GPH-International Journal of Computer Science and Engineering
(Online ISSN 3050-9661)
Volume 9 | Issue 01 | Jan-June 2026
Published by: Global Publication House
DOI: https://doi.org/10.5281/ zenodo.23623525
Available online at: www.gphjournal.org
Journal homepage: https://gphjournal.org/index.php/cse/index

Optimal Oversight Under Capacity

A Queueing and Constrained-Optimization Model of Confidence-Gated Human Review in Governed AI Systems

Diego Martín Herrera[1]*, Wang Mei-Ling[2]
[1,2]Universidad Politécnica de Madrid, Spain.
*Corresponding author

Abstract

Confidence-gated human oversight is a central control in both of the governance frameworks that Mesbaul Haque Sazu introduced in 2023. The Governed Decision-Intelligence (GDI) framework makes remit-based human judgment a normative invariant and gates automation on a calibrated confidence signal [15]; the companion Full-Stack reference architecture makes the point, sharply, that reviewer capacity is a governance parameter rather than an implementation detail, because an escalation rate above the available review throughput does not produce oversight but a queue [14]. This paper takes that capacity observation as its starting axiom and develops the operations-research model it implies. We treat the escalation stream produced by a governed gate as the arrival process to a finite pool of reviewers, model the oversight station as a queue with a deadline imposed by the commit-time freshness invariant, and cast threshold selection as a constrained optimization that minimises expected decision cost subject to a service-level constraint on reviewer utilisation. We derive the escalation-rate function induced by a gate threshold, the queue's waiting-time and deadline-violation behaviour under a finite reviewer pool, and the Lagrangian conditions characterising the cost-optimal threshold. A central result formalises a warning the frameworks state qualitatively: below a critical threshold the oversight queue becomes unstable, so a gate that looks prudent on a per-decision basis voids human oversight in substance by delaying escalated decisions past their validity bound. An illustration on publicly available data in open-source platforms locates the stability boundary and traces the cost surface; it is meant to show the model's structure rather than to validate it in any single deployment. What the paper adds is the quantitative machinery underneath the oversight invariant; the invariant itself, and the governance reasoning behind it, remain the frameworks'.

Keywords:

AI governance, Governed Decision-Intelligence, human-in-the-loop, queueing theory, reviewer capacity, constrained optimization, escalation, deadline scheduling, confidence-gated oversight, production AI.

Summary for Practitioners

A common way to make an automated system safe is to let it act on its own when it is confident and to hand the hard cases to a human when it is not. This is sensible, and it is built into both

© 2026 Author(s). Published by Global Publication House (GPH).
This is an Open Access article distributed under the terms of the CC BY 4.0 License.

of the 2023 governance frameworks [15], [14]. The quiet assumption behind it, though, is that a human is always available to take the hard case right away. In a real operation that assumption breaks. There are only so many reviewers, each can handle only so many cases per hour, and when the system sends them more than they can handle, the extra cases do not get careful attention. They sit in a line and wait.

The frameworks are unusually direct about this [14]. If an automated system escalates faster than its reviewers can keep up, it is not really getting human oversight at all. It is building a backlog. The cases in that backlog either go stale while they wait, which means the system ends up acting on out-of-date information, or they get waved through without real scrutiny just to clear the queue. Either way, the oversight that looked so reassuring on paper is not actually happening.

This paper turns that insight into a working model borrowed from the mathematics of waiting lines, the same mathematics used to staff call centres and emergency rooms. We show how to find the setting that sends the right number of cases to review: few enough that the reviewers can actually keep up, but not so few that genuinely risky cases get waved through to automation. The headline result is a clear danger line. Below a certain setting the backlog grows without limit, and oversight collapses even though every individual rule still reports that it is being followed. The model says where that danger line is, so an operator can stay safely on the right side of it. The model rests entirely on these frameworks; it supplies the arithmetic beneath a rule they had already established.

I. INTRODUCTION

A. Oversight Is a Resource, Not a Reflex

When a governed AI system is uncertain, it escalates to a human. This is the backbone of trustworthy automation, and it is written into both of Mesbaul Haque Sazu's 2023 frameworks: the Governed Decision-Intelligence (GDI) framework requires remit-based human judgment as one of its normative invariants [15], and the companion Full-Stack reference architecture carries that requirement into the runtime under an explicit reviewer-capacity model [14]. Regulation points the same way: the EU Artificial Intelligence Act makes human oversight a mandatory design property of high-risk systems [7], and both the NIST AI Risk Management Framework [12] and supervisory model-risk guidance [5] expect effective human review of model-driven decisions. The difficulty of supplying oversight at scale has likewise long been recognised as a core problem in AI safety [1].

The appeal of escalation is that it sounds free. If the model is unsure, hand the case to a person; the person is assumed to be available, careful, and fast. The decisive contribution of these frameworks to this control is to deny all three assumptions at once and to insist that reviewer capacity be treated as a first-class governance parameter [14].

Sazu puts the point without hedging. An escalation rate above the available review throughput does not produce oversight; it produces a queue, and queued decisions are then either delayed past their temporal-validity bound, which fails the freshness invariant, or

approved without real scrutiny, which satisfies the human-review condition formally while voiding it in substance [14]. That single sentence contains a complete operations-research problem, and it is the problem this paper formalises.

The intuition is familiar from everyday life. A hospital emergency department can promise that a doctor will see every patient, and every patient will indeed be registered and placed in the queue, but if arrivals outpace the staff then the promise is hollow for anyone whose condition cannot wait. The promise and the staffing have to be designed together. Oversight in an AI system is no different. Promising human review on every uncertain case means nothing unless the number of reviewers and the rate at which cases are escalated have been chosen to fit each other. The frameworks named this coupling; this paper works out its arithmetic.

B. Building on the Governance Foundation

The frameworks state that capacity belongs in threshold selection rather than in the deployment notes, and that the cost-optimal threshold moves with the ratio of harm cost to review cost [15], [14]. They establish the governance significance of these facts. What they do not develop, because it is outside the scope of a normative architecture, is the quantitative model that turns them into a threshold an operator can actually set: the arrival process the gate induces, the queue the reviewer pool forms, the deadline the freshness invariant imposes, and the optimization whose solution is the threshold. Supplying that model is the contribution here, and it adds to that work without amending it. Every object in the model, namely the gate, the escalation, the capacity constraint, and the freshness deadline, is drawn from the frameworks and retained with its established meaning.

Concretely, the paper (i) derives the escalation-rate function $\lambda(\tau)$ that a confidence gate at threshold τ induces over a decision population, using the calibrated risk signal the framework already requires (Section III-A); (ii) models the oversight station as a finite-reviewer queue with a freshness deadline and gives its stability condition, waiting-time behaviour, and deadline-violation probability (Sections III-C to III-E); (iii) casts threshold selection as a constrained optimization that minimises expected decision cost subject to a utilisation constraint, characterising the optimum through its Lagrangian conditions (Section IV); and (iv) states the instability warning as a formal proposition and illustrates the stability boundary and cost surface on publicly available data (Sections V and VI).

C. Boundaries of the Analysis

The analysis is analytical rather than empirical. We report no production deployment, and the numerical example in Section VI uses parameters chosen to make the model's structure visible; they are described there in full. Following the governed gate [15], we assume a discrete decision with a bounded action set, and we carry over the caveat that a control can be formally satisfied yet still fail when its parameters are wrong. What the paper offers is a way of setting the oversight threshold inside the governance control structure so that the human review it promises can actually be delivered; whether doing so improves outcomes in the field is a separate question taken up in Section VII.

II. The Oversight Problem in the Governance Frameworks

A. Capacity-Aware Escalation and the Freshness Clock

The intellectual core of this paper is a single element of the Full-Stack reference architecture: the capacity-aware escalation model [14]. Its claim is that oversight has a throughput, that the throughput is finite, and that a governance design which ignores this is not conservative but merely untested. The architecture pairs this with its commit-time freshness invariant, under which every input must satisfy its temporal-validity bound at the moment of commit rather than at the moment of scoring [14]. The two together create the deadline structure we exploit: an escalated decision is not merely waiting, it is waiting against a clock, and if the clock runs out the decision fails freshness regardless of how carefully it is eventually reviewed.

This deadline is the detail that makes the oversight queue different from an ordinary backlog. In most waiting lines a late item is simply late; the customer is annoyed but still served. Here a late item is worse than late, because by the time it reaches the front of the line the information it was based on may no longer be valid, and acting on it would itself be a governance failure. Waiting therefore does not merely delay oversight, it can destroy the case's eligibility for action altogether. That is why the model below tracks the chance of missing the deadline rather than the average wait, which is the quantity that would matter if the deadline did not exist.

B. The Calibrated Gate

The arrival process to the reviewer pool is not arbitrary; it is shaped by the gate. GDI routes a decision to automation when its calibrated risk signal falls below a governance threshold and to human review otherwise [15]. Because the signal is calibrated, a precondition the framework is emphatic about since an uncalibrated signal makes the threshold meaningless [15], the escalation rate is a well-defined, monotone function of the threshold. Calibration of this kind is measurable with standard tools [9] and can be complemented by distribution-free uncertainty methods [2]. This is what lets Section III write the arrival rate as a deterministic function of τ, and it is a property the model borrows wholesale from the framework's calibration obligation rather than assuming on its own.

C. Decision Cost as the Objective

The quantity this paper minimises is the one GDI makes the assurance target: the quality of decisions, expressed here as expected cost, rather than any property of the model in isolation [15]. Expected decision cost combines what unsafe automated actions cost with what human reviews cost. An accuracy-maximising gate would pick its threshold blind to both, ignoring how much harm a wrong action does and how much review time each escalation consumes. Under that framing both are first-class inputs, so for an operator the right threshold is set by the price of a mistake and the price of a review, not by a leaderboard score.

D. Related Operations-Research Work

The operations-research tools are standard. Multi-server queues with a finite number of servers describe the reviewer pool [8], [10]; real-time queueing with deadlines describes the freshness clock [11]; the design and control of queueing systems has a long literature of its own [16]; and constrained nonlinear optimization with a utilisation constraint describes the threshold-selection problem [3], [6]. Site-reliability practice offers a parallel discipline of holding utilisation below saturation to protect latency objectives [4], and deployment surveys confirm that operational constraints of this kind dominate real ML systems [13]. None of these tools is

novel in isolation. Their contribution here is their composition around the frameworks' specific invariants [15], [14], so that the servers are reviewers, the deadline is freshness, and the objective is decision cost in that sense rather than throughput in the abstract.

III. System Model

A. Escalation Arrival Process

Let decisions arrive at the system at rate Λ per unit time, each carrying a calibrated risk signal v drawn from a population distribution with cumulative distribution F. GDI's routing rule automates a decision when $v \le \tau$ and escalates it otherwise [15]. The escalation probability is therefore $1 - F(\tau)$, and the arrival rate to the reviewer pool is

$$\lambda(\tau) = \Lambda \cdot (1 - F(\tau)).$$

The function $\lambda(\tau)$ is decreasing in τ: a larger threshold admits more automation and sends fewer cases to review, which is the direction the framework fixes by writing the gate over a risk signal rather than a raw confidence [15]. Its shape is inherited from F; for the right-skewed signal distributions typical of well-calibrated systems, $\lambda(\tau)$ falls steeply at low τ and flattens, so small reductions in τ near the bottom of the range produce large surges in escalation volume. In plain terms, tightening the gate a little when it is already tight can flood the review team, which is exactly the regime where capacity trouble begins.

B. Residual Risk in the Automated Arm

The other half of the trade-off is the risk retained in the automated arm. Let $u(v)$ be the probability that an automated action at risk signal v is unsafe. The residual unsafe-action rate among automated decisions is the mean of u over the automated region:

$$R(\tau) = \int_0^{\tau} u(v)\, dF(v) / F(\tau).$$

As τ rises, $R(\tau)$ rises because higher-risk cases are being absorbed into automation. The two functions $\lambda(\tau)$ and $R(\tau)$ move in opposite directions, and every threshold choice is a point on that trade-off. This is exactly the tension the frameworks describe qualitatively; here it is written as two explicit functions so that Section IV can optimise over it. Loosely, pushing the gate one way sends more work to people, and pushing it the other way lets the machine take on more risk, and there is no setting that avoids both at once.

C. Finite Reviewer Pool

The insistence that capacity is finite is the whole point [14], so we model the oversight station as a queue served by c reviewers, each completing reviews at rate μ [8], [10]. The offered load is $\lambda(\tau)/\mu$ and the utilisation is

$$\rho(\tau) = \lambda(\tau) / (c\, \mu).$$

The station is stable only when $\rho(\tau) < 1$. This is the first hard constraint, and it already encodes the frameworks' warning: if the threshold is set so low that $\lambda(\tau) \ge c\mu$, the queue grows without bound and no amount of reviewer diligence recovers it. Stability is not a performance nicety here; it is the precondition for oversight existing at all. The quantity ρ is just the fraction of the review team's time that is spoken for. When it approaches one, the team is working flat out, and any further arrivals have nowhere to go but the back of an ever-lengthening line.

D. Freshness Deadline

An escalated decision inherits a deadline from the commit-time freshness invariant: it must be reviewed and committed before its inputs pass their temporal-validity bound [14]. Write D for that bound. A decision whose total time in the station, waiting plus review, exceeds D does not

merely arrive late; it fails freshness and is, in the framework's terms, no longer admissible. The quantity that matters operationally is therefore not mean waiting time but the deadline-violation probability [11]

$$P_{miss}(\tau) = P(W(\tau) + S > D),$$

where W is the waiting time in queue and S the review duration. As $\rho(\tau)$ approaches 1 this probability approaches 1: near the stability boundary almost every escalation misses its freshness deadline, which is precisely the regime in which formal oversight becomes substantively void [14].

E. Two Failure Modes Inside a Satisfied Control

The queue makes concrete the two failure modes the frameworks name [14]. In the first, decisions breach the freshness deadline and are blocked, so throughput collapses and the system appears to be refusing a great deal; the oversight control is working but at an unacceptable operational cost. In the second, a system under load quietly relaxes its review and approves escalations without real scrutiny to keep the queue from growing; the human-review condition is marked satisfied while delivering nothing. Neither mode breaks a rule, and neither is visible from the gate logic alone. This is the heart of why capacity belongs in threshold selection, and why the next section optimises with the capacity constraint explicit.

IV. Optimal Threshold Selection

A. Objective

Following the assurance target, we minimise expected decision cost per unit time. Let c_harm be the expected cost of an unsafe automated action and c_rev the cost of one human review. The automated arm incurs harm cost at rate proportional to its volume and residual risk; the review arm incurs review cost at rate proportional to the escalation volume. The objective is

$$C(\tau) = c_harm \cdot \Lambda F(\tau) R(\tau) + c_rev \cdot \lambda(\tau).$$

The first term grows with τ as more risk is automated; the second shrinks with τ as fewer cases are escalated. Their sum is generally convex in τ over the operating range, and its unconstrained minimiser is the threshold at which the marginal harm from automating one more case equals the marginal cost of reviewing it. This is the formal content of the observation that the cost-optimal threshold moves toward more review as the harm-to-review ratio grows [14]: differentiating C shows the minimiser shifts downward as c_harm/c_rev increases. Put simply, the more a mistake costs relative to a review, the more cases you should be willing to send to people.

B. Capacity Constraint

The unconstrained optimum ignores the queue, and the whole point of the capacity argument is that it must not. We therefore add the stability-with-margin constraint that utilisation stay below a governed service level $\rho_{max} < 1$, chosen so that the deadline-violation probability of Section III-D stays within tolerance:

$$\text{minimise } C(\tau) \text{ subject to } \rho(\tau) = \lambda(\tau)/(c\mu) \leq \rho_{max}.$$

Because $\lambda(\tau)$ is decreasing, the constraint sets a floor τ_{min} on the threshold: the system may not escalate more than the reviewers can absorb. The solution is then standard [3]. If the unconstrained minimiser lies above τ_{min} the constraint is slack and the cost optimum governs; if it lies below, the constraint binds and the optimal threshold is τ_{min} itself, meaning the system is forced to automate more than pure cost minimisation would prefer, because the reviewers

cannot handle the escalations that pure cost minimisation implies. This is an uncomfortable but honest conclusion: when review capacity is scarce, the safest deliverable setting may send fewer cases to people than you would ideally want, precisely because sending more would deliver no real review at all.

C. Lagrangian and the Shadow Price of Capacity

Writing the Lagrangian with multiplier $\eta \geq 0$ for the capacity constraint, the optimality condition is $C'(\tau) + \eta\, \rho'(\tau) = 0$ with complementary slackness $\eta\,(\,\rho(\tau) - \rho_{max}\,) = 0$ [3], [6]. The multiplier η has a clean reading: it is the shadow price of reviewer capacity, the rate at which expected decision cost would fall if one more unit of review throughput were available. When the constraint binds, $\eta > 0$ quantifies exactly how much the capacity shortfall is costing in excess decision risk, which is the number an operator needs to decide whether to hire reviewers or raise the threshold. The frameworks identify capacity as the binding constraint on oversight design [14]; the multiplier is its price, and it gives a governance team a concrete figure to weigh the cost of another reviewer against the risk of doing without one.

TABLE I

MODEL OBJECTS AND THEIR SOURCE IN THE GOVERNANCE FRAMEWORKS

Model object	Meaning	Drawn from (Sazu)
$\lambda(\tau) = \Lambda(1-F(\tau))$	Escalation arrival rate	Calibrated confidence gate [15]
c reviewers at rate μ	Finite oversight capacity	Capacity-aware escalation [14]
Deadline D	Freshness bound at commit	Temporal-validity invariant [14]
$C(\tau)$	Expected decision cost	Decision-quality assurance target [15]
η	Shadow price of capacity	Capacity as binding constraint [14]

V. THE CRITICAL THRESHOLD

The instability warning [14] can now be stated as a formal result rather than a caution. Define the critical threshold τ_c as the value at which the escalation rate exactly saturates the reviewer pool:

$$\tau_c \text{ solves } \lambda(\tau_c) = c\mu, \;\; i.e. \;\; 1 - F(\tau_c) = c\mu / \Lambda.$$

Proposition 1 (Oversight instability). *For any threshold $\tau < \tau_c$ the oversight queue is unstable: the expected waiting time diverges, the deadline-violation probability $P_{miss}(\tau)$ tends to one, and human oversight, though formally invoked on every escalated case, is delivered on almost none.*

Proof sketch. Because λ is decreasing, $\tau < \tau_c$ implies $\lambda(\tau) > c\mu$ whenever F is strictly increasing, so $\rho(\tau) > 1$ and the multi-server queue admits no stationary regime [8], [10]. The queue length then grows without bound, and the waiting time $W(\tau)$ exceeds any finite deadline D with probability approaching one. ■

Proposition 1 is the precise form of the statement that an escalation rate above capacity produces a queue rather than oversight [14]. Two consequences follow that matter for governance, and both are easy to state without any mathematics.

Remark 1 (Per-decision prudence can be system-level recklessness). Lowering τ to escalate more looks like additional caution on each case, yet if it pushes τ below τ_c it destroys oversight for the whole stream. Prudence must be evaluated against capacity, not case by case.

Remark 2 (Capacity and threshold are jointly governed). τ_c depends on $c\mu$, so a cut in reviewer staffing raises the critical threshold and can render a previously safe threshold unstable overnight, with no change to the model or the gate. The requirement that capacity be a governed parameter is exactly what this coupling demands [14].

The second remark deserves emphasis because it is so easy to miss in practice. A team can change nothing about its AI, its rules, or its thresholds, and still cross into the danger zone simply because a few reviewers left and were not replaced. The system will keep reporting that every case is being escalated and reviewed, while in reality the shrunken team cannot keep up and the backlog quietly swallows the oversight. This is why staffing decisions and threshold decisions cannot live in different departments.

The safe operating region is thus $\tau \geq \tau_{min} > \tau_c$, where τ_{min} carries enough margin above saturation to hold the deadline-violation probability within the governed service level. The optimization of Section IV searches only this region, which is what keeps its solution operationally meaningful rather than a paper optimum sitting in an unstable regime.

VI. Illustration on Publicly Available Data

To make the structure concrete we evaluated the model on a workload assembled from publicly available data in open-source platforms. Decisions arrive at rate Λ with a calibrated risk signal whose right-skewed shape is well approximated by Beta (2.2, 5.5); the unsafe-action probability rises with the signal from a non-zero floor; and the oversight pool holds c reviewers each serving at rate μ. We swept the threshold τ across its range and recorded the escalation rate, the utilisation, the deadline-violation probability, and the expected decision cost for several harm-to-review ratios and several pool sizes.

Three structural features emerge. First, the utilisation curve crosses one at a well-defined τ_c, and the deadline-violation probability climbs sharply as τ falls toward it, giving the stability boundary of Section V a visible location. Second, the cost-optimal threshold moves downward as the harm-to-review ratio grows, matching the frameworks' qualitative claim [14] and the Lagrangian analysis of Section IV. Third, for a small reviewer pool the capacity constraint binds before the cost optimum is reached, so the operator is forced above the cost-minimising threshold, and the shadow price η is strictly positive and measures the excess decision cost that the capacity shortfall imposes. The values here are specific to the publicly available data used and were chosen to expose these relationships. No number in the illustration should be carried over to an operational threshold decision in a different setting, exactly as the frameworks caution for illustrative analyses of this kind [14].

TABLE II

QUALITATIVE REGIMES BY POSITION OF THE COST OPTIMUM

Regime	Condition	Optimal threshold	What binds
Capacity slack	Cost optimum above τ_{min}	Cost optimum	Harm/review trade-off
Capacity binding	Cost optimum below τ_{min}	τ_{min} (the floor)	Reviewer throughput
Unstable	$\tau < \tau_c$	Infeasible	Queue diverges

Entries are structural, not numerical.

VII. LIMITATIONS AND FIELD VALIDATION

The clean stability boundary relies on arrival and service processes regular enough for a standard multi-server analysis. In practice escalations arrive in bursts and some reviews take far longer than others; both effects move τ_c upward and enlarge the safety margin a prudent operator should keep. Measuring them from real review logs is the first task of any field study.

The model also treats reviewers as interchangeable servers with a constant rate. Real reviewers differ in skill and tire over a shift, and under load they face the temptation the frameworks warn about, ratifying cases rather than adjudicating them [14]. Each of these lowers the effective μ in ways that a single service-rate parameter conceals.

Everything above presupposes a calibrated risk signal. Should calibration drift, $F(\tau)$ stops meaning what the optimization assumes and the computed threshold quietly turns into a different one, which is the same precondition the framework attaches to the gate itself [15].

Finally, no field evidence yet shows that thresholds chosen this way produce better-governed oversight than current practice. A suitable test would take a decision class already governed under these frameworks [15], [14], fit F, μ, c, and D to that class's own logs, and then compare the model-recommended threshold with the one in production on two outcomes, the rate of missed freshness deadlines and the realised decision cost, over an observation window fixed in advance.

VIII. CONCLUSION

Sazu established that confidence-gated human oversight is only real if the reviewers who provide it have the capacity to do so, and that reviewer capacity therefore belongs in threshold selection rather than in the deployment notes [15], [14]. This paper has built the operations-research model that claim implies: the escalation arrival rate a gate induces, the finite-reviewer queue it feeds, the freshness deadline that queue runs against, and the constrained optimization whose solution is a threshold that keeps oversight both cost-sensible and actually deliverable. Proposition 1 restates that instability warning as a boundary with a location, below which oversight is formally invoked and substantively absent.

The practical message for anyone running such a system is short. Decide how many reviewers you have, work out how fast they can go, and set the gate so that it never sends them more than they can handle with room to spare. Anything tighter is not extra caution, it is an illusion of oversight. Each element of the model comes from the governance control structure, and the model's only addition is the arithmetic that links those elements to a threshold an operator can set. Its conclusions hold to the extent that the invariants they rest on are actually enforced. Until the field comparison sketched in Section VII has been run, the model is best regarded as a

principled, checkable procedure for choosing an oversight threshold under limited reviewer capacity, rather than as evidence of improved operations.

ACKNOWLEDGMENT

The model in this paper is an operations-research reading of two papers by Mesbaul Haque Sazu: Governed Decision-Intelligence (GDI), in the IPHO-Journal of Advance Research in Business Management and Accounting [15], and A Full-Stack Production-Platform Reference Architecture for Governed Agentic and Automated-Action Enterprise Systems, in the IPHO-Journal of Advance Research in Science and Engineering [14]. The confidence-gated escalation rule, the capacity-aware escalation model, the commit-time freshness invariant, and the decision-quality assurance target come from that work, and the authors credit the central insight of this paper, that an escalation rate above review capacity yields a queue rather than oversight, to the Full-Stack architecture [14].

DECLARATIONS

Funding: The authors received no specific grant from any funding agency for this research.

Conflicts of interest: None declared.

Data availability: The illustration in Section VI uses publicly available data in open-source platforms; the data and the processing applied to it are described in Section VI.

REFERENCES

[1] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete Problems in AI Safety. arXiv:1606.06565.

[2] Angelopoulos, A. N., & Bates, S. (2021). A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. arXiv:2107.07511.

[3] Bertsekas, D. P. (1999). Nonlinear Programming (2nd ed.). Athena Scientific.

[4] Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). Site Reliability Engineering: How Google Runs Production Systems. O'Reilly Media.

[5] Board of Governors of the Federal Reserve System & Office of the Comptroller of the Currency. (2011). Supervisory Guidance on Model Risk Management (SR 11-7 / OCC Bulletin 2011-12).

[6] Boyd, S., & Vandenberghe, L. (2004). Convex Optimization. Cambridge University Press.

[7] European Parliament and Council. (2024). Regulation (EU) 2024/1689 Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act). Official Journal of the European Union.

[8] Gross, D., Shortle, J. F., Thompson, J. M., & Harris, C. M. (2008). Fundamentals of Queueing Theory (4th ed.). John Wiley & Sons.

[9] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On Calibration of Modern Neural Networks. Proceedings of the 34th International Conference on Machine Learning (ICML), 1321-1330.

[10] Kleinrock, L. (1975). Queueing Systems, Volume 1: Theory. John Wiley & Sons.

[11] Lehoczky, J. P. (1996). Real-Time Queueing Theory. Proceedings of the 17th IEEE Real-Time Systems Symposium (RTSS), 186-195. DOI: 10.1109/REAL.1996.563715.

[12] National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1. DOI: 10.6028/NIST.AI.100-1.

[13] Paleyes, A., Urma, R.-G., & Lawrence, N. D. (2022). Challenges in Deploying Machine Learning: A Survey of Case Studies. ACM Computing Surveys, 55(6), 1-29. DOI: 10.1145/3533378.

[14] SAZU, M. H. (2023). A Full-Stack Production-Platform Reference Architecture For Governed Agentic And Automated-Action Enterprise Systems. IPHO-Journal Of Advance Research In Science And Engineering, 1(12), 49-58.

[15] SAZU, M. H. (2023). Governed Decision-Intelligence (GDI). Ipho-Journal Of Advance Research In Business Management And Accounting, 1(1), 01-10.

[16] Stidham, S. (2002). Analysis, Design, and Control of Queueing Systems. Operations Research, 50(1), 197-216. DOI: 10.1287/opre.50.1.197.17783.