty) and purely data-driven models (lack of extrapolation, interpretability). Physics-informed neural networks (PINNs) embed governing equations into loss functions, ensuring predictions respect physical laws while fitting observed data.

In edge–cloud MCM, DT architectures partition asymmetrically: cloud-side twins maintain high-fidelity physics-based simulations requiring substantial computation; edge-side twins provide reduced-order models (ROMs) for real-time inference. This dual-twin paradigm reconciles the tension between simulation fidelity and inference latency.

#### 4.5.2. Architectural Variants for Edge–Cloud MCM

Dual digital twin architectures provide one of the clearest examples of cloud–edge role specialization. Xie et al. [17] proposed a dual-twin design for electric vehicle (EV) battery management in which cloud twins execute comprehensive electrochemical models while edge twins use gated recurrent unit (GRU)-based reduced-order models for real-time state estimation. To counter data drift, Lyapunov-based incremental learning constrains reduced-order models to remain close to full-twin outputs, achieving 1.70% average prediction error and 43.3% accuracy improvement.

Other studies broaden the digital-twin design space. Chen et al. [125] proposed federated twin construction so that partial twins can be built from local data at edge servers and aggregated in the cloud without violating data locality. Shen et al. [123] developed RVCPT for manufacturing, emphasizing coevolution, anomaly discrimination, and real–virtual fusion between edge and cloud twins. Liu et al. [18] demonstrated a timescale-based decomposition for battery management, assigning slow-timescale degradation modeling to the cloud and fast-timescale state estimation to the edge. Nie et al. [124] further showed that digital twin orchestration can be integrated with multi-agent coordination for distributed production control.

#### 4.5.3. Computational Properties and Fidelity–Latency Tradeoffs

DT architectures exhibit pronounced fidelity–latency tradeoffs. High-fidelity physics-based models (computational fluid dynamics (CFD), finite element analysis (FEA), and electrochemical partial differential equations (PDEs)) require $O(10^3)$–$O(10^6)$ floating-point operations per timestep, suitable only for cloud deployment. Reduced-order models (proper orthogonal decomposition, dynamic mode decomposition, neural network surrogates) achieve $O(10^1)$–$O(10^3)$ operations per timestep, enabling millisecond-scale edge inference. Memory requirements for high-fidelity models scale with mesh resolution ($O(10^6)$–$O(10^9)$ parameters), while ROMs require $O(10^3)$–$O(10^6)$ parameters.

Synchronization overhead between cloud and edge twins depends on state dimensionality and update frequency. Event-triggered synchronization reduces overhead by transmitting updates only when divergence exceeds thresholds. Lyapunov-based incremental learning ensures bounded synchronization error while minimizing communication frequency.

### *4.6. Comparative Analysis and Selection Guidelines*

Table 7 provides a comparative analysis of the five enabling technologies across computational complexity, communication requirements, privacy properties, and suitability for edge deployment.

**Table 7.** Comparative Analysis of Enabling Technologies for Edge–Cloud MCM.

| Technology | Edge Computation | Communication | Privacy | Key Variants | Primary Application |
|---|---|---|---|---|---|
| Federated Learning | $O(E \cdot n_k \cdot d)$ per round | $O(K \cdot d \cdot R)$ | High (data locality) | FedRUL, FedMAKE, Hierarchical FL, Event-triggered FL | Privacy-preserving distributed training |
| Transfer Learning | $O(\lvert\mathcal{D}_T\rvert \cdot d)$ fine-tuning | $O(d)$ one-time | Medium | Adversarial domain adaptation (DA), MSDF-VAE, OTO | Cross-domain adaptation |
| Knowledge Distillation | $O(\lvert\mathcal{D}\rvert \cdot d_S)$ inference | $O(d_S)$ deployment | Low | MKDC, Bidirectional KD, Policy Distillation | Model compression |
| Deep RL | $O(\lvert\mathcal{S}\rvert \cdot h)$ inference | $O(h)$ policy deployment | N/A | DQN, DDPG, MADDPG, Risk-sensitive RL | Dynamic resource orchestration |
| Digital Twin | $O(10^1$–$10^3)$ ROM inference | $O$(state dim) sync | Medium | Dual DT, Federated DT, Process DT | High-fidelity simulation |

Technology selection depends on the dominant deployment constraint. Privacy-critical multi-site monitoring typically favors FL with secure aggregation or differential privacy. Cross-domain generalization is better served by transfer learning and domain adaptation, or by federated transfer learning when data sharing is restricted. Where edge resources are severely constrained, knowledge distillation remains the most practical route because compression ratios of 10–50$\times$ can often be achieved with modest accuracy loss. DRL is most suitable when the central problem is adaptive orchestration under time-varying workloads, whereas digital twin architectures are preferable when high-fidelity simulation and state synchronization are essential. In practice, hybrid combinations are increasingly common, including FL with distillation for privacy-preserving compression, DRL with digital twins for adaptive simulation, and federated transfer learning for privacy-preserving adaptation. The appropriate combination ultimately depends on the relative importance of accuracy, latency, privacy, and resource efficiency in the target MCM deployment.

## 5. Application Domains and Deployment Patterns

The reviewed literature shows that ECC for MCM is no longer confined to a single industrial niche. Instead, it has developed along several application trajectories shaped by different sensing modalities, response-time requirements, and infrastructure constraints. Table 8 summarizes the dominant application settings represented in the 147 reviewed publications.

**Table 8.** Application Domain Distribution and Equipment Categories in Edge–Cloud MCM.

| Domain | Papers | Share | Equipment | Primary Sensor | Key References |
|---|---|---|---|---|---|
| Manufacturing/IIoT | 95 | 64.6% | Bearings (35+)<br>Rotating Machinery (20)<br>CNC tools (8)<br>Gearboxes (6) | Vibration, AE<br>Vibration, Temperature<br>Vibration, Vision<br>Vibration, Acoustic | FedRUL [44], BearingFM [37]<br>LiteDiagNet [35], MSDF-VAE [71]<br>Tool condition [34], CNC [120]<br>QSCN [9] |
| Energy Systems | 35 | 23.8% | Wind Turbines (10)<br>Li-ion Batteries (12)<br>Photovoltaic (8)<br>Power Electronics (5) | Vibration, SCADA<br>Current, Voltage, Temp<br>I-V Curve, Thermal<br>Current, Voltage | Offshore WT FL [50]<br>Dual DT [17]<br>Photovoltaic (PV) fault [42]<br>PES monitoring [92] |
| Transportation | 8 | 5.4% | Trains, EVs, IoV | Multi-modal, GPS | TEC [115], IoV MADRL [103] |
| Infrastructure | 6 | 4.1% | Power Lines, Grid | Visual, Current | Insulator [41] |
| Others | 3 | 2.1% | General frameworks | Various | Cross-domain surveys |
| **Total** | **147** | **100%** | – | – | – |

### *5.1. Domain Distribution and Statistical Overview*

Manufacturing and energy systems account for the largest share of the literature, while transportation and infrastructure-oriented deployments remain smaller but technically distinctive. This imbalance is unsurprising: manufacturing offers abundant rotating machinery data and strong incentives for predictive maintenance, whereas energy applications motivate ECC through geographical dispersion, intermittent connectivity, and the need for long-horizon health estimation.

### *5.2. Cross-Domain Mechanism, QoS, and Maturity Comparison*

To move from illustrative domain descriptions to a comparative perspective, Table 9 summarizes mechanism prevalence, sensing modalities, quality-of-service priorities, and maturity levels across the four main application domains. Percentages for mechanisms are overlap-aware and indicate the proportion of papers in each domain where that mechanism appears as a primary or secondary label.

**Table 9.** Cross-Domain Comparative Matrix: Mechanisms, QoS, Sensing, and Maturity.

| Domain | FL-Linked | DT-Linked | Task-Orch. Linked | QoS Priority | Dominant Sensing Modalities | Maturity Level |
|---|---|---|---|---|---|---|
| Manufacturing | 47% | 19% | 28% | Low latency and stable throughput in continuous production lines | Vibration, acoustic emission, motor current, visual inspection | High |
| Energy | 31% | 46% | 23% | Reliability under intermittent links and long-horizon prognostics | Voltage/current/temperature (battery), SCADA and vibration (wind), thermal imagery (PV) | Medium–High |
| Transportation | 25% | 25% | 63% | Ultra-low latency and safety-aware scheduling under mobility | Multi-modal onboard sensing, trajectory/context streams, fleet telemetry | Medium |
| Infrastructure | 17% | 33% | 33% | Bandwidth-efficient remote inspection and robust event-triggered upload | Visual/UAV imagery, power signals, sparse distributed sensing | Medium |

The comparison reveals three robust patterns. First, federated learning dominates manufacturing because multi-site factories share similar asset classes yet face data-governance constraints, making privacy-preserving collaborative training especially attractive. Second, digital twins are more prevalent in energy systems, particularly batteries and wind assets, because lifecycle-aware state tracking and long-horizon degradation reasoning are central requirements in geographically distributed environments. Third, task orchestration becomes the dominant mechanism in transportation because mobility, handovers, and strict safety constraints require continuous runtime reallocation of computation and communication.

QoS priorities and sensing modalities also differ systematically. Manufacturing is driven by sub-second anomaly response on dense vibration and current streams; energy systems prioritize robust monitoring under sparse or unstable connectivity; transportation emphasizes deadline-aware scheduling with strong reliability guarantees; infrastructure applications prioritize communication-efficient perception due to remote deployment and high upload costs. These differences explain why reported maturity is currently highest in manufacturing, while other domains remain more heterogeneous in deployment practices.

### *5.3. Manufacturing and Industrial Internet of Things*

Manufacturing dominates the literature, accounting for 95 papers (64.6%). The strong concentration in this domain reflects the abundance of instrumented rotating assets, the economic importance of downtime reduction, and the availability of plant-level communication infrastructure that makes edge–cloud deployment practical. Bearings are the most heavily studied component because their degradation is both common and measurable

through vibration analysis. In multi-site bearing monitoring, privacy-preserving collaboration is a recurrent theme: FedRUL [44], FedMAKE [62], and FDG-CM [52] show how federated training can combine data from distributed factories without centralizing raw signals. At the same time, BearingFM [37] illustrates a newer direction in which large cloud-trained representations are adapted at the edge through lightweight fine-tuning and distillation.

Work on rotating machinery, tool monitoring, and gearboxes emphasizes deployability under noise, workload variation, and tight hardware budgets. LiteDiagNet [35], MSDF-VAE [71], and the shallow–deep prediction architecture of Jing et al. [82] show how cloud assistance can improve robustness without sacrificing edge responsiveness. In machining applications, lightweight visual models and task-aware offloading schemes are prominent because tooling faults must be recognized under rapidly changing cutting conditions [34,120]. Gearbox studies similarly favor compressed inference and transfer-based adaptation, as seen in QSCN and CECFTL [9,55]. Across these settings, the typical deployment pattern is a three-level hierarchy in which sensors and embedded devices perform preprocessing and first-pass inference, plant-level servers support richer diagnosis and model adaptation, and cloud services maintain historical knowledge, foundation models, and cross-site coordination. In practice, this translates into edge latency targets below 10 ms for anomaly screening, plant-level diagnosis on the order of 10–100 ms, and cloud-supported prognostics on the order of seconds.

### *5.4. Energy Systems*

Energy applications account for 35 papers (23.8%) and place different demands on ECC because assets are geographically dispersed, communication links are often intermittent, and health assessment may require long-horizon electrochemical or aero-mechanical reasoning. Wind turbine studies illustrate the need to balance sparse communication with collaborative learning. Event-triggered FL reduces unnecessary parameter exchange in offshore settings [50], while digital-twin-oriented work extends the analysis toward lifecycle-aware health management [122]. The dominant constraints in this subdomain are limited backhaul bandwidth, strong operating-condition variability, expensive fault labeling, and the reluctance of operators to share raw data across wind farms.

Battery management has become one of the fastest-growing ECC topics because fleet monitoring requires local protection, system-level prognosis, and model updating across large populations of cells or vehicles. Dual digital twin architectures [17] and AI-assisted battery twins [18] partition high-fidelity electrochemical reasoning and reduced-order estimation across cloud and edge tiers, while transfer-based RUL prediction reduces the need for extensive local retraining [76]. In this area, monitoring targets extend beyond fault class labels to state of charge, state of health, remaining useful life, internal temperature, and degradation trajectory. Photovoltaic monitoring introduces a different sensing profile, with thermal or visual inspection dominating many deployments. Tang et al. [42] showed that a UAV-cloud collaborative pipeline can reduce inspection time dramatically by combining edge-side hotspot screening with cloud-side high-resolution analysis, and Liu et al. [85] extended this logic to small-sample PV fault diagnosis under changing operating conditions. Compared with manufacturing, energy deployments rely more heavily on wireless links, event-triggered upload, and low-power edge hardware.

### *5.5. Transportation, Infrastructure, and Surveillance*

Transportation applications remain fewer in number, but they place some of the strictest demands on ECC design. Railway monitoring work demonstrates how mobility and safety requirements reshape the hierarchy: onboard devices perform fast

anomaly screening, trackside mobile edge computing (MEC) nodes handle richer diagnosis, and cloud services support fleet-level scheduling and digital twin maintenance [74,115]. Internet-of-Vehicles scenarios extend this pattern through cooperative offloading and multi-agent control, where handovers, intermittent connectivity, and heterogeneous vehicle resources make static deployment policies ineffective [103]. In this domain, robustness to network variation is as important as algorithmic accuracy.

Infrastructure inspection and surveillance share a related emphasis on bandwidth-efficient perception. In power infrastructure monitoring, coarse recognition at the edge and fine recognition in the cloud remain a common architecture because UAV or remote devices cannot support full-resolution analysis continuously [41]. Power-electronics monitoring further shows the value of novelty-triggered transmission in reducing communication load without sacrificing diagnostic sensitivity [92]. Video surveillance studies follow a similar logic: edge devices perform first-pass object detection and upload only uncertain samples or compact metadata, which substantially reduces backbone traffic while retaining cloud-level recognition quality [43]. These studies collectively show that ECC is most valuable when the communication burden of raw data transport would otherwise dominate the end-to-end monitoring pipeline.

## 6. Challenges and Open Issues

Although the reviewed literature demonstrates that ECC can substantially improve the practicality of MCM across diverse industrial settings, it also shows that the field remains constrained by a set of unresolved system-level problems that cannot be addressed by gains in benchmark accuracy alone. The central challenge is no longer whether edge and cloud can collaborate, but how such collaboration can remain efficient, trustworthy, and maintainable under realistic industrial variability.

First, the computational tractability of advanced models remains a persistent bottleneck. Recent studies indicate that large pre-trained or foundation-style models can improve transferability and reduce dependence on labeled data [37,156], yet practical deployment still relies heavily on quantization, lightweight model design, distillation, and selective partitioning to satisfy latency, memory, and energy constraints at the edge [9,35,70]. What remains missing is a principled framework for deciding which model components should remain cloud-resident, which should be specialized locally, and how updates can be propagated without destabilizing edge-side services.

Second, trustworthy deployment is still insufficiently resolved. In industrial maintenance, decisions must often be justified to operators, planners, and safety managers, which means that ECC pipelines must provide not only accurate predictions but also interpretable evidence, uncertainty awareness, secure data exchange, and verifiable coordination. Existing work on secure collaborative monitoring, privacy-preserving federated learning, and blockchain-backed coordination provides important building blocks [6,8,137], but unified solutions that combine explainability, privacy, auditability, and low-latency execution remain rare. A related obstacle is interoperability: heterogeneous equipment, communication protocols, and service platforms still require substantial integration effort despite progress in gateway design, service orchestration, and low-code infrastructure [140–142].

Third, synchronization across hierarchical digital twins and operational models remains technically fragile. Dual-twin and cloud-edge DT frameworks have improved the fidelity–latency balance for batteries, process tracking, and other industrial assets [17,18,123], yet robust bidirectional synchronization under multimodal sensing, missing data, asynchronous sampling, and harsh industrial environments is far from solved. As ECC systems become more tightly coupled to simulation, control, and decision support, synchronization errors can propagate across tiers and undermine both diagnosis and downstream optimization.

Finally, long-horizon adaptation under industrial change remains an open problem. Machine aging, operating-condition variation, rare fault emergence, and non-independent-and-identically-distributed (non-IID) multi-site data all erode models trained offline. Current research on federated continual learning, class-incremental distillation, episodic learning, and offline-to-online adaptation offers promising mechanisms for updating models without full retraining [51,59,69,78], but a practically deployable combination of continual adaptation, privacy preservation, and resistance to catastrophic forgetting has not yet emerged. Closing this gap is essential if ECC is to evolve from a collection of successful prototypes into a dependable industrial infrastructure for lifelong condition monitoring.

## 7. Future Research Directions

Future research is likely to move toward foundation-model-based ECC, namely, edge-cloud collaborative architectures in which large pre-trained models provide transferable representations and high-level reasoning, while edge-side lightweight or adapted variants support resource-constrained real-time monitoring. Progress, however, will depend on making such models operationally tractable for industrial deployment. Large pre-trained representations for vibration, acoustic, thermal, and maintenance-log data may provide stronger cross-machine transfer than current task-specific models, and recent work further suggests that cloud-based artificial intelligence (AI) systems built around large language models can support not only fault diagnosis but also autonomous self-healing through the semantic interpretation of logs, maintenance records, and streaming operational data [37,156]. For machine condition monitoring, however, the central question is not simply whether larger models are beneficial, but how they should be compressed, partitioned, updated, and governed across edge and cloud resources under realistic latency, memory, and energy constraints [9,35,70].

Another important direction is the tighter integration of physics-based reasoning, uncertainty quantification, explainability, and trustworthy collaboration. Physics-informed models and hybrid digital twins can reduce unphysical predictions and improve extrapolation beyond observed operating ranges, while interpretable evidence and calibrated uncertainty are essential if ECC outputs are to be accepted in safety-critical maintenance workflows. At the same time, secure collaboration, privacy preservation, and interoperable service infrastructures remain necessary conditions for practical deployment across heterogeneous industrial environments [6,8,137,140–142]. In this sense, future ECC systems should embed explainability and trust directly into their collaborative pipelines rather than treating them as auxiliary functions added after diagnosis.

Human-in-the-loop ECC is also likely to become more important as industrial systems move toward increasingly autonomous but still supervised operation. Future monitoring platforms will need to accommodate operator feedback, maintenance context, and textual knowledge alongside sensor streams, which points to architectures in which cloud services handle long-context reasoning, historical retrieval, and decision support, while edge systems remain responsible for low-latency detection and fail-safe execution. This direction is closely related to the development of synchronized multi-resolution digital twins, because cloud-side twins can support long-horizon simulation and what-if analysis, whereas edge-side twins can maintain timely state awareness for local monitoring and control [17,18,123]. A key research challenge is, therefore, to maintain consistency among human expertise, physical assets, and cyber representations without introducing excessive synchronization overhead.

Finally, continual learning and sim-to-real digital twins remain central to the long-term evolution of the field. Edge–cloud systems that can update incrementally without catastrophic forgetting would better match the realities of machine aging, operating-condition

drift, and the emergence of new fault classes. Recent studies on federated continual learning, class-incremental adaptation, episodic learning, and offline-to-online transfer provide useful starting points [51,59,69,78], but future work should integrate these mechanisms into deployment-ready pipelines that remain privacy-preserving, stable, and auditable throughout long service lifecycles. The most promising ECC systems will likely combine continual adaptation with high-fidelity twins and knowledge-rich cloud services, allowing edge-resident models to evolve safely while preserving the robustness required in industrial condition monitoring.

## 8. Conclusions

This review systematically examined 147 publications published between 2019 and February 2026 and synthesized the literature on edge–cloud collaboration for machine condition monitoring through a four-dimensional taxonomy comprising model-centric, data-centric, resource and task-centric, and architecture and trust-centric mechanisms. The review shows that edge–cloud collaboration has evolved from a simple computation offloading strategy into a broader system design paradigm in which sensing, inference, communication, orchestration, and trust management are jointly configured to satisfy industrial monitoring requirements.

The analysis further indicates that current research is unevenly distributed across these four dimensions. Model-centric collaboration remains the dominant stream, reflecting the strong emphasis placed on federated learning, knowledge transfer, distillation, and split inference. By contrast, data-centric collaboration remains comparatively limited despite its direct relevance to bandwidth-constrained and latency-sensitive deployments. Resource and task-centric studies have strengthened adaptive scheduling and offloading capabilities, whereas architecture and trust-centric studies have highlighted the growing importance of digital twins, interoperable platforms, and secure coordination mechanisms. Taken together, these findings suggest that effective ECC for MCM cannot be achieved by optimizing any single layer in isolation; rather, it requires coordinated design across algorithms, data flows, computing resources, and deployment infrastructures.

The review also identifies several issues that will shape the next stage of the field, including the tractable deployment of foundation-model-based ECC, trustworthy and interoperable industrial implementation, robust synchronization of hierarchical digital twins, and lifelong adaptation under non-stationary operating conditions. Addressing these issues is essential if ECC is to progress from successful case-specific prototypes toward dependable and reusable industrial monitoring systems. Overall, this review provides a structured account of the state of the art and, more importantly, clarifies the principal directions through which edge–cloud collaboration may mature into a practical foundation for next-generation machine condition monitoring.

**Author Contributions:** Conceptualization, J.F. and L.Y.; literature search and screening, L.Y., Q.W. and F.L.; literature classification and synthesis, L.Y., Q.W., F.L. and H.L.; visualization, L.Y. and H.L.; writing—original draft preparation, L.Y.; writing—review and editing, J.F., Q.W., F.L. and H.L.; supervision, J.F. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Shandong Province Technology Innovation Guidance Program (Central Government Guided Local Science and Technology Development Fund), Grant No. YDZX2024127.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** No new data were created or analyzed in this study.

**Acknowledgments:** The authors thank the research community for advancing edge–cloud collaboration technologies and industrial partners for deployment case studies.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| AE | Acoustic Emission |
| AI | Artificial Intelligence |
| BMS | Battery Management System |
| CAE | Convolutional Autoencoder |
| CDC | Coded Distributed Computing |
| CFD | Computational Fluid Dynamics |
| CM-PdM | Condition Monitoring and Predictive Maintenance |
| CNC | Computer Numerical Control |
| CNN | Convolutional Neural Network |
| CPU | Central Processing Unit |
| CV | Computer Vision |
| D4PG | DDPG with double actors and double critics |
| DDPG | Deep Deterministic Policy Gradient |
| DA | Domain Adaptation |
| DG | Domain Generalization |
| DL | Deep Learning |
| DMD | Dynamic Mode Decomposition |
| DNN | Deep Neural Network |
| DQN | Deep Q-Network |
| DRL | Deep Reinforcement Learning |
| DT | Digital Twin |
| DT-PHM | Digital Twin-Driven Prognostics and Health Management |
| ECC | Edge–Cloud Collaboration |
| EEMD | Ensemble Empirical Mode Decomposition |
| FD | Fault Diagnosis |
| FEA | Finite Element Analysis |
| FL | Federated Learning |
| GAN | Generative Adversarial Network |
| FPGA | Field-Programmable Gate Array |
| FTL | Federated Transfer Learning |
| GPS | Global Positioning System |
| GPU | Graphics Processing Unit |
| GRU | Gated Recurrent Unit |
| IFD | Intelligent Fault Diagnosis |
| IIoT | Industrial Internet of Things |
| IMHM | Intelligent Machinery Health Monitoring |
| IoT | Internet of Things |
| IoV | Internet of Vehicles |
| KD | Knowledge Distillation |
| KPI | Key Performance Indicator |
| LLM | Large Language Model |
| LLMs | Large Language Models |
| LMMs | Large Multimodal Models |
| LoRaWAN | Long Range Wide-Area Network |

| | |
|---|---|
| LSTM | Long Short-Term Memory |
| MADDPG | Multi-Agent Deep Deterministic Policy Gradient |
| MADRL | Multi-Agent Deep Reinforcement Learning |
| MARL | Multi-Agent Reinforcement Learning |
| MCM | Machine Condition Monitoring |
| MDP | Markov Decision Process |
| MEC | Mobile Edge Computing |
| ML | Machine Learning |
| NB-IoT | Narrowband Internet of Things |
| NLP | Natural Language Processing |
| non-IID | Non-Independent and Identically Distributed |
| O&M | Operations and Maintenance |
| PDE | Partial Differential Equation |
| PES | Power Electronics Systems |
| OTO | Offline-Transfer-Online |
| PHM | Prognostics and Health Management |
| PINN | Physics-Informed Neural Network |
| PLC | Programmable Logic Controller |
| POD | Proper Orthogonal Decomposition |
| PV | Photovoltaic |
| QoS | Quality of Service |
| RAG | Retrieval-Augmented Generation |
| ResNet | Residual Network |
| RL | Reinforcement Learning |
| ROM | Reduced-Order Model |
| RUL | Remaining Useful Life |
| SCADA | Supervisory Control and Data Acquisition |
| SOH | State of Health |
| TEC | Train-Edge-Cloud |
| TL | Transfer Learning |
| UAV | Unmanned Aerial Vehicle |
| VAE | Variational Autoencoder |
| WT | Wind Turbine |
| YOLO | You Only Look Once |

## References

1. Xu, J.; Sun, Q.; Han, Q.L.; Tang, Y. When Embodied AI Meets Industry 5.0: Human-Centered Smart Manufacturing. *IEEE/CAA J. Autom. Sin.* **2025**, *12*, 485–501. [CrossRef]
2. Calabrese, F.; Regattieri, A.; Bortolini, M.; Gamberi, M.; Pilati, F. Predictive Maintenance: A Novel Framework for a Data-Driven, Semi-Supervised, and Partially Online Prognostic Health Management Application in Industries. *Appl. Sci.* **2021**, *11*, 3380. [CrossRef]
3. Huang, C.; Bu, S.; Lee, H.H.; Chan, C.H.; Kong, S.W.; Yung, W.K.C. Prognostics and Health Management for Predictive Maintenance: A Review. *J. Manuf. Syst.* **2024**, *75*, 78–101. [CrossRef]
4. Hafeez, T.; Xu, L.; Mcardle, G. Edge Intelligence for Data Handling and Predictive Maintenance in IIOT. *IEEE Access* **2021**, *9*, 49355–49371. [CrossRef]
5. Lu, S.; Lu, J.; An, K.; Wang, X.; He, Q. Edge Computing on IoT for Machine Signal Processing and Fault Diagnosis: A Review. *IEEE Internet Things J.* **2023**, *10*, 11093–11116. [CrossRef]
6. Bao, G.; Guo, P. Federated Learning in Cloud-Edge Collaborative Architecture: Key Technologies, Applications and Challenges. *J. Cloud Comput.* **2022**, *11*, 94. [CrossRef] [PubMed]
7. Li, J.; Gu, C.; Xiang, Y.; Li, F. Edge-Cloud Computing Systems for Smart Grid: State-of-the-art, Architecture, and Applications. *J. Mod. Power Syst. Clean Energy* **2022**, *10*, 805–817. [CrossRef]
8. Huang, K.K.; Wang, Q.Z.; Chen, Z.X.; Yang, C.H.; Gui, W.H. Secure Cloud-Edge Collaborative Method for Dynamic Industrial Process Monitoring Using Self-Updating Dictionary Learning. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 14170–14182. [CrossRef]

9. Wu, Q.H.; Luo, J.; Huang, W.B.; Ding, X.X. A Quantized Subtraction-Convolution Network for Industrial Lightweight Edge Interpretable Diagnosis. *Eng. Appl. Artif. Intell.* **2025**, *161*, 112241. [CrossRef]
10. Duan, Z.; Zhang, W.; Zhang, H.; Yang, F. Research on Intelligent Fault Diagnosis of Rotating Machinery for Edge Computing Platforms. *Mech. Syst. Signal Process.* **2025**, *237*, 113101. [CrossRef]
11. Ma, X.; He, X.; Wu, X.Y.; Wen, C.L. Multi-Level Federated Learning Based on Cloud-Edge-Client Collaboration and Outlier-Tolerance for Fault Diagnosis. *Meas. Sci. Technol.* **2023**, *34*, 125148. [CrossRef]
12. Guo, L.; He, Y.L.; Wan, C.C.; Li, Y.T.; Luo, L.K. From Cloud Manufacturing to Cloud-Edge Collaborative Manufacturing. *Robot. Comput.-Integr. Manuf.* **2024**, *90*, 102790. [CrossRef]
13. Shi, J.; Wang, R.; Xian, M.; Wo, T. A Cloud-edge Collaborative Architecture for Data-driven Health Condition Monitoring of Machines. In Proceedings of the 2021 IEEE International Conference on Joint Cloud Computing (JCC), Oxford, UK, 23–26 August 2021; pp. 51–58. [CrossRef]
14. Chen, Y.Y.; Jhong, S.Y.; Tu, S.K.; Lin, Y.H.; Wu, Y.C. Autonomous Smart-Edge Fault Diagnostics via Edge-Cloud-Orchestrated Collaborative Computing for Infrared Electrical Equipment Images. *IEEE Sens. J.* **2024**, *24*, 24630–24648. [CrossRef]
15. Wang, Q.; Li, Q.; Wang, K.; Wang, H.; Zeng, P. Efficient Federated Learning for Fault Diagnosis in Industrial Cloud-Edge Computing. *Computing* **2021**, *103*, 2319–2337. [CrossRef]
16. Zhang, Y.D.; Yin, H.P.; Wang, R.; Zhang, T.F. A Two-Phase Federated Learning Framework for Machinery Fault Diagnosis with Cloud-Edge Collaborative Computing. *Expert Syst. Appl.* **2026**, *297*, 129323. [CrossRef]
17. Xie, J.H.; Yang, R.F.; Hui, S.Y.R.; Nguyen, H.D. Dual Digital Twin: Cloud-edge Collaboration with Lyapunov-based Incremental Learning in EV Batteries. *Appl. Energy* **2024**, *355*, 122237. [CrossRef]
18. Liu, M.W.; Wang, Y.X.; Li, J.F.; Yang, K.; Guo, S.L.; Zhao, L.; Wang, Z.B. AI-assisted Digital Twin Modeling Technology for Lithium-Ion Batteries. *Energy* **2025**, *331*, 137016. [CrossRef]
19. Wang, C.T.; Guo, R.F.; Yu, H.Y.; Hu, Y.; Liu, C.; Deng, C.Y. Task Offloading in Cloud-Edge Collaboration-Based Cyber Physical Machine Tool. *Robot. Comput.-Integr. Manuf.* **2023**, *79*, 102439. [CrossRef]
20. Zhang, K.; Xia, T.; Si, G.; Pan, E.; Xi, L. An Edge-Based Framework for Real-Time Prognosis and Opportunistic Maintenance in Leased Manufacturing System. *IEEE Trans. Autom. Sci. Eng.* **2024**, *21*, 4177–4187. [CrossRef]
21. Hemmati, A.; Raoufi, P.; Rahmani, A.M. Edge Artificial Intelligence for Big Data: A Systematic Review. *Neural Comput. Appl.* **2024**, *36*, 11461–11494. [CrossRef]
22. Bala, A.; Rashid, R.Z.J.A.; Ismail, I.; Oliva, D.; Muhammad, N.; Sait, S.M.; Al-Utaibi, K.A.; Amosa, T.I.; Memon, K.A. Artificial Intelligence and Edge Computing for Machine Maintenance-Review. *Artif. Intell. Rev.* **2024**, *57*, 119. [CrossRef]
23. Soldani, J.; Brogi, A. Anomaly Detection and Failure Root Cause Analysis in (Micro) Service-Based Cloud Applications: A Survey. *ACM Comput. Surv.* **2022**, *55*, 59. [CrossRef]
24. Bao, Z.Q.; Liu, C.F.; Yang, H.; Zhang, J.Y.; Li, Y.Q. From Theory to Industry: A Survey of Deep Learning-Enabled Bearing Fault Diagnosis in Complex Environments. *Eng. Appl. Artif. Intell.* **2026**, *163*, 113068. [CrossRef]
25. Garcia, J.; Rios-Colque, L.; Peña, A.; Rojas, L. Condition Monitoring and Predictive Maintenance in Industrial Equipment: An NLP-Assisted Review of Signal Processing, Hybrid Models, and Implementation Challenges. *Appl. Sci.* **2025**, *15*, 5465. [CrossRef]
26. Saeed, A.; Khan, M.A.; Akram, U.; Obidallah, W.J.; Jawed, S.; Ahmad, A. Deep Learning Based Approaches for Intelligent Industrial Machinery Health Management and Fault Diagnosis in Resource-Constrained Environments. *Sci. Rep.* **2025**, *15*, 1114. [CrossRef]
27. Liu, J.; Du, Y.; Yang, K.; Wu, J.; Wang, Y.; Hu, X.; Wang, Z.; Liu, Y.; Sun, P.; Boukerche, A.; et al. Edge-Cloud Collaborative Computing on Distributed Intelligence and Model Optimization: A Survey. *arXiv* **2025**, arXiv:2505.01821. [CrossRef]
28. Kok, I.; Demirci, O.; Ozdemir, S. When IoT Meet LLMs: Applications and Challenges. *arXiv* **2024**, arXiv:2411.17722. [CrossRef]
29. Xiao, B.; Zhong, J.; Bao, X.; Chen, L.; Bao, J.; Zheng, Y. Digital Twin-Driven Prognostics and Health Management for Industrial Assets. *Sci. Rep.* **2024**, *14*, 13443. [CrossRef]
30. Xu, Y.; Khan, T.M.; Song, Y.; Meijering, E. Edge Deep Learning in Computer Vision and Medical Diagnostics: A Comprehensive Survey. *Artif. Intell. Rev.* **2025**, *58*, 93. [CrossRef]
31. Xia, D.; Xu, P.P.; Han, G.J.; Jiang, J.F. Emerging Perspectives on Embodied Intelligence in Future Smart Manufacturing. *J. Ind. Inf. Integr.* **2026**, *50*, 101020. [CrossRef]
32. Calabrese, M.; Cimmino, M.; Fiume, F.; Manfrin, M.; Romeo, L.; Ceccacci, S.; Paolanti, M.; Toscano, G.; Ciandrini, G.; Carrotta, A.; et al. SOPHIA: An Event-Based IoT and Machine Learning Architecture for Predictive Maintenance in Industry 4.0. *Information* **2020**, *11*, 202. [CrossRef]
33. Wang, H.; Li, F.; Mo, W.; Tao, P.; Shen, H.; Wu, Y.; Zhang, Y.; Deng, F. Novel Cloud-Edge Collaborative Detection Technique for Detecting Defects in PV Components, Based on Transfer Learning. *Energies* **2022**, *15*, 7924. [CrossRef]
34. You, Z.C.; Gao, H.L.; Li, S.C.; Guo, L.; Liu, Y.K.; Li, J.B. Multiple Activation Functions and Data Augmentation-Based Lightweight Network or In Situ Tool Condition Monitoring. *IEEE Trans. Ind. Electron.* **2022**, *69*, 13656–13664. [CrossRef]

35. Li, J.; Liu, S.J.; Tao, Z.Y.; Zhou, X.; Wu, Y.F. LiteDiagNet: Enhancing Noise-Resistant Fault Diagnosis in Rotating Machinery via IoT-Based Cloud-Edge Collaboration. *J. Vib. Control* **2026**, *32*, 1702–1716. [CrossRef]
36. Huang, K.K.; Tao, Z.; Wang, C.; Guo, T.X.; Yang, C.H.; Gui, W.H. Cloud-Edge Collaborative Method for Industrial Process Monitoring Based on Error-Triggered Dictionary Learning. *IEEE Trans. Ind. Inform.* **2022**, *18*, 8957–8966. [CrossRef]
37. Lai, Z.; Yang, C.; Lan, S.; Wang, L.; Shen, W.; Zhu, L. BearingFM: Towards a Foundation Model for Bearing Fault Diagnosis by Domain Knowledge and Contrastive Learning. *Int. J. Prod. Econ.* **2024**, *275*, 109319. [CrossRef]
38. Wang, A.X.; Li, J.J. A Novel Cloud-Edge Collaboration Based Short-Term Load Forecasting Method for Smart Grid. *Front. Energy Res.* **2022**, *10*, 977026. [CrossRef]
39. Sui, Q.K.; Wang, Y.L.; Liu, C.L.; Huang, K.K.; Yang, C.H. Interpretable Switching Deep Markov Model for Industrial Process Monitoring via Cloud-Edge Collaborative Framework. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 3501614. [CrossRef]
40. Ren, L.; Jia, Z.D.; Wang, T.; Ma, Y.H.; Wang, L.H. LM-CNN: A Cloud-Edge Collaborative Method for Adaptive Fault Diagnosis with Label Sampling Space Enlarging. *IEEE Trans. Ind. Inform.* **2022**, *18*, 9057–9067. [CrossRef]
41. Song, C.H.; Xu, W.X.; Han, G.J.; Zeng, P.; Wang, Z.F.; Yu, S.M. A Cloud Edge Collaborative Intelligence Method of Insulator String Defect Detection for Power IIoT. *IEEE Internet Things J.* **2021**, *8*, 7510–7520. [CrossRef]
42. Tang, W.; Yang, Q.; Hu, X.; Yan, W. Deep Learning-Based Linear Defects Detection System for Large-Scale Photovoltaic Plants Based on an Edge-Cloud Computing Infrastructure. *Sol. Energy* **2022**, *231*, 527–535. [CrossRef]
43. Chen, Y.Y.; Lin, Y.H.; Hu, Y.C.; Hsia, C.H.; Lian, Y.A.; Jhong, S.Y. Distributed Real-Time Object Detection Based on Edge-Cloud Collaboration for Smart Video Surveillance Applications. *IEEE Access* **2022**, *10*, 93745–93759. [CrossRef]
44. Guo, L.; Yu, Y.; Qian, M.; Zhang, R.; Gao, H.; Cheng, Z. FedRUL: A New Federated Learning Method for Edge-Cloud Collaboration Based Remaining Useful Life Prediction of Machines. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 350–359. [CrossRef]
45. Yu, Y.; Guo, L.; Gao, H.; He, Y.; You, Z.; Duan, A. FedCAE: A New Federated Learning Framework for Edge-Cloud Collaboration Based Machine Fault Diagnosis. *IEEE Trans. Ind. Electron.* **2024**, *71*, 4108–4119. [CrossRef]
46. Su, Z.; Wang, Y.; Luan, T.H.; Zhang, N.; Li, F.; Chen, T.; Cao, H. Secure and Efficient Federated Learning for Smart Grid with Edge-Cloud Collaboration. *IEEE Trans. Ind. Inform.* **2022**, *18*, 1333–1344. [CrossRef]
47. Xiao, D.; Diao, J.; Ding, J.; Jiang, L.; Zhao, C. FedRad: A Dynamic Low-Cost Federated Learning for Fault Diagnosis of Rotating Machinery. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 3551914. [CrossRef]
48. Su, Y.; Fan, W.; Meng, Q.; Chen, P.; Liu, Y. Joint Adaptive Aggregation and Resource Allocation for Hierarchical Federated Learning Systems Based on Edge-Cloud Collaboration. *IEEE Trans. Cloud Comput.* **2025**, *13*, 369–382. [CrossRef]
49. Ma, X.; Wen, C.L. An Asynchronous Quasi-Cloud/Edge/Client Collaborative Federated Learning Mechanism for Fault Diagnosis. *Chin. J. Electron.* **2021**, *30*, 969–977. [CrossRef]
50. Lu, S.; Gao, Z.; Zhang, P.; Xu, Q.; Xie, T.; Zhang, A. Event-Triggered Federated Learning for Fault Diagnosis of Offshore Wind Turbines with Decentralized Data. *IEEE Trans. Autom. Sci. Eng.* **2024**, *21*, 1271–1283. [CrossRef]
51. Chen, J.; He, J.Y.; Tang, J.H.; Li, W.H.; Yin, Z.H. Knowledge Efficient Federated Continual Learning for Industrial Edge Systems. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 2107–2120. [CrossRef]
52. Eslaminia, A.; Meng, Y.Q.; Nahrstedt, K.; Shao, C.H. Federated Domain Generalization for Condition Monitoring in Ultrasonic Metal Welding. *J. Manuf. Syst.* **2024**, *77*, 1–12. [CrossRef]
53. He, Y.M.; Shen, W.M. FedITA: A Cloud-Edge Collaboration Framework for Domain Generalization-Based Federated Fault Diagnosis of Machine-Level Industrial Motors. *Adv. Eng. Inform.* **2024**, *62*, 102853. [CrossRef]
54. Ye, Z.Z.; Wu, J.; He, X.S.; Jiang, W.X. A Gradient Alignment Federated Domain Generalization Framework for Rotating Machinery Fault Diagnosis. *IEEE Internet Things J.* **2025**, *12*, 24764–24774. [CrossRef]
55. Liang, X.T.; Sun, Y.L.; Kang, S.Q.; Zhao, Z.H.; Wang, Y.J.; Wang, Q.Y. A Rolling Bearing Fault Diagnosis Method Based on Cloud-Edge Collaboration Federated Transfer Learning. *Meas. Sci. Technol.* **2025**, *36*, 046111. [CrossRef]
56. Sima, W.; Zhang, H.; Yang, M. Edge–Cloud Collaboration Detection Approach for Small-Sample Imbalanced Faults in Power Lines. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 3513711. [CrossRef]
57. Wang, R.; Huang, W.G.; Lu, Y.X.; Wang, J.; Ding, C.C.; Liao, Y.; Shi, J.J. Cloud-Edge Collaborative Transfer Fault Diagnosis of Rotating Machinery via Federated Fine-Tuning and Target Self-Adaptation. *Expert Syst. Appl.* **2024**, *250*, 123859. [CrossRef]
58. Li, B.X.; Zhao, C.H. Federated Zero-Shot Industrial Fault Diagnosis with Cloud-Shared Semantic Knowledge Base. *IEEE Internet Things J.* **2023**, *10*, 11619–11630. [CrossRef]
59. Li, B.X.; Zhao, C.H. Federated Episodic Learning to Extrapolate Unseen From Seen Conditions for Industrial IoT Monitoring. *IEEE Internet Things J.* **2025**, *12*, 7518–7531. [CrossRef]
60. Wang, J.Y.; Song, P.Y.; Zhao, C.H.; Ding, J.L. Federated Knowledge Amalgamation with Unbiased Semantic Attributes under Cloud-Edge Collaboration for Heterogeneous Fault Diagnosis. *J. Process Control* **2023**, *131*, 103095. [CrossRef]
61. Jiang, G.; Zhao, K.; Liu, X.; Cheng, X.; Xie, P. A Federated Learning Framework for Cloud–Edge Collaborative Fault Diagnosis of Wind Turbines. *IEEE Internet Things J.* **2024**, *11*, 23170–23185. [CrossRef]

62. Wang, J.Y.; Song, P.Y.; Zhao, C.H. Towards Consensual Representation: Model-agnostic Knowledge Extraction for Dual Heterogeneous Federated Fault Diagnosis. *Neural Netw.* **2024**, *179*, 106618. [CrossRef]
63. Gao, X.; Hou, L.; Chen, B.; Yao, X.; Suo, Z. Compressive-Learning-Based Federated Learning for Intelligent IoT with Cloud–Edge Collaboration. *IEEE Internet Things J.* **2025**, *12*, 2291–2294. [CrossRef]
64. Zhao, L.D.; Ni, S.X.; Wu, D.; Zhou, L. Cloud-Edge-Client Collaborative Learning in Digital Twin Empowered Mobile Networks. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 3491–3503. [CrossRef]
65. Chen, L.; Gao, H.; Guo, L.; Liang, J.; Peng, L. Remaining Useful Life Prediction of Machinery Using Federated Public Feature Representation in Edge-Cloud Collaboration Architecture. *Eng. Appl. Artif. Intell.* **2025**, *143*, 110059. [CrossRef]
66. Zhou, X.K.; Zheng, X.Z.; Cui, X.S.; Shi, J.S.; Liang, W.; Yan, Z.; Yang, L.T.; Shimizu, S.; Wang, K.I.K. Digital Twin Enhanced Federated Reinforcement Learning with Lightweight Knowledge Distillation in Mobile Networks. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 3191–3211. [CrossRef]
67. Gao, Y.; Liu, L.; Zheng, X.; Zhang, C.; Ma, H. *Federated Sensing*: Edge-Cloud Elastic Collaborative Learning for Intelligent Sensing. *IEEE Internet Things J.* **2021**, *8*, 11100–11111. [CrossRef]
68. Jang, I.; Kim, H.; Lee, D.; Son, Y.S.; Kim, S. Knowledge Transfer for On-Device Deep Reinforcement Learning in Resource Constrained Edge Computing Systems. *IEEE Access* **2020**, *8*, 146588–146597. [CrossRef]
69. Ji, J.S.; Shu, Z.; Lu, M.S.; Li, H.Q.; Zheng, Y.J.; Jiang, X.D. Feature-Aligned and Class-Aware Incremental Learning for Online Partial Discharge Monitoring. *IEEE Trans. Power Deliv.* **2025**, *40*, 2717–2728. [CrossRef]
70. Wang, Y.J.; Zhang, Z.G.; Yang, Y.; Xue, C.R.; Zhang, W.H.; Wang, L.M.; Ding, X.X. MKDC: A Lightweight Method for Cloud-Edge Collaborative Fault Diagnosis Model. *IEEE Sens. J.* **2024**, *24*, 32607–32618. [CrossRef]
71. Li, X.B.; Xiao, B.; Chen, X.J.; Jiang, P.; Wang, X.V.; Zheng, P.; Xia, L.Q.; Yin, C. MSDF-VAE: A Cloud-Edge Collaborative Method for Fault Diagnosis Based on Transfer Learning. *IEEE Internet Things J.* **2025**, *12*, 22393–22403. [CrossRef]
72. Zou, P.; Zhang, Z.J.; Jiang, L. Fault Diagnosis of Transfer Learning Equipment Based on Cloud Edge Collaboration plus Confrontation Network. *Teh. Vjesn.-Tech. Gaz.* **2023**, *30*, 1854–1869. [CrossRef]
73. Liu, B.; Chen, C.H. An Adaptive Multihop Branch Ensemble-Based Graph Adaptation Framework with Edge-Cloud Orchestration for Condition Monitoring. *IEEE Trans. Ind. Inform.* **2023**, *19*, 10102–10113. [CrossRef]
74. Zhang, K.; Huang, W.; Hou, X.; Xu, J.; Su, R.; Xu, H. A Fault Diagnosis and Visualization Method for High-Speed Train Based on Edge and Cloud Collaboration. *Appl. Sci.* **2021**, *11*, 1251. [CrossRef]
75. Tang, X.H.; Xu, L.; Chen, G.S. Research on the Rapid Diagnostic Method of Rolling Bearing Fault Based on Cloud-Edge Collaboration. *Entropy* **2022**, *24*, 1277. [CrossRef]
76. Zhou, Y.; Gu, H.H.; Su, T.; Han, X.B.; Lu, L.G.; Zheng, Y.J. Remaining Useful Life Prediction with Probability Distribution for Lithium-Ion Batteries Based on Edge and Cloud Collaborative Computation. *J. Energy Storage* **2021**, *44*, 103342. [CrossRef]
77. Zhang, Y.; Wang, X.; He, J.; Xu, Y.; Zhang, F.; Luo, Y. A Transfer Learning-Based High Impedance Fault Detection Method Under a Cloud-Edge Collaboration Framework. *IEEE Access* **2020**, *8*, 165099–165110. [CrossRef]
78. Zeng, T.Y.; Zhang, X.X.; Duan, J.P.; Yu, C.; Wu, C.; Chen, X. An Offline-Transfer-Online Framework for Cloud-Edge Collaborative Distributed Reinforcement Learning. *IEEE Trans. Parallel Distrib. Syst.* **2024**, *35*, 720–731. [CrossRef]
79. Zhou, Q.; Wu, S.; Jiang, C.X.; Zhang, R.H.; Jing, X.J. Cloud-Edge-Terminal Collaboration-Enabled Device-Free Sensing Under Class-Imbalance Conditions. *IEEE Internet Things J.* **2024**, *11*, 5980–5992. [CrossRef]
80. He, Z.C.; Liang, D.G.; Yang, Y. Fault Diagnosis Scheme for the Rotary Machine Group: A Deep Mutual Learning-Based Approach with Cloud-Edge-End Collaboration. *IEEE Trans. Circuits Syst. II-Express Briefs* **2023**, *70*, 3209–3213. [CrossRef]
81. Fan, H.; Xu, H.; Fan, W.; Chen, B.; Li, Q.; Zhang, T.; Cao, X.; Zhang, X. Cloud–Edge Collaborative Intelligent Fault Diagnosis of Rotor-Bearing System: Methodology and Experiment. *IEEE Sens. J.* **2025**, *25*, 5044–5059. [CrossRef]
82. Jing, T.; Tian, X.; Hu, H.; Ma, L. Deep Learning-Based Cloud–Edge Collaboration Framework for Remaining Useful Life Prediction of Machinery. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7208–7218. [CrossRef]
83. Sun, X.C.; Ding, H.; Li, N.; Dong, X.X.; Sun, J.C.; Zheng, G.Y. Intelligent Fault Diagnosis Method for Shearer Rocker Gear Based on Swin Transformer and Multiscale Convolution Parallel Integration. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 3519816. [CrossRef]
84. Dong, J.; Li, Z.; Zheng, Y.; Luo, J.; Zhang, M.; Yang, X. Real-Time Fault Detection for IIoT Facilities Using GA-Att-LSTM Based on Edge-Cloud Collaboration. *Front. Neurorobot.* **2024**, *18*, 1499703. [CrossRef]
85. Liu, Q.; Yang, B.; Cai, M.X.; Liu, Y.X.; Ma, K.; Guan, X.P. Self-Correcting-Guided Generalized Contrastive Learning Framework for Small-Sample PV Fault Diagnosis with Cloud-Edge Collaboration. *IEEE Trans. Ind. Inform.* **2025**, *21*, 7466–7476. [CrossRef]
86. Sun, T.; Wang, S.Q.; Jiang, S.; Xu, B.W.; Han, X.B.; Lai, X.; Zheng, Y.J. A Cloud-Edge Collaborative Strategy for Capacity Prognostic of Lithium-Ion Batteries Based on Dynamic Weight Allocation and Machine Learning. *Energy* **2022**, *239*, 122185. [CrossRef]
87. Jia, Z.; Ren, L. A Cloud-Edge Adaptive Framework for Equipment Predictive Maintenance in IIoT. In Proceedings of the IECON 2024—50th Annual Conference of the IEEE Industrial Electronics Society, Chicago, IL, USA, 3–6 November 2024; pp. 1–6. [CrossRef]

88. Wang, R.Q.; Song, Q.H.; Peng, Y.Z.; Liu, Z.Q.; Ma, H.F.; Liu, Z.J.; Xu, X. Milling Surface Roughness Monitoring Using Real-Time Tool Wear Data. *Int. J. Mech. Sci.* **2025**, *285*, 109821. [CrossRef]
89. Zhang, X.Y.; Ma, L.; Peng, K.X.; Zhang, C.F.; Shahid, M.A. A Cloud-Edge Collaboration Based Quality-Related Hierarchical Fault Detection Framework for Large-Scale Manufacturing Processes. *Expert Syst. Appl.* **2024**, *256*, 124909. [CrossRef]
90. Zhang, X.Y.; Ma, L.; Peng, K.X.; Zhang, C.F.; Shahid, M.A.; Wang, Y.F. A Cloud-Edge Collaborative Hierarchical Diagnosis Framework for Key Performance Indicator-Related Faults in Manufacturing Industries. *J. Process Control* **2025**, *152*, 103462. [CrossRef]
91. Zhang, M.; Zhang, H.; Zhang, C.; Yuan, D. Communication-Efficient Quantized Deep Compressed Sensing for Edge-Cloud Collaborative Industrial IoT Networks. *IEEE Trans. Ind. Inform.* **2023**, *19*, 6613–6623. [CrossRef]
92. Bhoi, S.K.; Chakraborty, S.; Verbrugge, B.; Helsen, S.; Robyns, S.; El Baghdadi, M.; Hegazy, O. Intelligent Data-Driven Condition Monitoring of Power Electronics Systems Using Smart Edge–Cloud Framework. *Internet Things* **2024**, *26*, 101158. [CrossRef]
93. Shi, Y.; Yi, C.; Chen, B.; Yang, C.; Zhu, K.; Cai, J. Joint Online Optimization of Data Sampling Rate and Preprocessing Mode for Edge–Cloud Collaboration-Enabled Industrial IoT. *IEEE Internet Things J.* **2022**, *9*, 16402–16417. [CrossRef]
94. Wu, Y.; Yang, B.; Zhu, D.; Liu, Q.; Li, C.; Chen, C.; Guan, X. To Transmit or Predict: An Efficient Industrial Data Transmission Scheme with Deep Learning and Cloud-Edge Collaboration. *IEEE Trans. Ind. Inform.* **2023**, *19*, 11322–11332. [CrossRef]
95. Erbao, X.; Yan, L.; Mingshun, Y.; Xi, C. Design of Intelligent Monitoring System for Power Distribution Equipment Based on Cloud Edge Collaborative Computing. In Proceedings of the 2019 International Conference on Precision Machining, Non-Traditional Machining and Intelligent Manufacturing (PNTIM 2019), Xi'an, China, 22–24 November 2019. [CrossRef]
96. Sheng, Y.; Zhang, G.; Zhang, Y.; Luo, M.; Pang, Y.; Wang, Q. A Multimodal Data Sensing and Feature Learning-Based Self-Adaptive Hybrid Approach for Machining Quality Prediction. *Adv. Eng. Inform.* **2024**, *59*, 102324. [CrossRef]
97. Ren, C.; Li, M.; Chen, C.L.; Guan, X.P.; Huang, G.Q. Multi-Modal Digital Twins for Industrial Anomaly Detection: Framework, Method, and Application. *Robot. Comput.-Integr. Manuf.* **2026**, *97*, 103068. [CrossRef]
98. Zhang, C.; Dong, J.; Peng, K.X.; Zhang, H.W. Spatio-Temporal Information Analytics Based Performance-Driven Industrial Process Monitoring Framework with Cloud-Edge-Device Collaboration. *J. Manuf. Process.* **2024**, *110*, 224–237. [CrossRef]
99. Zhou, H. Research on Intelligent Manufacturing Data Collection and Transmission Optimization Methods Based on Cloud-Edge Collaboration. *Int. J. Hous. Sci. Its Appl.* **2025**, *46*, 2997–3010. [CrossRef]
100. Somu, N.; Sannagowdara Dasappa, N. An Edge-Cloud Iiot Framework for Predictive Maintenance in Manufacturing Systems. *Adv. Eng. Inform.* **2025**, *65*, 103388. [CrossRef]
101. Wu, T.; Tang, S.; Zhou, Q.; Chen, X. Cloud-Edge Synergistic MEC Multi-Task Offloading and Resource Allocation Method Based on Convex Optimization. *Clust. Comput.* **2025**, *28*, 781. [CrossRef]
102. Wang, J.; Yang, Y. Attention-Based High-Dimensional Offloading with Deep Recurrent Q-Network in a Cloud-Edge Environment. *J. Netw. Syst. Manag.* **2026**, *34*, 3. [CrossRef]
103. Zhang, T.R.; Wu, F.; Chen, Z.Y.; Chen, S.Y. Optimization of Edge-Cloud Collaborative Computing Resource Management for Internet of Vehicles Based on Multiagent Deep Reinforcement Learning. *IEEE Internet Things J.* **2024**, *11*, 36114–36126. [CrossRef]
104. Hu, L.; Sun, G.; Ren, Y. CoEdge: Exploiting the Edge-Cloud Collaboration for Faster Deep Learning. *IEEE Access* **2020**, *8*, 100533–100541. [CrossRef]
105. Li, X.H.; Chen, B.T.; Fan, J.C.; Kang, J.W.; Ye, J.; Wang, X.; Niyato, D. Cloud-Edge-End Collaborative Intelligent Service Computation Offloading: A Digital Twin Driven Edge Coalition Approach for Industrial IoT. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 6318–6330. [CrossRef]
106. Cui, H.S.; Tang, Z.Q.; Wu, Y.; Jia, W.J. Layer-Aware Cost-Effective Container Updates with Edge-Cloud Collaboration in Edge Computing. *IEEE Trans. Mob. Comput.* **2025**, *24*, 12314–12328. [CrossRef]
107. Wang, Y.; Yang, S.; Zhang, W.; Jiang, J.; Bai, Z.; Yang, H.; Fang, H. Cloud-Edge Cooperative Optimization Scheduling Strategy of Metro Integrated Supervision and Control System Based on Edge Computing. *Complex Intell. Syst.* **2025**, *11*, 361. [CrossRef]
108. Wang, Y.; Fang, J.J.; Cheng, Y.; She, H.; Guo, Y.A.; Zheng, G. Cooperative End-Edge-Cloud Computing and Resource Allocation for Digital Twin Enabled 6G Industrial IoT. *IEEE J. Sel. Top. Signal Process.* **2024**, *18*, 124–137. [CrossRef]
109. Yang, F.; Feng, T.; Xu, F.M.; Jiang, H.W.; Zhao, C.L. Collaborative Clustering Parallel Reinforcement Learning for Edge-Cloud Digital Twins Manufacturing System. *China Commun.* **2022**, *19*, 138–148. [CrossRef]
110. Sathupadi, K.; Achar, S.; Bhaskaran, S.V.; Faruqui, N.; Abdullah-Al-Wadud, M.; Uddin, J. Edge-Cloud Synergy for AI-Enhanced Sensor Network Data: A Real-Time Predictive Maintenance Framework. *Sensors* **2024**, *24*, 7918. [CrossRef] [PubMed]
111. Hu, Y.; Cao, N.; Lu, H.; Jiang, Y.Z.; Liu, Y.Q.; He, X.M. Multi-Dimensional Resource Management with Deep Deterministic Policy Gradient for Digital Twin-Enabled Industrial Internet of Things in 6 Generation. *Trans. Emerg. Telecommun. Technol.* **2024**, *35*, e4962. [CrossRef]
112. Liu, B.; Guo, J.; Li, C.; Luo, Y. Workload Forecasting Based Elastic Resource Management in Edge Cloud. *Comput. Ind. Eng.* **2020**, *139*, 106136. [CrossRef]

113. Yang, Y.Y.; Shi, Y.; Yi, C.Y.; Cai, J.; Kang, J.W.; Niyato, D.; Shen, X.M. Dynamic Human Digital Twin Deployment at the Edge for Task Execution: A Two-Timescale Accuracy-Aware Online Optimization. *IEEE Trans. Mob. Comput.* **2024**, *23*, 12262–12279. [CrossRef]
114. Wang, W.A.; Zhu, X.R. Reliable Multidimensional Resource Scheduling for Heterogeneous Computing Networks via Coded Distributed Computing and Hypergraph Neural Networks. *IEEE Internet Things J.* **2025**, *12*, 48242–48261. [CrossRef]
115. Zhu, L.; Lin, S.; Yu, F.R.; Li, Y. Collaborative Computing Optimization in Train-Edge-Cloud-Based Smart Train Systems Using Risk-Sensitive Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2024**, *73*, 3129–3141. [CrossRef]
116. Bozkaya, E.; Erel-Özçevik, M.; Bilen, T.; Özçevik, Y. Proof of Evaluation-based Energy and Delay Aware Computation Offloading for Digital Twin Edge Network. *Ad Hoc Netw.* **2023**, *149*, 103254. [CrossRef]
117. Lin, Z.W.; Chen, C.H.; Chen, J.Z.; Liu, Z.F. A Hierarchical Fractal Space NSGA-II-Based Cloud-Fog Collaborative Optimization Framework for Latency and Energy-Aware Task Offloading in Smart Manufacturing. *Mathematics* **2025**, *13*, 3691. [CrossRef]
118. Miao, Y.; Li, Y.; Pan, J.; Liu, Z.; Liu, L.; Wang, Z.; Wang, Z. Bio-Inspired Fault Diagnosis for Aircraft Fuel Pumps Using a Cloud-Edge System. *Biomimetics* **2023**, *8*, 601. [CrossRef]
119. Li, Z.; Zhao, Y.; Li, Y.; Liu, M.; Zeng, Z.; Xin, X.; Wang, F.; Li, X.; Zhang, J. Self-Optimizing Optical Network with Cloud-Edge Collaboration: Architecture and Application. *IEEE Open J. Comput. Soc.* **2020**, *1*, 220–229. [CrossRef]
120. Zhou, Z.; Yu, D.; Chen, M.; Qiao, Y.S.; Hu, Y.; He, W.W. An Edge Intelligence-Based Model Deployment Method for CNC Systems. *J. Manuf. Syst.* **2024**, *74*, 716–751. [CrossRef]
121. Jia, P.Y.; Wang, X.B.; Shen, X.M. Digital-Twin-Enabled Intelligent Distributed Clock Synchronization in Industrial IoT Systems. *IEEE Internet Things J.* **2021**, *8*, 4548–4559. [CrossRef]
122. Li, L.L.; Gu, F.; Li, H.; Guo, J.F.; Gu, X.J. Digital Twin Bionics: A Biological Evolution-Based Digital Twin Approach for Rapid Product Development. *IEEE Access* **2021**, *9*, 121507–121521. [CrossRef]
123. Shen, B.Q.; Yu, H.; Hu, P.; Cai, H.M.; Guo, J.Z.; Xu, B.Y.; Jiang, L.H. A Cloud-Edge Collaboration Framework for Generating Process Digital Twin. *IEEE Trans. Cloud Comput.* **2024**, *12*, 388–404. [CrossRef]
124. Nie, Q.W.; Tang, D.B.; Liu, C.C.; Wang, L.P.; Song, J.Y. A Multi-Agent and Cloud-Edge Orchestration Framework of Digital Twin for Distributed Production Control. *Robot. Comput.-Integr. Manuf.* **2023**, *82*, 102543. [CrossRef]
125. Chen, R.Y.; Yi, C.Y.; Zhou, F.H.; Kang, J.W.; Wu, Y.; Niyato, D. Federated Digital Twin Construction via Distributed Sensing: A Game-Theoretic Online Optimization with Overlapping Coalitions. *IEEE Trans. Mob. Comput.* **2025**, *24*, 12221–12238. [CrossRef]
126. Lei, M.; He, G.; Ouyang, Y.; Shan, P.; Wang, M. Cloud-Edge Fusion for Advanced Dynamic Digital Twin Models in Intelligent Distribution Networks. *J. Electr. Eng. Technol.* **2026**, *21*, 1037–1044. [CrossRef]
127. Yi, B.; Lv, J.H.; Wang, X.W.; Ma, L.B.; Huang, M. Digital Twin Driven and Intelligence Enabled Content Delivery in End-Edge-Cloud Collaborative 5G Networks. *Digit. Commun. Netw.* **2024**, *10*, 328–336. [CrossRef]
128. Xiang, F.; Zhou, S.; Zuo, Y.; Tao, F. Digital Twin Driven End-Face Defect Control Method for Hot-Rolled Coil with Cloud-Edge Collaboration. *IEEE Trans. Ind. Inform.* **2023**, *19*, 1674–1682. [CrossRef]
129. Ranpariya, A.; Sharma, S. A Digital Twin Solution for Fault Detection in Time-Critical IIoT Applications. *J. Simul.* **2025**, *19*, 441–454. [CrossRef]
130. Guan, W.Q.; Li, P.C.; Zhang, H.J.; Wu, Y. Integrating Generative AI with Network Digital Twin for 6G: An Edge-Cloud Collaborative Approach. *IEEE Commun. Mag.* **2025**, *63*, 181–187. [CrossRef]
131. Madani, S.S.; Shabeer, Y.; Fowler, M.; Panchal, S.; Chaoui, H.; Mekhilef, S.; Dou, S.X.; See, K. Artificial Intelligence and Digital Twin Technologies for Intelligent Lithium-Ion Battery Management Systems: A Comprehensive Review of State Estimation, Lifecycle Optimization, and Cloud-Edge Integration. *Batteries* **2025**, *11*, 298. [CrossRef]
132. Li, F.; Li, L.; Peng, Y. Research on Digital Twin and Collaborative Cloud and Edge Computing Applied in Operations and Maintenance in Wind Turbines of Wind Power Farm. In *Advances in Transdisciplinary Engineering*; Dobrotă, D., Cheng, C., Eds.; IOS Press: Amsterdam, The Netherlands, 2021. [CrossRef]
133. Wang, Y.T.; Feng, Y.H.; Xi, C.F.; Wang, B.C.; Tang, B.; Geng, Y.Z. Development of an Intelligent Coal Production and Operation Platform Based on a Real-Time Data Warehouse and AI Model. *Energies* **2024**, *17*, 5205. [CrossRef]
134. Chai, T.Y.; Xing, F.X.; Zheng, R.; Jia, Y.; Song, Y.J. An Anti-Latency Intelligent Control for 5G Wireless Networks Based on End-Edge-Cloud Collaboration. *IEEE J. Sel. Areas Commun.* **2025**, *43*, 3281–3294. [CrossRef]
135. Feng, H.; Qiao, L.; Lv, Z. Innovative Soft Computing-Enabled Cloud Optimization for next-Generation IoT in Digital Twins. *Appl. Soft Comput.* **2023**, *136*, 110082. [CrossRef]
136. Han, L.; Lei, M.; He, G.L.; Li, Y.Y.; Zhao, Y.P. Energy-Efficient Cloud-Edge Collaborative Model Integrating Digital Twins and Machine Learning for Scalable and Adaptive Distributed Networks. *Sustain. Comput. Inform. Syst.* **2025**, *47*, 101157. [CrossRef]
137. Li, T.Y.; Wang, X.W.; Zeng, R.F.; Zhao, L.; Hawbani, A.; Zhang, Y.X.; Huang, M. Enhancing Edge-Cloud Collaboration with Blockchain-Assisted Digital Twin Intelligence Offloading Scheme. *IEEE Trans. Mob. Comput.* **2025**, *24*, 9619–9635. [CrossRef]
138. Yang, F.; Xu, F.M.; Feng, T.; Qiu, C.; Zhao, C.L. pDPoStsPBFT: A High Performance Blockchain-Assisted Parallel Reinforcement Learning in Industrial Edge-Cloud Collaborative Network. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 2744–2759. [CrossRef]

139. Yang, H.; Ong, S.K.; Nee, A.Y.C.; Jiang, G.; Mei, X. Microservices-Based Cloud-Edge Collaborative Condition Monitoring Platform for Smart Manufacturing Systems. *Int. J. Prod. Res.* **2022**, *60*, 7492–7501. [CrossRef]
140. Zhang, Y.; Tang, D.B.; Zhu, H.H.; Zhou, S.H.; Zhao, Z. An Efficient IIoT Gateway for Cloud-Edge Collaboration in Cloud Manufacturing. *Machines* **2022**, *10*, 850. [CrossRef]
141. Hastbacka, D.; Halme, J.; Barna, L.; Hoikka, H.; Pettinen, H.; Larranaga, M.; Bjorkbom, M.; Mesia, H.; Jaatinen, A.; Elo, M. Dynamic Edge and Cloud Service Integration for Industrial IoT and Production Monitoring Applications of Industrial Cyber-Physical Systems. *IEEE Trans. Ind. Inform.* **2022**, *18*, 498–508. [CrossRef]
142. Fang, L.; Wan, J.F.; Cai, H.; Wang, S.Y.; Pang, Z.B.; Safran, M.; Alqahtani, S.A. A Scalable Cloud-Edge Collaborative Approach for Intelligent Low-Code Fault Diagnosis: Successful Applications of Agile Migration Deployment in Heterogeneous Fault Diagnosis Scenarios. *IEEE Ind. Electron. Mag.* **2024**, *18*, 16–29. [CrossRef]
143. Hu, P.F.; He, C.M.; Zhu, Y.M.; Li, T.H. The Product Quality Inspection Scheme Based on Software-Defined Edge Intelligent Controller in Industrial Internet of Things. *J. Cloud Comput.-Adv. Syst. Appl.* **2023**, *12*, 113. [CrossRef]
144. Byrtus, R. Methodology for Condition Monitoring of Industrial Systems Based on Edge Computing. Ph.D. Thesis, Technical University of Ostrava, Ostrava, Czech Republic, 2024.
145. Wang, Z.J.; Wang, C.Z.; Yang, C.H. Cloud-Edge Collaborative Control for Roaster Furnace: A Hybrid First-Principles and Data-Driven Method. *J. Ind. Inf. Integr.* **2023**, *36*, 100531. [CrossRef]
146. Wang, W.B.; Hu, T.T.; Gu, J.A. Edge-Cloud Cooperation Driven Self-Adaptive Exception Control Method for the Smart Factory. *Adv. Eng. Inform.* **2022**, *51*, 101493. [CrossRef]
147. Ruan, J.; Yang, X.H.; Fan, Y. Collaborative and Observability Enhanced Fault Injection for Resilient Cloud-Edge Computing. *Int. J. Softw. Eng. Knowl. Eng.* **2025**, *35*, 1623–1650. [CrossRef]
148. Wan, J.F.; Li, X.M.; Dai, H.N.; Kusiak, A.; Martinez-Garcia, M.; Li, D. Artificial-Intelligence-Driven Customized Manufacturing Factory: Key Technologies, Applications, and Challenges. *Proc. IEEE* **2021**, *109*, 377–398. [CrossRef]
149. Lou, P.; Liu, S.; Hu, J.; Li, R.; Xiao, Z.; Yan, J. Intelligent Machine Tool Based on Edge-Cloud Collaboration. *IEEE Access* **2020**, *8*, 139953–139965. [CrossRef]
150. Jiang, W.; Hu, F. Artificial Intelligence Agent-Enabled Predictive Maintenance: Conceptual Proposal and Basic Framework. *Computers* **2025**, *14*, 329. [CrossRef]
151. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*; PMLR: Cambridge, MA, USA, 2017; pp. 1273–1282.
152. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
153. Hinton, G.; Vinyals, O.; Dean, J. Distilling the Knowledge in a Neural Network. *arXiv* **2015**, arXiv:1503.02531. [CrossRef]
154. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018.
155. Glaessgen, E.; Stargel, D. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2012; p. 1818. [CrossRef]
156. Ji, C.; Luo, H. Cloud-Based AI Systems: Leveraging Large Language Models for Intelligent Fault Detection and Autonomous Self-Healing. *arXiv* **2025**, arXiv:2505.11743. [CrossRef]


**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.