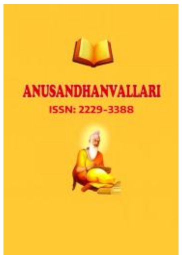

Anusandhanvallari
Vol 2023, No.1
August 2023
ISSN 2229-3388

Deterministic Reflexes and Stochastic Planning for Decoupling Control and Learning in Resource Constrained Edge Offloading

Amit Malik[1], Amita Rani[2]
[1]Department of Computer Science and Engineering, SRM University, Delhi-NCR, Sonepat, Haryana, India.
Email: malik.dcrust@gmail.com, ORCID: 0009-0008-0407-7183
[2]Department of Computer Science and Engineering, DCRUST, Murthal, Sonepat, Haryana, India.
Email: amitamalik.cse@dcrustm.org, ORCID: 0000-0002-7385-4045

Abstract: Current edge computing research highly favors complex deep learning models for resource management. However, these data-heavy approaches often conflict with the physical reality of edge devices, which demand low latency, low energy usage, and absolute predictability. This review critically examines the trade-offs between stochastic learning strategies and lightweight, deterministic alternatives. By analyzing recent literature, we highlight that simple, rule-based logic frequently outperforms complex learning algorithms in stability and response time. The findings suggest a need to shift design priorities away from unnecessary complexity and toward sufficiency, prioritizing reliability over raw intelligence at a high operational cost.

Keywords: Edge Computing, Task Offloading, Nature Inspired, Deterministic, Machine Learning.

Introduction

For the better part of the last decade, the field of edge computing has been engaged in a race to squeeze as much "intelligence" as possible into smaller and smaller devices. The prevailing assumption which is evident in hundreds of recent studies is that if a system is not learning, it is not smart. This trend has pushed the design of edge offloading strategies toward increasingly complex models, particularly those based on Deep Reinforcement Learning (DRL) and heavy neural networks [1]. The narrative is attractive: an autonomous system that learns from its environment, predicts the future, and optimizes itself without human intervention. However, this pursuit of complexity often ignores the harsh reality of the physical world where these devices actually live. When we deploy systems to monitor oil rigs, control autonomous drones in windstorms, or manage traffic in congested cities, the primary requirement is not that the system is "smart" in a human sense, but that it is predictable, stable, and fast [2], [3].
The central problem this paper addresses is the mismatch between the tools we are currently obsessed with (complex learning algorithms) and the problems we are trying to solve (reliable operation under constraint). Edge computing is fundamentally defined by limitations: limited battery life, limited processing power, and, most critically, limited time to make a decision [4]. A drone recovering from a sudden gust of wind does not have the luxury of running a complex inference model to decide which motor to throttle; it needs a hard, deterministic rule that executes in microseconds. Yet, the literature is flooded with proposals to replace these efficient control loops with stochastic learning models that require massive amounts of data to train and significant energy to run. While these models perform well in simulations or cloud environments with infinite resources, they frequently fail the "sufficiency" test in the real world. They introduce unpredictability, in the sense, that the same input does not always guarantee the same output, specially into systems which require safety and auditability above other things [5].
This review propagates that determinism needs to be considered as a deliberate design philosophy. Determinism in this context does not mean rigid methods but it means "structured" way of solving the problem. A deterministic system is one where the behavior is explicitly defined to meet specific constraints. Upon examining the successful biological systems, which are often cited as the inspiration for AI, it can be observed that they are not purely "learning" machines. An insect does not learn how to fly from scratch every time it takes off, rather it relies on pre-wired, deterministic reflexes that are highly efficient and incredibly robust. Similarly, effective edge systems should rely on lightweight, rule-based, or threshold-based logic for their core operations, reserving expensive learning models only for high-level tasks where they are strictly necessary [6].

Anusandhanvallari
Vol 2023, No.1
August 2023
ISSN 2229-3388

The motivation for this shift is practical, not just theoretical. In safety-critical environments, such as industrial automation or healthcare monitoring, the operators need to know exactly why a system made a specific decision. If a robotic arm halts production, an engineer should be able to debug a deterministic threshold violation e.g. Temperature > 80°C. Comparatively, debugging a neural network's "black box" decision is significantly harder and often impossible in real-time [7]. Furthermore, the energy cost of training and running these models on edge devices contradicts the "green computing" goals often supported by the same researchers proposing them. These techniques are effectively burning battery life to solve problems that could be solved with simple control theory or heuristic algorithms [8].

Therefore, the objective of this paper is to validate the role of sufficiency in system design. The paper defines "sufficiency" as the ability of a model to meet the operational requirements of a task without exceeding the resource budget or introducing unnecessary complexity [9]. By analyzing recent literature, it shows that simple, deterministic offloading strategies often outperform their complex learning-based counterparts in metrics that matter most to the edge: latency, jitter, and energy consumption. We will explore how "lightweight" does not mean "less capable," but rather "better fitted" to the environment [10].

This paper is organized to guide the reader from the physical constraints of the hardware back to the philosophy of design. It begins by establishing the hard requirements of edge environments that make standard AI approaches difficult to deploy. It then survey the existing lightweight and deterministic offloading models, highlighting their performance benefits. Following this, the paper examines the specific limitations of learning-based strategies, identifying where they break down in practice. Finally, the paper look into other disciplines like biology and control theory to show that high-functioning systems in nature and engineering increasingly favor structure and determinism over unbounded learning.

Learning-Based Edge Orchestration

To understand the necessity of deterministic designs, it is necessary to first examine the current trajectory of edge computing research over the last decade. The field has largely focused on one main idea that networks are becoming more complex, and the software managing them must also be complex [6], [11]. As a result, Machine Learning (ML) and Deep Reinforcement Learning (DRL) have shifted from being optional tools to becoming the default methods for resource orchestration. A review of current literature reveals a pervasive assumption that a "smart" system must be a "learning" system. This trend is clear in hundreds of recent studies that propose using DRL agents for everything from dynamic task offloading and channel allocation to voltage scaling and drone swarm coordination [1, 5, and 12]. Unlike static heuristics, these learning agents promise to handle high-dimensional state spaces without requiring an explicit mathematical model of the environment, theoretically allowing systems to adapt to unseen user behaviors autonomously.

However, this push for complexity often obscures a critical mismatch. The computationally heavy tools being deployed frequently do not fit the physical reality of the resource constrained devices they operate on. While these learning strategies work well in cloud-based simulations where resources are effectively infinite and latency is a soft constraint, they face major practical challenges when deployed in physical edge environments. The primary friction point lies in the definition of "performance" [4]. In the cloud, performance is typically measured by combined metrics, such as, average throughput or mean response time. In contrast, the edge is defined by strict physical bounds and "hard" constraints [14]. The literature indicates that for mission-critical applications such as industrial automation (Industry 4.0), Vehicle-to-Everything (V2X) communication, and tactile internet services, the primary requirement is not maximizing the average reward, but guaranteeing stability under worst-case conditions [15].This conflict is most visible in the trade-off between inference accuracy and temporal determinism. Deep learning models, by their nature, introduce significant variance in execution time which is a phenomenon known as inference jitter. The execution path through a deep neural network can vary based on the sparseness of the input data, and the underlying software frameworks often trigger unpredictable garbage collection cycles or memory swapping events [16]. For a control loop in a robotic arm that requires a decision every 10 milliseconds, a DRL agent that usually responds in 2 milliseconds but occasionally spikes to 50 milliseconds is functionally unacceptable [17]. Research highlights that while stochastic

Anusandhanvallari
Vol 2023, No.1
August 2023
ISSN 2229-3388

policies may statistically optimize long-term network utility, they often fail to provide the Worst-Case Execution Time guarantees required for safety-critical certification. In these contexts, a simpler, deterministic threshold-based algorithm, which offers $O(1)$ decision complexity and zero variance, is objectively superior despite being "less intelligent" in the academic sense [18]. Apart from latency, the energy economics of learning-based strategies present a paradox for battery-constrained devices. Mobile edge nodes, such as UAVs or remote IoT sensors, operate in a zero-sum energy environment where every joule spent on decision-making is a joule taken away from the actual application or data transmission. The literature often highlights the "training tax" which introduces the significant computational cost required to keep a learning model relevant. Federated Learning (FL) and online DRL approaches require continuous model updates, back-propagation, and gradient exchanges to adapt to changing environments.

Recent energy audits of these systems reveal that for many transient or short-lived tasks, the energy consumed by the orchestration layer, for training and inference, exceeds the energy saved by the resulting optimization. Battery life is effectively being wasted to solve an optimization problem that could have been approximated by a greedy heuristic at a fraction of the cost. Sustainable "Green Edge" computing demands a careful utilization of resources by algorithms, prioritizing logic that maximizes utility per watt rather than logic that maximizes theoretical convergence [19], [20].

Furthermore, the "black box" nature of deep neural networks creates a significant barrier to auditability and trust. In regulated industries like healthcare monitoring, smart grids, and autonomous transport, system operators require causal traceability which is the ability to explicitly reconstruct why a system made a specific decision [1]. If a smart grid controller fails to load-balance a surge, resulting in a blackout, engineers must be able to debug the logic. If the decision was made by a neural network, the "reasoning" is buried in millions of floating-point weights, making it mathematically non-trivial and operationally impossible to diagnose in real-time. Deterministic systems, governed by explicit algebraic formulations or rule sets, offer "correctness by construction." This transparency is not merely a feature but a fundamental requirement for liability and safety certification [21].

Finally, there is the issue of "generalization bias." The academic community often prioritizes models that can generalize to any possible scenario, leading to massive, over-parameterized architectures. However, edge devices typically operate in highly specific, bounded environments, such as, a specific factory floor, a specific traffic intersection, or a specific patient monitor. They do not need to solve every problem; they only need to solve their problem reliably. The drive to deploy heavy "general purpose" AI on "specific purpose" hardware results in unnecessary overhead. The evidence suggests that "sufficiency", which is the ability to meet requirements with the simplest possible model, should replace "generalization" as the gold standard for edge design. By adhering to this principle, it becomes evident that the limitations of learning-based approaches are not temporary hurdles to be overcome by better hardware, but fundamental indications that may be the wrong tool is being used for the job. The subsequent sections will explore how deterministic, lightweight alternatives effectively address these constraints where heavier models struggle [6], [22].

Lightweight And Deterministic Offloading Strategies

In response to the operational constraints detailed in the previous section, a significant portion of research has persisted in developing offloading strategies that prioritize structure over learning. Unlike the stochastic models that dominate recent headlines, these approaches are characterized by their determinism, according to which, given a specific state (e.g., queue length, battery level), the system produces a fixed, predictable output. This section reviews the three primary categories of lightweight orchestration, i.e., Threshold-Based Heuristics, Control-Theoretic Feedback Loops, and Bio-Inspired Algorithms, highlighting their ability to maintain system stability with minimal computational overhead [23].

The most fundamental form of deterministic control relies on conditional logic triggered by system states. These strategies, often referred to in the literature as "Lyapunov-guided" or "Greedy" heuristics, operate on the principle that complex global behavior can be regulated through simple local constraints [24]. The mechanism is straightforward and states that an offloading decision is triggered only when a specific metric, like, local CPU utilization or buffer occupancy, violates a pre-defined safety margin. Recent work demonstrates that these simple logic engines are often sufficient for high-speed edge environments [25]. For instance, studies comparing Lyapunov optimization against

Deep Reinforcement Learning (DRL) in multi-user scenarios found that while DRL agents required thousands of training epochs to converge, threshold-based algorithms achieved near-optimal queue stability instantly upon deployment [26]. The primary advantage here is computational complexity. A threshold check operates in $O(1)$ time, requiring only a single comparison operation. This stands in stark contrast to the matrix multiplications required for neural inference. For ultra-low-latency applications, such as 5G network slicing or tactile internet services, this difference is critical. When the deadline for a task is under 5 milliseconds, the time spent deciding where to send the task must be negligible. Threshold-based heuristics effectively reduce the "decision cost" to zero, ensuring that the bulk of the resource budget is spent on execution rather than orchestration [26].

Moving beyond static rules, control theory offers a mathematical framework for dynamic adaptation without the opacity of neural networks. PID (Proportional-Integral-Derivative) controllers, which have governed industrial automation for decades, are increasingly being adapted for edge resource management. These systems maintain a target state (e.g., a specific energy consumption rate or average response time) by continuously adjusting control variables (e.g., CPU frequency or transmission power) based on the error difference between the current state and the target [7]. The distinguishing feature of control-theoretic approaches is the presence of formal stability guarantees. Unlike Reinforcement Learning agents, which must "explore" the environment to learn optimal policies, control loops are mathematically proven to converge to a stable state provided the system parameters remain within designed bounds [6]. This "bounded stability" makes them ideal for safety-critical edge sectors. For example, in a smart grid managing voltage fluctuations, a PID controller provides a traceable, predictable response to load changes. If the system oscillates or fails, the failure can be analyzed using standard control engineering methods (e.g., examining the gain margins), whereas a failure in a neural network is often indistinguishable from random noise. By treating edge offloading as a fluid control problem rather than a pattern recognition problem, these strategies achieve robust performance with negligible processing overhead [7], [12].

The third category of lightweight strategies looks to biological systems, in this case, not for their capacity to learn, but for their efficiency in decentralized coordination. It is observed that nature rarely uses centralized optimization and instead, complex global order emerges from simple, deterministic local interactions. Swarm Intelligence algorithms inspired by insect colony behavior or homeostatic regulation allow distributed edge nodes to self-organize using rigid local rules [1], [14]. A key example found in the literature is the use of "pheromone-based" routing or "virtual forces" for task migration. In these models, a task is treated like a particle moving through a field, attracted to nodes with high energy and repelled by nodes with high latency. Crucially, these bio-inspired models are often stripped of the stochastic elements found in evolutionary optimization, like Genetic Algorithms, to ensure predictability. Recent implementations of Homeostatic control, in which a node automatically sheds load to neighbors to maintain healthy state of its thermal status, have shown remarkable resilience. These systems function similarly to the human body's temperature regulation; they do not need to "predict" a fever to treat it, they simply react to the thermal deviation. This reactive capability allows for highly scalable networks where nodes can be added or removed without retraining a central model. The literature confirms that for large-scale Industrial IoT (IIoT) networks, these bio-inspired deterministic methods often outperform centralized DRL schedulers in terms of network longevity and partition tolerance [14], [26].

The common thread uniting these approaches is the concept of sufficiency. They do not attempt to solve the general case of all possible network states. Rather, they are designed to handle specific, bounded operational constraints. By restricting the solution space to simple rules, control loops, and bio-mimetic reactions, these models trade "general intelligence" for "specific reliability." The evidence suggests that in the resource-starved environment of the edge, this trade-off is not a compromise but an optimization. The elimination of the training phase, the reduction of inference jitter, and the guarantee of auditability make these lightweight strategies the pragmatic choice for systems that must work correctly the first time, and every time [27]. Finally, determinism must be viewed through the lens of engineering ethics. As edge computing is made up of critical infrastructure, the cost of a "black box" error increases. If an algorithm is too complex to be fully understood by its operator, it introduces a layer of moral hazard. Deterministic sufficiency restores the chain of accountability. It asserts that one should understand the boundaries of the system they deploy. By prioritizing models that are "correct by construction" rather than "correct by training," the industry moves toward a standard of explainable reliability, ensuring that the systems governing our physical world are transparent, auditable, and fundamentally safe [1], [40].

Model Selection

The critique of learning-based strategies presented in this review does not suggest that Machine Learning (ML) has no place in edge computing. Rather, it argues against its omnipresence. A mature design philosophy requires a shift from "AI-for-Everything" to a stratified approach, where model complexity is matched to the specific temporal and safety requirements of the layer it occupies. This section outlines a framework for hybrid architectures and analyzes the specific domains where deterministic sufficiency offers a clear advantage over stochastic generalization.

A. The Control Plane

Biological systems offer a compelling template for resolving the tension between responsiveness and intelligence. In vertebrate biology, high-level planning (e.g., hunting strategy) is handled by the brain, while immediate survival actions (e.g., pulling a hand away from fire) are handled by the spinal cord via reflex arcs. These reflexes do not "learn" in real-time; they are pre-wired, deterministic, and instant [36]. Edge architectures should mirror this kind of "Reflex-Brain" split into orchestration and resource planning [28].

Functions requiring sub-millisecond responses, such as, packet routing, safety braking, or voltage regulation, must remain purely deterministic. Threshold-based heuristics and PID controllers are the optimal choices here, providing the "spinal" stability required for preventing the system failure [7]. On the other hand, the functions operating on longer timescales, such as, predicting weekly traffic patterns or optimizing energy prices, can use Deep Reinforcement Learning (DRL). These models can reside in the upper tier (Cloud or Fog), sending down updated parameters (e.g., tuning the PID gains) without interfering with the real-time control loop. This separation ensures that the unpredictability of learning never compromises the safety of the immediate action [29].

B. Domain-Specific Suitability

The utility of deterministic models varies significantly across different edge sectors. A review of the application-layer literature highlights three domains where the "sufficiency" of deterministic logic outweighs the "optimality" of DRL. The manufacturing environments in Industrial IoT (Industry 4.0), establishes that the primary metric is synchronization, not throughput. A robotic assembly line requires multiple actuators to move in perfect lockstep. Stochastic jitter introduced by a neural network inference engine can desynchronize the line, leading to physical damage. Consequently, recent implementations have favored Time-Sensitive Networking (TSN) scheduled by strict, deterministic cyclic algorithms over dynamic ML schedulers [30].

For Autonomous Vehicles (V2X), the vision systems of autonomous cars rely heavily on Deep Learning, the decision-logic for maneuvering (e.g., "Do not merge if distance < 5m") is increasingly reverting to rule-based constraints. This "Safety Cage" architecture ensures that no matter what the AI "hallucinates," the vehicle cannot violate the laws of physics or traffic safety. Here, determinism acts as the ultimate auditor of the learning model [31].

Strategy Family	Type	Worst-Case Latency	Energy Overhead	Auditability	Best Use Case
Deep RL [1], [12], [17], [20]	Stochastic	High Variance (Jitter)	High (Training + Inference)	Black Box	Cloud/Fog Planning, Traffic Prediction
Genetic / Evolutionary [4], [6], [27]	Randomized	Unbounded	Medium (Iterative)	Complex	Offline Optimization, Design Exploration
Threshold Heuristics [18], [25],[30]	Deterministic	Bounded ($O(1)$)	Very Low	Full Traceability	Safety Interrupts, Routing, Voltage Control
PID / Control Theory [5], [23]	Deterministic	Bounded ($O(1)$)	Low	Mathematical Proof	Resource Scaling, Thermal Management
Bio-Inspired [2], [24]	Deterministic	Low Variance	Low	Local Rules	Distributed Load Balancing, Swarms

The implantable devices (e.g., pacemakers), or the critical monitors used in healthcare, the energy cost of on-device training is prohibitive. Furthermore, the regulatory requirement for auditability (FDA/CE certification) favors algorithms that can be formally verified. Bio-inspired homeostatic models, which simply maintain physiological metrics within a "healthy" range, provide the necessary reliability without the "black box" liability of neural networks [26].

C. Open Challenges in Deterministic Design

Despite the clear advantages of lightweight strategies, significant challenges remain in their standardization. Unlike the ML ecosystem, which benefits from massive frameworks like TensorFlow and PyTorch, [32] the ecosystem for deterministic edge control is fragmented. There is currently no standard library for "Bio-Inspired Offloading" or "Control-Theoretic Resource Management." This tooling gap forces researchers to implement these complex-yet-lightweight models from scratch, often leading to poor reproducibility. Future work in this field must focus on developing standardized "lightweight control primitives" which are reusable, modular logic blocks that allow developers to deploy deterministic strategies as easily as they currently deploy neural networks [33] approach.
Table 1 provides an empirical justification for the different type of architectures mentioned above, illustrating the distinct operational profiles of stochastic versus deterministic strategies. The comparison reveals a clear divsion in suitability using the "Reflex-Planning" split approach. Learning-based approaches, such as Deep RL, incur high energy overhead and latency jitter, effectively disqualifying them from the real-time reflex layer despite their planning capabilities. In contrast, deterministic mechanisms, such as, spanning threshold heuristics, PID loops, and bio-inspired models, offer the necessary guarantees of bounded execution time ($O(1)$) and complete traceability. This analysis confirms that while heavy learning models are best utilized for long-term optimization in the cloud, the lightweight deterministic strategies possess the stability and efficiency required for immediate, safety-critical control at the edge.

Conclusion

This review has critically examined the existing trend of edge resource management, arguing that the research community's intense focus on complex Deep Learning and Reinforcement Learning models has created a fundamental misalignment with the operational realities of edge hardware. While the theoretical appeal of autonomous, general-purpose learning agents is undeniable for cloud-based optimization, our analysis demonstrates that these stochastic approaches often fail to meet the strict physical constraints of the edge. The comparative analysis highlighted in this study reveals that the "intelligence" of a model cannot be decoupled from its operational cost. As summarized in the supporting comparisons, learning-based strategies incur unacceptable penalties in terms of inference jitter, energy consumption ("training tax"), and lack of traceability of decisions. For real time applications in industrial IoT, autonomous mobility, and healthcare, these are not merely technical hurdles but disqualifying characteristics. In contrast, lightweight deterministic strategies like threshold heuristics, PID control loops, and bio-inspired homeostatic mechanisms are shown to provide the necessary bounded execution times ($O(1)$), minimal energy overhead, and "correct-by-construction" transparency required for real-time safety. Ultimately, this paper advocates for a shift in design philosophy away from the pursuit of unbounded generalization and toward the principle of sufficiency. The future of robust edge architecture does not lie in forcing every node to be a learner, but rather in a stratified approach that recognizes distinct operational planes. By reserving heavy, stochastic learning models for high-level planning in the cloud and relying on rigorous, deterministic logic for immediate "reflex" actions at the edge, designers can achieve systems that are both adaptive over the long term and inherently stable in real-time. The truly intelligent edge is one that knows when to learn, and more importantly, when to simply follow the rules.

References

[1] Y. Hao, Y. Jiang, M. S. Hossain, M. F. Alhamid, and S. U. Amin, "Learning for Smart Edge: Cognitive Learning-Based Computation Offloading," *Mobile Networks and Applications*, Jun. 2020, doi: 10.1007/S11036-018-1119-7.

[2] N. TaheriNejad, A. Herkersdorf, and A. Jantsch, "Autonomous Systems, Trust and Guarantees," *IEEE Design & Test of Computers*, Sep. 2020, doi: 10.1109/MDAT.2020.3024145.

[3] K. Hoffpauir *et al.*, "A Survey on Edge Intelligence and Lightweight Machine Learning Support for Future Applications and Services," *Journal of Data and Information Quality*, Jan. 2023, doi: 10.1145/3581759.

[4] E. Nueve, S. Shahkarami, S. Park, and N. J. Ferrier, "Addressing the Constraints of Active Learning on the Edge," in *International Parallel and Distributed Processing Symposium*, Jun. 2021. doi: 10.1109/IPDPSW52791.2021.00126.

[5] A. A. Al Makdah, V. Katewa, and F. Pasqualetti, "Accuracy Prevents Robustness in Perception-based Control," in *Advances in Computing and Communications*, Jul. 2020. doi: 10.23919/ACC45564.2020.9147981.

[6] "Edge AI: Addressing the Efficiency Paradigm," Nov. 2022. doi: 10.1109/metrocon56047.2022.9971140.

[7] H. D. Doran, "Security and Safety Aspects of AI in Industry Applications," *arXiv.org*, Jul. 2022, doi: 10.48550/arXiv.2207.10809.

[8] M. Ahmed and W. Ahmed, "State-of-the-Art Power Management Techniques," 2022. doi: 10.1007/978-981-16-3346-1_18.

[9] A. Artemenko, I. Mehrez, K. Govindaraj, A. Kirstaedter, and M. Kuznietsov, "Empirical Investigation of Offloading Decision Making in Industrial Edge Computing Scenarios," in *European Conference on Networks and Communications*, Jun. 2021. doi: 10.1109/EUCNC/6GSUMMIT51104.2021.9482556.

[10] R. Hossen and M. A. Islam, "Mobile Task Offloading Under Unreliable Edge Performance," May 2021, doi: 10.1145/3466826.3466838.

[11] B. Varghese *et al.*, "Revisiting the Arguments for Edge Computing Research," *IEEE Internet Computing*, Sep. 2021, doi: 10.1109/MIC.2021.3093924.

[12] A. Alwarafy, M. M. Abdallah, B. S. Ciftler, A. Al-Fuqaha, and M. Hamdi, "The Frontiers of Deep Reinforcement Learning for Resource Management in Future Wireless HetNets: Techniques, Challenges, and

Research Directions," *IEEE open journal of the Communications Society*, Jan. 2022, doi: 10.1109/ojcoms.2022.3153226.
[13] K. Todi, G. Bailly, L. A. Leiva, and A. Oulasvirta, "Adapting User Interfaces with Model-based Reinforcement Learning," in *Human Factors in Computing Systems*, May 2021. doi: 10.1145/3411764.3445497.
[14] M. S. Aslanpour, M. S. Aslanpour, S. S. Gill, and A. N. Toosi, "Performance Evaluation Metrics for Cloud, Fog and Edge Computing: A Review, Taxonomy, Benchmarks and Standards for Future Research," Dec. 2020, doi: 10.1016/J.IOT.2020.100273.
[15] N. A. Mohammed, A. M. Mansoor, and R. Ahmad, "Mission-Critical Machine-Type Communication: An Overview and Perspectives Towards 5G," *IEEE Access*, Jan. 2019, doi: 10.1109/ACCESS.2019.2894263.
[16] S. Liu *et al.*, "On Removing Algorithmic Priority Inversion from Mission-critical Machine Inference Pipelines," in *Real-Time Systems Symposium*, Dec. 2020. doi: 10.1109/RTSS49844.2020.00037.
[17] G. T. T. Bernardo, G. Miranda Jr., and D. F. Macedo, "Analysis of Network Performance over Deep Reinforcement Learning Control Loops for Industry 4.0," in *Brazilian Symposium on Computer Networks and Distributed Systems*, May 2022. doi: 10.5753/sbrc.2022.221898.
[18] M. Korun, B. Vodenik, and B. Zorko, "An alternative approach to the decision threshold," *Applied Radiation and Isotopes*, Jul. 2017, doi: 10.1016/J.APRADISO.2017.07.038.
[19] N. Shalavi, G. Perin, A. Zanella, and M. Rossi, "Energy Efficient Deployment and Orchestration of Computing Resources at the Network Edge: a Survey on Algorithms, Trends and Open Challenges," *arXiv.org*, Sep. 2022, doi: 10.48550/arXiv.2209.14141.
[20] "Enabling Deep Reinforcement Learning on Energy Constrained Devices at the Edge of the Network," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2022. doi: 10.1109/wcnc51071.2022.9771901.
[21] H. Forsberg, J. Linden, J. Hjorth, T. Manefjord, and M. Daneshtalab, "Challenges in Using Neural Networks in Safety-Critical Applications," in *Document Analysis Systems*, Oct. 2020. doi: 10.1109/DASC50938.2020.9256519.
[22] "Brief overview of Edge AI accelerators for energy-constrained edge," Mar. 2022. doi: 10.1109/sami54271.2022.9780669.
[23] M. Avgeris, D. Dechouniotis, N. Athanasopoulos, and S. Papavassiliou, "Adaptive Resource Allocation for Computation Offloading: A Control-Theoretic Approach," *ACM Transactions on Internet Technology*, Apr. 2019, doi: 10.1145/3284553.
[24] S. Maignan, C. Bernon, and P. Glize, "Speeding up the Search of a Global Dynamic Equilibrium from a Local Cooperative Decision," in *International Conference on Agents and Artificial Intelligence*, Mar. 2018. doi: 10.5220/0006540101430150.
[25] L. Schonberger *et al.*, "Offloading Safety- and Mission-Critical Tasks via Unreliable Connections," in *Euromicro Conference on Real-Time Systems*, Jan. 2020. doi: 10.4230/LIPICS.ECRTS.2020.18.
[26] "No Free Lunch: Balancing Learning and Exploitation at the Network Edge," in *ICC 2022 - IEEE International Conference on Communications*, May 2022. doi: 10.1109/icc45855.2022.9838277.
[27] P. Radanliev and D. De Roure, "Review of Algorithms for Artificial Intelligence on Low Memory Devices," *IEEE Access*, Aug. 2021, doi: 10.1109/ACCESS.2021.3101579.
[28] T. Rausch and S. Dustdar, "Edge Intelligence: The Convergence of Humans, Things, and AI," in *IEEE International Conference on Cloud Engineering*, Jun. 2019. doi: 10.1109/IC2E.2019.00022.
[29] D. Baskakov and V. P. Shkodyrev, "Deep Predictive Control," 2021. doi: 10.1007/978-981-33-6632-9_29.

[30] Y. Lu *et al.*, "An Intelligent Deterministic Scheduling Method for Ultralow Latency Communication in Edge Enabled Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, Jul. 2022, doi: 10.1109/TII.2022.3186891.

[31] S. Nair, S. Shafaei, S. Kugele, M. H. Osman, and A. Knoll, "Monitoring Safety of Autonomous Vehicles with Crash Prediction Networks," in *National Conference on Artificial Intelligence*, Jan. 2019.

[32] G. Nguyen *et al.*, "Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey," *Artificial Intelligence Review*, Jan. 2019, doi: 10.1007/S10462-018-09679-Z.

[33] Y. Gong, H. Yao, J. Wang, D. Wu, N. Zhang, and F. R. Yu, "Decentralized Edge Intelligence-Driven Network Resource Orchestration Mechanism," *IEEE Network*, Jan. 2022, doi: 10.1109/mnet.120.2200086.