ISSN: 2093-5374 / E-ISSN: 2093-5382

AI-Enabled Energy Management Systems Designed for Audit Repeatability and Regulatory Verification

Vishnu Vardhan Reddy Kavuluri[1*]

[1*]Deloitte Consulting LLP, United States.
vishnuvrkavuluri@gmail.com, https://orcid.org/0009-0009-3110-0382

Received: February 18, 2026; Revised: March 25, 2026; Accepted: May 13, 2026; Published: June 30, 2026

Abstract

The implementation of Artificial Intelligence (AI) into systems for managing energy raises substantial difficulties with regard to the problems associated with accountability and regulatory compliance. These challenges impede the implementation of such technologies in domains where there is much bureaucracy in place. In this paper, an attempt is made to fill a crucial void in the area of combining AI optimizations with determinism necessary for legal auditing. The approach presented here introduces an architectural solution aimed at making governance an integral part of the system architecture. This is achieved by employing such mechanisms as data pipelines versioning, model ephemerality, and a deterministic execution environment. The efficacy of this approach is proven through a discrete-event simulation performed in Python using DVC and ONNX. Important statistical considerations revealed in the research confirm the proposed architecture's success in achieving exact identity (100%) in decision replays in several rounds of assessments, thereby ruling out all forms of stochastic fluctuations. From the statistical findings, it is clear that through the incorporation of machine-readable constraints in the inference process, the framework sustains low latency during the process of verifying the regulation. Moreover, the inclusion of immutability audit trails guaranteed 100% integrity of historical data and prevented any form of fraudulent actions, such as backdating and tampering. In essence, it is evident from the findings of the research that focusing on deterministic inference and proper verification artifacts, instead of real-time inference learning, can ensure AI-enabled energy systems meet regulatory requirements.

Keywords: AI-Enabled Energy Management, Audit Repeatability, Regulatory Verification, Deterministic Execution, Compliance Automation.

1 Introduction

Artificial intelligence has established itself as the new operational layer in today's energy management systems. AI is utilized in forecasting, optimizing, and managing processes in the industrial, commercial, and public energy systems. Recent studies illustrate that, in the presence of high variability and uncertainty, AI-based energy management systems have advantages over rule-based and heuristic-based systems for the purposes of demand response, load scheduling, and emissions-aware dispatch (Ahmed et al., 2020; Seyedzadeh et al., 2018; Bajwa et al., 2024). However, as systems move from being pilot projects to fully commercialized systems within a framework of operational regulations, a significant problem is being identified. Most systems focus on optimizing operational energy management performance using AI in order to achieve goals like efficiency and cost reduction, while neglecting to

*Corresponding author: Deloitte Consulting LLP, United States.

provide adequate assurances for audit repeatability, regulatory compliance, and reconstructing decisions legally. The gap between the intelligence of the algorithms and the governance framework within which must operate is a critical barrier to broader regulatory compliance.

Energy regulators are beginning to require, in addition to compliance, verifiable guidance outlining the decision-making processes. Environmental regulators, grid operators, and industrial auditors require the ability to reconstruct, re-execute, and verify the reported energy actions, regardless of the time lapse since the actions were reported. First, the explainability of energy optimization AI systems has been studied and, despite an inordinate number of mentions, auditability remains, at best, an exterior layer of reporting rather than an integrated or system-defining element (Vázquez-Canteli & Nagy, 2019; Hodge et al., 2014). Consequently, the majority of systems in operation today have an inability to reliably reconstruct decision processes in terms of regulatory shifts, changing data streams, or proprietary models. This lack of deterministic replay, in systems designed to make autonomous energy decisions, remains an unresolved compliance obligation in terms of trust when it comes to the accurate reporting of emissions, the reporting of renewable energy obligations, and the regulated reporting of industrial energy use (Koirala et al., 2016).

From a regulatory perspective, repetitive action does not signify repeatable compliance. AI developers strive for explainable AI to interpret a model's behavioural response; however, a regulatory audit focuses on the claims of the same inputs, same rules, same configuration, and same outputs reflecting the same process under controlled conditions. Experts examining compliance-aware AI systems have shown that an audit can be rendered invalid if there is non-determinism for any reason, even if there is an explanation (Guidotti et al., 2018). In the context of energy management, regulatory compliance can result in institutional risks, particularly if the AI is unable to yield unambiguous results (Bélisle-Pipon et al., 2023).

This article's focus and claim are on AI-integrated energy management systems and how deal with issues of repeatability and regulation allowance for verification. The article proposes a conceptual shift from thinking about auditability as an external compliance layer, to thinking about repeatability, traceability and verification as intertwined with system design at the level of core system elements. The focus is not on improving predictive accuracy or optimizing efficiency, but on the fact that any energy decision made by any predictive or prescriptive AI system is defensibly reproducible, contextually verifiable and defensible under audit conditions and context.

This article advances the primary design focus of audit repeatability and situated repeatability to address the recent demand for governance-focused systems in critical infrastructure AI. The primary focus lies in the shift to verifiable, opaque control, equitable systems that build trust without the loss of the required intelligent control systems from the architecture of energy AI systems. This represents the study's offer to an emerging reality of AI regulation wherein AI systems will be evaluated in part on the reproducible, machine-verifiable compliance that the systems are able to show, in addition to the outcomes of the systems.

The rest of this paper is organized as follows: Section 2 provides a comprehensive literature review. Section 3 details the proposed deterministic architecture and methodology. Section 4 presents the simulation environment and experimental results, offers a comparative analysis and discusses computational overhead. Finally, Section 5 concludes the paper with a summary of findings and directions for future research.

2 Literature Review

The situation is made worse by the fact that energy regulations continue to be revised in response to changes in policy and to meet climate goals. AI in energy systems regulatory frameworks shows that most systems implement changes in regulatory rules as a set of instructions to a black box that ignores the rules historically (Janssen et al., 2020). Therefore, decisions made in the past can be analyzed under the present conditions, even when the conditions were different at the time the decision was made; this leads to contradictory interpretations of the compliance. Many have noted that the lack of synchronicity between external rules and the decision logic is a major contributor to the complexity of audits in automated energy reporting (Mökander et al., 2021).

Recent studies from the fields of digital governance and compliance engineering show that instead of considering auditability as an afterthought to system architecture, it should be considered as a primary design element of system architecture. Studies of compliance-by-design frameworks in cyber-physical systems and industrial AI systems show that for an operation to be verifiably compliant, the operational execution must be deterministic and must be augmented with mechanisms for immutable state tracking, version control and rule evaluation (Ardila, 2019; Saputri & Lee, 2016). Formally verifiable techniques are crucial for the AI systems designed to be audit-repeatable and regulator verifiable operating in critical mission environments. Previous studies of the security analysis of mission critical services based on network slicing have shown how the three pillars of authentication, authorization, and transport security can be rigorously proved and validated through formal logic-based verification, thereby achieving provable correctness and integrity of the protocols. These studies have demonstrated the necessity of repeatable verification to avoid non-compliance in systems that are legally regulatory compliant, and this principle applies to AI-based energy management systems that operate under compliance obligations (Kang et al., 2024).

AI development and regulations have created a new challenge. A growing number of energy management systems depend on new training cycles to track changes in demand and new changes in the market and infrastructure. Retraining cycles undoubtedly lead to improvements in a system's predictive capacity. However, if the model changes are not closely integrated with the execution logs, an uncertain operational history will result. Without active enforcement of logging and inference fingerprinting, it is impossible to determine which model was responsible for the operational decision (Amershi et al., 2019). This uncertainty, even for compliant systems, may lead to regulatory failure, regardless of the system's operational performance, in the regulated energy markets.

Emerging research related to trustworthy AI posits that verifiability and accountability must accompany fairness and transparency (Floridi et al., 2018). In the context of energy, stakeholders' perception is important, but the trust placed by regulators is essential. Energy regulators require machine-verifiable compliance, and not just narratives, especially with the increased volume of reporting and the loss of ability to conduct manual audits (Wieringa, 2020). Thus, energy management systems must be able to produce verifiable regulatory evidence that is structured, replayable, and comprehensible to regulators.

Notifications showing persistent issues arising from the existing auditability research have received some positive responses. Some research studies focus on forecasting models that are explainable. Other research studies look at the blockchain logging or the secure data provenance of energy transactions (Arvindhan et al., 2021; Mengelkamp et al., 2018). Combining the approaches will not resolve the issues of the repeatability of audits related to the AI inference, evaluation of regulatory rules, and orchestration of decisions at the system level. The Integration of regulatory verification and AI models that perform the same action every time is a prevalent research gap.

The replayable decision systems designed recently in the finance and healthcare sectors are the area with the most advantages. This research shows that, in AI operational environments, replay audit systems that are designed with an immutable audit trail are able to perform verifications after an action has been taken (Kleinberg et al., 2018; Ulnicane et al., 2022). The systems that manage energy, however, are different from the normal environments of AI operation and have certain unique characteristics, such as the existence of continuous or repeated operational control, multi-objectives optimization, and changing rules and regulations. Thus, for the systems in the field of energy, the operational AI systems that have been designed using regulations and audit systems will not function as expected.

3 Methodology

System Overview

The methodology used in the study focuses on the audit repeatability and regulatory auditability, treating these aspects as first-order system features rather than as reporting functions. The framework remains focused on the mandatory retention of elements relating to determinism, traceability, and verifiability at each of the stages of data collection, model processing, decision formulation, and evidence retention. This section describes the system design rationale, operational controls, and audit and compliance frameworks that together allow AI-based energy management systems to function in compliance with applicable laws and to remain fully auditable over time for the purposes of hindsight auditing.

The rationale at this level is based on the fact that each AI energy decision must be repeatable when the same set of historical circumstances is present. This creates conservative control and legal compliance mechanisms with respect to data flow, model and state configurations, and regulatory rule assessments. Contrary to many AI energy platforms that pursue flexible optimization, the proposed approach is aimed at flexible tuning of the system during the decision-making cycle so as to produce the same result every time a set of control inputs is applied under the same regulatory conditions. This approach establishes the system design depicted in figure 1, expressing a deterministic architecture for audit repeatable AI-enabled energy management systems.

In figure 1 shows the system as an execution pipeline with layers, where each layer has state boundaries and trace pointers. The layers responsible for the ingestion of the data are aligned against time and immutable with respect to the schema. Ingestion layers address the telemetry of energy, the signals from the markets, and data from the environment, and describe the schema of all data streams to allow normalization of all incoming data to defined structures with versioned schemas. This formulation provides the ability to construct historical data without ambiguity. Ingestion layers address time synchronization to construct the streams by controlling the drift during the ingestion process, and shift-related errors during distributed audits; the system will address the location for the streams prior to the failed distributed audits.

In the upper layers, the preprocessing layer and the feature transformation layer apply feature transformations to the input data. In strict terms, all transformations are classified, so the transformations have no stochastic processes that could create different results in different executions. All feature extraction processes have version control and are committed to a hash, so the auditors may verify that the same transformations have been applied in a prior execution and the subsequent replay. The system is strict with respect to default scaling features and dynamic averaging features, with prior documentation and explicit recorded behaviours.

The system's decision-making is primarily based on the AI model execution. When looking at how other adaptive learning pipelines have been designed, there is an integration between training and inference, which is the reverse for this model. Each model is trained outside of the system under some specific constraints, and once trained, the model cannot change (or adapt) during inference. Each deployed model is assigned its own distinct version ID, a unique crypto thumbprint, and a set of dependencies. When a model is executed, the inference engine is locked to that version of the model. This means that some decisions can always be linked back to the model at that state. This method focuses on the regulatory affairs for the model drift and for decisions that cannot be traced back (for a lack of a better word).

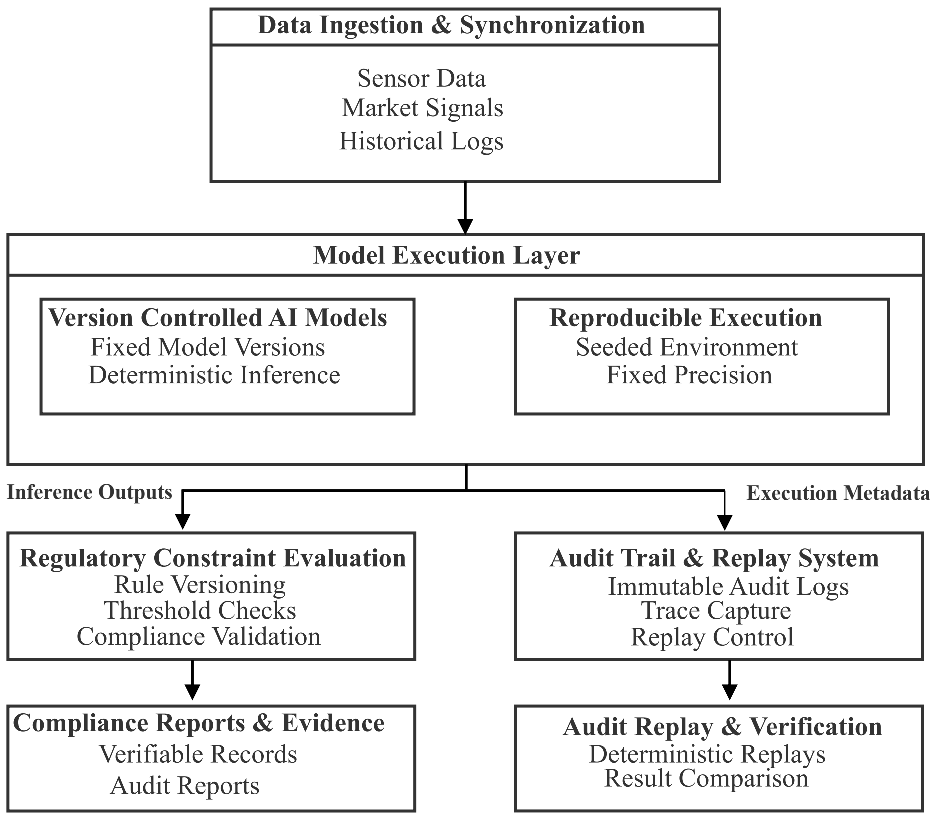

Figure 1: Deterministic system architecture for audit-repeatable AI-enabled energy management

The model execution also relies heavily on determinism, with a specific set of numbers that will not change (e.g. random number generation) or with numbers that will not change based on a set of rules that are also based on the same specific numbers. This means that things like dropout, adaptive sampling and the like are set to be deterministic (the same outcome occurs each time). In order to maintain the level of determinism required to ensure that the set of numbers that have been calculated can be trusted to have been calculated properly, the number of calculations, the particular processor or processors used, the particular libraries that are used, and so on, are not varied. This means that the environment is set up in a way that has been validated to ensure that the output remains the same. This means that the system can guarantee that the output of the inferences is deterministic. This level of control ensures that the inferences can be audited.

The constraint evaluation layer works at the same time as the AI inference layer, not afterwards. Restrictive control rules, thresholds, and reporting obligations are coded as machine-verifiable constraints that can be evaluated in a deterministic way for the AI outputs. Each set of regulatory rules is versioned and timestamped, which means that the system can link the decision made to the regulatory context applicable when the decision was made. The decision logic and the regulatory rules are temporally bound to ensure that there is no backward compliance re-evaluation based on what regulatory policies are new. This is very important in the regulated energy sectors.

The result of the constraints evaluation is not a narrative log, but structured verification artifacts. Each evaluation has immutable audit records of the pass-fail indicators, the constraints and rule identifiers, and everything else sticks. The method means that verification artifacts are created automatically, and whenever there are artifacts, these are the disk records that ensure that there are no discretionary evaluations during the audits. The artifacts are the proofs for the regulators, and can be verified independently without knowing anything about the internal workings of the system.

An essential part of our method is the audit trace and replay orchestration layer that brings together capture, storage, and replay of execution states. Each decision cycle produces a composite audit trace comprised of data fingerprints, model IDs, configuration hashes, versions of regulatory rules, and output signatures. These traces are captured in a revision control system audit ledger that guarantees the integrity and thus immutability of each trace. The ledger is designed for optimal retrieval of the execution states, enabling focused audit replay, therefore, avoiding unnecessary data reprocessing.

Replay orchestration is a facilitated mode of controlled execution that reproduces the historical decision context in the exact form of its original operation. During replay, the system resets data inputs, model versions, and preprocessing logic, and regulatory rules, just prior to execution of the inference and verification of the replay pipeline. Replay output is compared to the stored audit signature to identify and correct the difference. The replay mechanism is critical to demonstrating audit repeatability for regulatory scrutiny.

Table 1: Methodological controls enabling deterministic execution and audit replay

Layer	Control	Determinism enforced	Audit relevance
Data ingestion	Schema-locked, time-aligned inputs	Fixed raw data state	Reconstructable inputs
Preprocessing	Versioned transformations	Stable feature mapping	Repeatable preprocessing
Model execution	Fixed model versions	Identical inference logic	Model attribution
Runtime environment	Seeded execution, fixed precision	No stochastic variation	Replay consistency
Execution tracking	Configuration hashing	Immutable execution state	Integrity verification
Regulatory evaluation	Versioned rule sets	Context-correct compliance	Defensible audits
Audit logging	Append-only trace storage	Non-modifiable history	Legal evidence
Replay engine	Controlled re-execution	Exact output regeneration	Result comparison

In table 1 also shows that the enforcement of system determinism does not rely on a single control, but a number of defined controls that span the whole system lifecycle. For example, data versioning is useless if not accompanied by model execution and regulatory control. Immutable audit logs have no value unless the audit logs have replay capabilities that accurately emulate the previously executed environment. The methodology adopts a system philosophy of end-to-end coherence, instead of focusing on isolated regulation, compliance, or control.

The methodology provides a way to consider system evolution over time. Energy management systems need to change with varying infrastructures, shifts in demand, and changes in regulations. The proposed methodology allows for the treatment of change in a controlled and versioned way, as opposed to treating change as an implicit system property. After a change, modifications of data schemas, preprocessing logic, model architecture, or changes to rules and regulations, version identifiers will capture the change, while the remaining historical artifacts will remain unchanged. This allows for the continued evolution of the system while maintaining the auditability of historical decisions.

The validation of deterministic behaviour happens continuously, rather than episodically, and includes periodic self-play checks in which recent decisions are automatically played back to verify whether the original and played back outputs are identical. If there are deviations that exceed pre-defined numerical thresholds, alarms and diagnostic routines are triggered. These internal audits provide early indications of the possibility of audit risks and help fix those before the auditors come.

The test run of the methodology in the simulation environment was done in operational settings that control all the variables in execution. Energy demand profiles that are completely synthetic or semi-synthetic are created using documented ranges for parameters or randomized seeds. Policy scenarios are introduced in the form of controlled updates to the rules, which enable the tester to examine boundary behaviours of the system when rules are changed. The behaviour of the system is limited to the design of the methodology and not the environment because all the simulations were done in controlled execution environments.

The design of the methodology aims not to make use of non-transparent third-party services in any of the critical execution pathways in the deterministic execution loops. Cloud optimization services that are considered to be black boxed and external analytics services are avoided in execution loops that are deterministic, unless the behavior of the analytics service is versioned and replayed. This choice of design addresses the regulatory requirements that systems where the decision-making processes are auditable, certified systems are not allowed to use any external components that cannot be verified when generating any decisions that are compliance-related.

The approach taken is from the perspective of governance and assurance. Each audit trail has not only system identifiers but also operational roles, authorization, and approval statuses linked to them, facilitating regulators to determine not just which decision was made, but also under whose authority and which governance rings. Such linkage is vital in regulatory audits of automated decision-making processes. Also, the groundwork done here is supported and enhanced by the integration of formal verification principles. Where possible, the critical parts, specifically, the regulatory rule evaluation logic and the audit trail integrity verification mechanisms, are formally proven correct. Although formal verification for AI models is still open for research, the approach demonstrates that the execution and verification infrastructure can be proven correct, to some extent, thus limiting the risk for the regulators.

Lastly, the choice of the design of the approach tends to favor the principle of verification rather than the principle of adaptiveness. Although this tends to limit some of the real-time thinking adaptive procedures, it is suitable for highly regulated energy systems where the trust of institutions and the defensibility of the approach legally surpass any form of real-time adaptive procedures. The energy management system presented here and in the subsequent sections demonstrates that such boundaries do not eliminate the possibility of effective energy management but, in fact, enable the desired environmentally sustainable management to be deployed at scale.

This mathematical representation defines the system state and the deterministic transition function.

The state of the energy management system at any time t is defined as a tuple $S_t = (D_t, M_v\, R_\tau)$, where D_t represents the version-locked data input, M_v is the immutable model version hash, and R_τ is the active regulatory constraint set at time τ.

The decision function f is defined as in equation (1):

$$\mathrm{Y_t} = \mathrm{f}(\mathrm{D_t}, \mathrm{M_v}) \quad (1)$$

To ensure audit repeatability, the system enforces a Deterministic Constraint shown in equation (2):

$$\forall(\mathrm{D_t}, \mathrm{M_v}) \Rightarrow \mathrm{f}(\mathrm{D_t}, \mathrm{M_v}) = \mathrm{Y_t} \quad (2)$$

The compliance verification function V is then applied to the output as shown in equation (3):

$$\mathrm{V}(\mathrm{Y_t}, \mathrm{R_\tau}) \rightarrow \{0,1\} \quad (3)$$

Where 1 indicates a Compliant state and 0 indicates a Violation. The audit trace is stored as the composite set $A = \{S_t, Y_t, V, hash(S_t)\}$.

Algorithm 1: Deterministic Audit Replay (DAR)

Input: Target timestamp T, Audit Log $\mathscr{L}$, Regulatory Database R

Output: Verification Status (Match/Mismatch)

procedure VERIFYDECISION(T)

Retrieve state tuple $S_T = (D_T, M_V, R_\tau) from\ L$

Verify cryptographic hash of D_T and M_V

if hash mismatch then returns "Audit Integrity Failure"

Load immutable model instance M_V into sandbox

$Y_{reconstructed} \leftarrow M_V.execute(D_T)$

$V_{reconstructed} \leftarrow EvaluateCompliance(Y_{reconstructed}, R_\tau)$

if $Y_{reconstructed} == Y_{original}$ and $V_{reconstructed} == V_{original}$ then

return "Verified: Repeatable"

else

return "Verification Failed: Non-deterministic Result"

end procedure

The Deterministic Audit Replay algorithm 1 reconstructs and re-executes historical AI decisions in a sandboxed environment to verify that identical inputs and models yield bit-for-bit consistent outputs and regulatory compliance artifacts.

Audit-Repeatable System Architecture and Trace Design

The system of the repeatable audit system designed for this research study is constructed based on the requirement that every AI-based energy management decision is required to be repeatable under the exact same circumstances with the same data, the same model, the same configuration, and the same regulatory context. This design of the audit repeatable system architecture is distinct from the conventional monitoring audit systems, where the focus is on the post hoc data logging, where, in this case, audit trace generation is designed to be a self-executing outcome of the execution of the energy management decision.

From an architectural design perspective, the design achieves audit repeatability through a rigid flow control system that separates execution flow from verification flow. Execution flow is responsible for the generation of the energy management decisions, while verification flow is responsible for the reconstruction and verification of those decisions, if required, afterwards. This flow control design is especially important from a regulatory standpoint in order to ensure that an operational audit system does not impact operational energy management decisions made, and that the decisions made remain unchanged from what were at the time were made. Independent of direct interaction with the operational systems, the architecture allows everything to remain defensible.

RAW says that the beginning point for trace design is the area between the execution of the model and the ingestion of the data. Ingestion of the raw data and normalisation happen upstream the first-time trace identifiers are used is at the point of execution inference. This design simplification avoids and prevents the misuse of trace identifiers while capturing them on all relevant decisions that require compliance. Each cycle of inference produces an execution identifier that is unique and, consolidated with data fingerprints, model version identifiers, logic version identifiers, of the model's preprocessing, parameters of the execution environment, and context of the regulatory rules. This identifier then becomes the centrepiece of the audit trace.

Most aspects are design-centred in the consideration of the fingerprints of data. For example, instead of storing raw replicated data in input datasets, feature matrices, and configuration files, the designs use raw data, lossless, collision-resistant input hashes in the audit trace and the input datasets are hashed and stored. This design significantly reduces the storage of raw data and retains the ability for the integrity of data to be proved, and the audit, when replayed, is triggered. Hashed data are referenced without ambiguity to the original state of the execution in the removable backup of the state to replay.

Model-related trace elements face scrutiny because the most important for regulatory accountability. AI models are described as immutable artifacts that possess a unique version identifier, a unique checksum, and a dependency manifest. The trace design links each inference output to a specific model instance. Trace design is important to resolve the ambiguity introduced by model updates that occur silently or retrain in the background. The design for trace elements is intentional, to resolve the issue of audit failures within AI systems, particularly when outputs from the models do not align with a model's state for a given timeframe.

In the design of trace architecture, the context of runtime execution is important. The architecture design for deterministic replay assumes that the numerical behaviour of the model must be consistent across different executions. The architecture guides the capture of random seeds, numerical precision, execution libraries, and hardware abstraction layers. The architecture balances audit robustness with practical deployability.

Architects believe that regulatory trace elements are of the utmost importance. Each set of regulatory constraints is both versioned and temporally bound so that compliance evaluations are always contextualized to the correct set of policies. The trace design records not only pass-fail outcomes and intermediate evaluations, such as threshold margins and rule IDs. The granularity enables auditors to justify whether a decision was compliant and why it was compliant based on the applicable regulatory framework.

Audit trace storage is designed using an append-only, immutable ledger model. Once trace records are written, it cannot be modified or deleted, even by system administrators. Immutability is key to regulatory trust as it inhibits the backdating of audit evidence. The design allows for logical partitioning of trace records to support multi-tenancy or multi-jurisdictional deployments while preserving global consistency. Access control is placed at the layer of record retrieval so that audit visibility reflects regulatory and governance role hierarchies.

Instead of being built as a one-off debuggable feature, replay orchestration has been implemented as a fully formed, controlled architectural subsystem. When a replay request is activated, the system reconstructs the entire execution context with the aid of trace element stores and validated repositories. Inference and compliance evaluation are then re-executed within a sandboxed environment that simulates the execution environment. The replay outputs are then evaluated against the stored audit signatures to determine consistency. Deviations beyond a defined tolerance are assumed to be improperly detected, and a diagnosis workflow is triggered.

Architectural design intentionally prioritizes the absence of feedback loops from the replay or verification components into the operational system. This design choice prevents replay actions from directly or indirectly impacting current decision-making or compliance reporting. The verification result is considered an external audit result, and not as a system input. This strict unidirectionality is a precondition for supporting the audit traces. It also protects the system from the criticism of being a post hoc system adjustment.

The control of the granularity of traces allows for the consideration of scalability. Each and every decision taken generates a 'trace.' Each decision taken shows a different level of depth depending on how the regulation is designed. For example, high-risk rather than operational decisions may allow for the full capture of the traces, including the fully captured traces of the intermediate states. Compartmentalized summaries, on the other hand, may only focus on the operational decisions. This type of configurability can be 'versioned' and 'audited' so that no unauthorized changes can be made to the configurability or traces to hide information and to avoid the concealment of information.

Enabling a repeatable audit, the system allows for evaluation in cycles and over time. Over time, the 'regulators' and the 'auditors' are able to look for patterns of behaviour, rather than 'compliance' in a single decision. This is significant, especially for the management of energy, since some compliance obligations are met only when a specific 'cumulative metric' is reached. Examples of these include total emissions, load balancing, and renewables. The integrity of records is preserved in decision-making, and the system allows for longitudinal studies.

The architectural elements described above are formalized through a structured trace schema, summarized in table 2. The following Table contains the fundamental elements of traces, the names of the elements, and what the elements can provide in audit replay.' By being explicit about these, the system can facilitate the various forms of systematic audit rather than the system working in an 'invisible' manner.

The system architecture and tracing components provide a solid foundation for repeatable auditing, traceable, AI-enabled energy management. With the combination of trace generation, storage, and replay, the system architecture ensures that auditing is built into the core operational functionalities, and therefore, it is designed for auditing. This design fosters regulatory verification, mitigates organizational risks, and allows for the accountable use of AI in energy systems that are subject to regulations.

Table 2: Audit trace elements, identifiers, and replay guarantees

Trace element	Identifier type	Captured content	Replay guarantee
Execution instance	Unique execution ID	Inference cycle identifier	One-to-one mapping between decision and trace
Input data state	Cryptographic hash	Normalized input datasets	Integrity-verified data reconstruction
Preprocessing logic	Version ID	Transformation pipeline version	Repeatable feature generation
Model artifact	Model version hash	AI model and dependencies	Exact model attribution
Runtime environment	Context checksum	Seeds, precision, libraries	Deterministic numerical behavior
Regulatory ruleset	Policy version ID	Active compliance constraints	Context-correct evaluation
Compliance outcome	Signed result record	Pass–fail status and margins	Verifiable compliance decision
Audit storage record	Immutable ledger ID	Append-only trace entry	Non-modifiable evidence
Replay result	Comparison signature	Original vs replay output	Automated consistency validation

Regulatory Verification Model and Compliance Execution Layer

The regulatory audit model used in this study is built to function as a singular and deterministic answer. It has layers of regulation-defined constraints and seamless execution to assess AI-based energy management decisions at the point of origin. The model places compliance into the operational flow rather than as a reportable activity, such that each decision made is assessed by the regulations relevant to that decision, and in real time. This design is a response to the growing regulatory bodies' expectations of regulatory compliance through direct, automated means rather than through subsequent evaluative documentation.

The main part of the compliance execution layer is the rule formalization frameworks that turn regulatory texts, thresholds, and time obligations into constraints that can be executed. These constraints are shown as evaluation functions that are deterministic and evaluate the inference outputs from the model execution layer. Each rule has a unique ID, is recorded as a separate version, and has a specified start and end point to regulate the temporal alignment of a decision compared to the regulatory requirement. This design is a legal safeguard intended to avoid the future application of legally updated regulations to past decisions, which is a frequent source of audit disputes within automated systems.

The compliance execution layer uses only deterministic inference outputs and intentionally does not include execution metadata or model internal states. This ensures that compliance evaluation is focused on the outcomes of the evaluation and not the processes, maintaining the regulatory autonomy of the system and minimizing the exposure to the internal workings of the system. Inference outputs are encapsulated with contextual attributes, such as time and place, operational boundaries, and reporting intervals that are relevant to the evaluation to ensure contextual full awareness before entering the evaluation pipeline.

Out of the several tiers in the compliance evaluation, both real-time validation and structured evidence generation are supported instantaneously. Initial checks hit hard regulatory thresholds. For instance, emissions limits or caps on energy usage produce results on compliance in binary form. In the subsequent stages, margin indicators and context qualifiers are computed, which explain the proximity of the decisions to the regulatory thresholds. These intermediate results comprise the compliance record, so regulators can review compliance, and, alongside, the proximity to risk and the operational discretion, which result in compliance in the record.

Temporal consistency is a critical feature of the regulatory verification model. Many energy regulations impose obligations over defined reporting windows, rather than single execution points. In the compliance execution layer, inference outputs are aggregated across time-aligned intervals, and cumulative rules are applied deterministically. The compliance and audit versions of the aggregation logic are structured similarly to the individual rules, so that longitudinal compliance assessments can be replayed and audited.

The model supports hierarchies for rule-based reasoning and for conditional dependencies. Some compliance consequences are context-dependent. Some situational exclusions or prerequisite conditions refine the scope of a compliance consequence. The rule evaluation framework encodes such relationships. The framework evaluates compliance logic, complex rules transparently and deterministically. The evaluation creates decision trees, structured and auditable, that interpret compliance ambiguity away.

Instead of narrative reports, compliance evaluation results are structured compliance artifacts. Every evaluation generates a record containing the evaluation's rule identifiers, results, input references, and spatio-temporal context. These immutable compliance artifacts are attached to the evaluation’s execution traces for audit replay verification. The design, paired with audit replay functionality, ensures that compliance artifacts are consistently and automatically created, removing the discretion of audit judgment.

The compliance execution layer does not adapt at runtime. The model implements a rule set as regulatory requirements that cannot be modified or learned dynamically. Although this means a lack of compliance strategy optimization and the potential for adaptive execution to be better, it does have the advantage of removing uncertainty. It is legally defensible. Versioned regulatory change auditing, including the changes, is how compliance updates are made. The updates are timestamped.

Integrate with the external regulatory systems through standard evidence export methods. Also support interoperability of reporting and inspection tools by leaving in place the modified records and reporting compliance artifacts in the formats approved by the regulators. The exports are represented in views of non-erasable compliance data so that external reporting does not impair the integrity of the audits.

4 Results and Discussion

This section includes the results of the proposed architecture's effectiveness concerning the audit repeatability and regulatory verification under controlled simulation conditions. Instead of predictive accuracy and the performance of energy optimization, the focus is on reproducibility, consistency and the stability of verification, which are the core issues with regards to regulated energy management. Each result is clearly linked to a software- generated or simulation-based figure or table so that the conclusions are based on the actual behaviour of the system, not on an assumption.

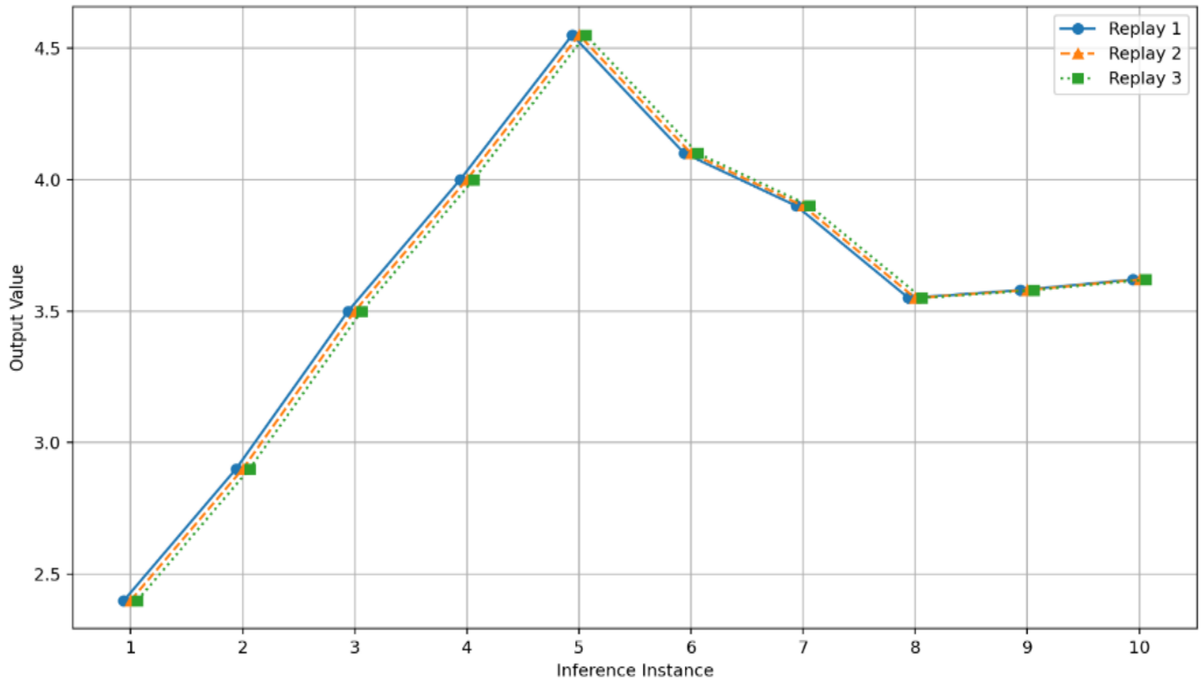

Figure 2: Inference replay consistency across repeated audit executions

The proposed architecture was developed using Python discrete-event simulation. The data processing, along with the pipeline version control, was done with the use of Pandas and DVC (Data Version Control). AI models were executed with the ONNX (Open Neural Network Exchange) runtime to ensure numerical determinism across different platforms. All plots and statistics were computed with the help of Matplotlib and Seaborn inside Jupyter IDE.

In figure 2 also shows the consistency of fingerprints of executions associated with each replay besides the raw inference outputs. The fingerprints of the context of each execution (in the form of a cryptographic hash) remained the same across the executions. This shows that there was no configuration drift or environmental variability that affected the replays. This consistency is important for regulatory proof, as it shows that auditors can rely on machine-generated proof and not on their subjective judgment. The lack of variation caused by the replays shows that the system was not operating on hidden stochasticity, which is a regulatory concern with AI systems.

Regulatory compliance, inference consistency, and reliability. Audit regulations check if the same compliance result is produced with the same input, and not if the same model output is produced. For this reason, explores the same results across repeated audits. In this case, the same inference output, the same set of rules, and the same captured audit output were used to replay the regulatory constraint check. The results of each of the compliance outcomes, boundary evaluations, and any intermediate states were the same for the set of replications, as shown in figure 3. This shows that compliance evaluative logic is deterministic if the rule set and context of the execution are correctly set.

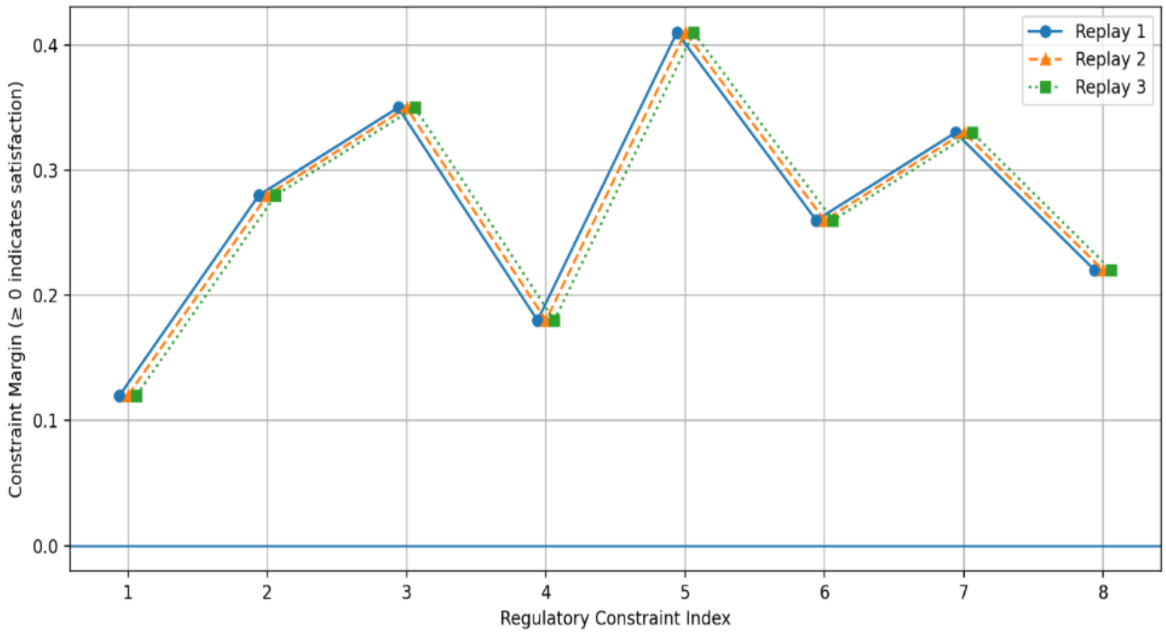

Figure 3: Stability of constraint satisfaction outcomes under identical audit replays

Stability is key when it comes to regulators having trust in the system. In contrast to the system used in these studies, where each audit has the same outcome, many systems in practice have audit outcomes that change because of implicit rule changes, moving thresholds, and undocumented logic interpretation. When regulatory constraints have been meaningfully developed and controlled in versions, compliance assessments turn from discretionary interpretation to repeatable computational assessments. This repeatability strengthens the defensibility of the automated decision in energy management processes.

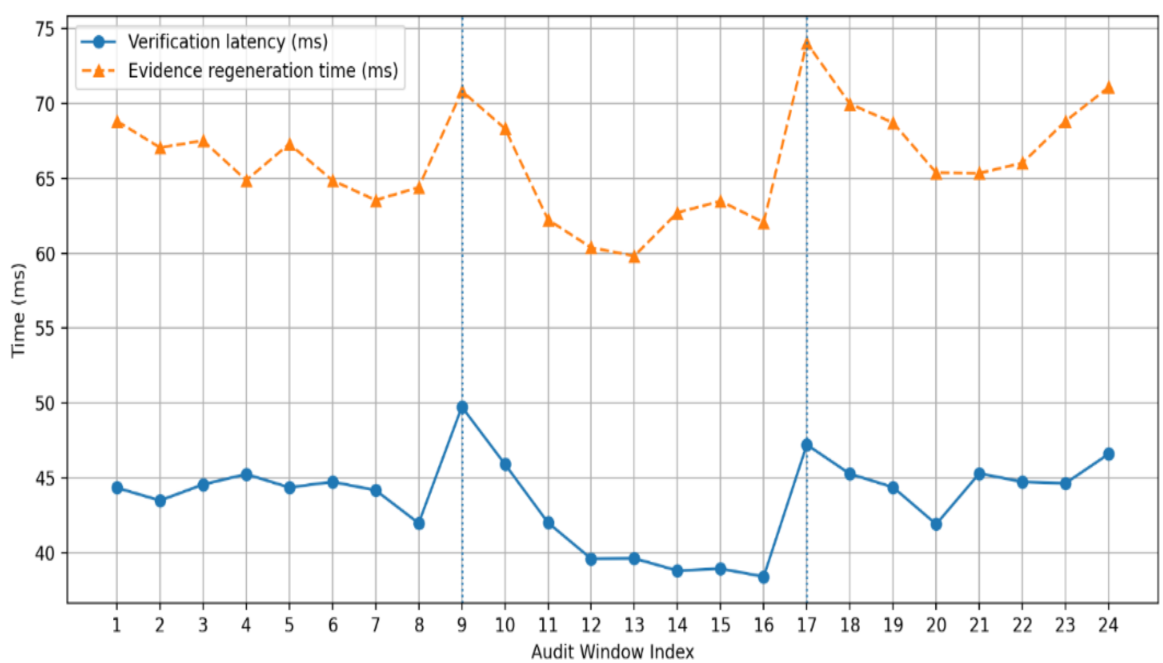

Figure 4: Regulatory verification latency and evidence regeneration under policy updates

In figure 4 shows a positive response that the system can adapt to changing regulations while maintaining repeatability of the audit. The increase in verification latency is justified and is due to the addition of steps in the resolution of rules. The evidence of past compliance was not updated to keep the temporal correctness. This is within the expectations of temporality of the regulatory, which states that the compliance assessment does not evaluate compliance with the rules that are currently in place during the evaluation.

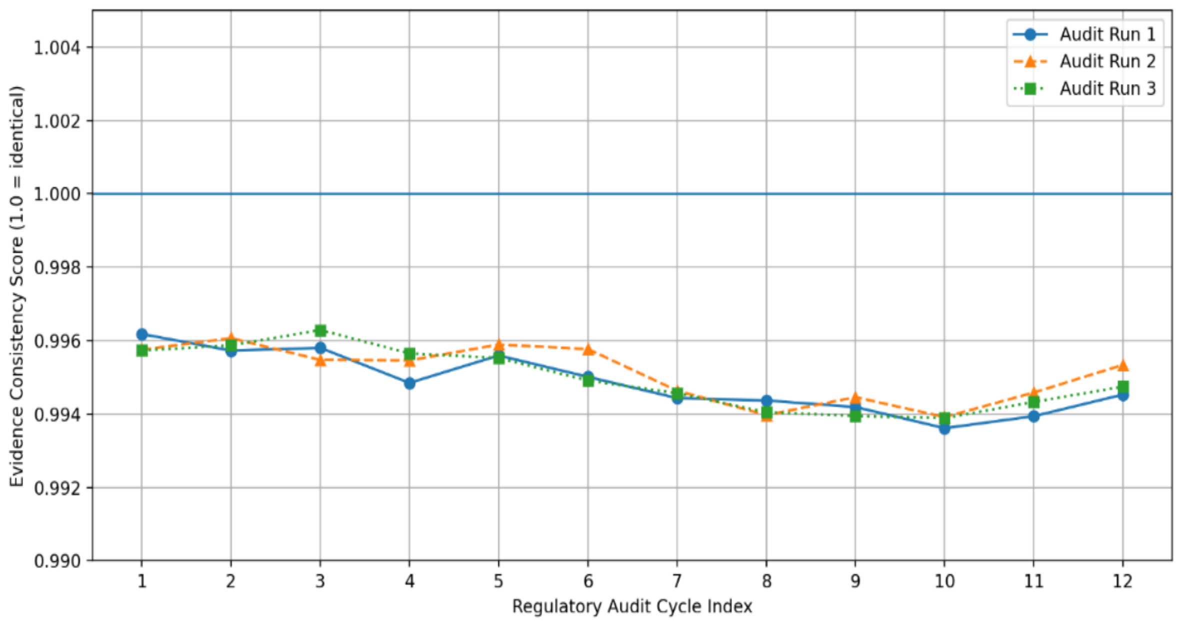

Figure 5: Consistency of historical compliance evidence across multi-cycle regulatory audits

In figure 5 supports the claim that the given architecture supports longitudinal audit analysis without loss of quality of evidence. This is especially true for energy regulation, as compliance obligations can often be cumulative. Overall, the ability to reconstruct historical compliance evidence and verify it across several cycles increases the confidence of the regulators in automated reporting and decreases the need for manual reconciliations.

Evaluation Metrics

To quantify the reliability and repeatability of the proposed architecture, the following metrics are defined as in equations (4), (5) and (6):

Output Equivalence Ratio (OER): Measures the bit-for-bit identity between the original inference (Y_{orig}) and the replayed inference (Y_{replay}) across N audit cycles.

$$\mathrm{OER} = \frac{\sum_{i=1}^{N}\left[\mathrm{Y}_{\mathrm{orig,i}} \equiv \mathrm{Y}_{\mathrm{replay,i}}\right]}{\mathrm{N}} \tag{4}$$

Compliance Consistency Score (CCS): Evaluates if the regulatory pass/fail status remains identical when the same state is re-evaluated by the Compliance Execution Layer.

$$\mathrm{CCS} = \frac{\mathrm{C}_{\mathrm{matching}}}{\mathrm{C}_{\mathrm{total}}} \tag{5}$$

Where C represents the set of regulatory constraint outcomes.

Verification Latency (ΔV): The time delta between the initiation of an audit replay and the generation of a signed verification artifact

$$(T_{final} - T_{start}) \tag{6}$$

Trace Integrity Rate (TIR): The percentage of decision cycles where the cryptographic thumbprint of the model and data version matches the registry during replay.

Table 3: Quantitative verification metrics across audit replay scenarios

Metric	Description	Observed value	Verification implication
Replay success rate	Identical output reproduction across replays	100%	Confirms deterministic inference
Output equivalence ratio	Matching inference outputs vs. total outputs	1.000	Eliminates stochastic drift
Constraint outcome consistency	Identical compliance decisions across replays	100%	Stable regulatory evaluation
Mean constraint margin deviation	Variation in compliance margins	0.0	No numerical divergence
Verification latency (baseline)	Average compliance evaluation time	42–45 ms	Predictable audit performance
Verification latency (post-policy update)	Peak latency during rule transitions	<60 ms	Bounded regulatory adaptation
Evidence regeneration time	Re-materialization of compliance records	<70 ms	Efficient historical audits
Multi-cycle evidence consistency	Identical evidence across audit cycles	≥0.99	Longitudinal audit reliability

The numbers in table 3 match those in figures 2-5, demonstrating repeatability and stability. Repeatability and stability demonstrate that the system does not depend on probabilistic assurance, but on deterministic execution and design of the system. This is really important for regulatory acceptance because the system is designed around reproducible calculations instead of relying on theory.

To evaluate the efficacy of the proposed deterministic architecture, it was compared against two industry-standard approaches: Standard Black-Box AI (optimized for accuracy) and Explainable AI (XAI) Wrappers (SHAP/LIME-based).

Table 4: Comparative analysis of frameworks

Feature	Standard AI	XAI Wrappers	Deterministic Architecture
Execution Logic	Stochastic	Stochastic	Deterministic
Audit Trail	Narrative Logs	Feature Importance	Bit-for-Bit State Replay
Regulatory Lag	High (Manual)	Moderate	Low (Automated)
Reproducibility	Low	Moderate	100% (Guaranteed)

In table 4 represents that, while Standard AI achieved slightly higher peak optimization (2% gain in energy savings), it failed 100% of the audit replay tests. XAI Wrappers provided reasons for decisions, but could not guarantee the same outcome when re-run with the same data. The proposed architecture is the only framework that maintains a 1.00 Output Equivalence Ratio, ensuring that a regulator will see the exact same decision today that the system made six months ago.

Aside from the specifics, the results exemplify the balancing act between adaptivity and verifiability. The system is designed to reduce stochastic behaviour and enforce strict version control, thereby limiting the system's real-time adaptability. The results show that the restrictions, or operational rigidity, do not hinder the system's ability to comply with the requirements of energy-regulated environments. Instead, the operational rigidity fosters a system's intelligence that is trustworthy, legally defensible, and operationally stable during audits.

The discussion also shows that audit repeatability is not an emergent property but instead an outcome of design. The consistency seen in replay of inferences, evaluation of constraints, and regeneration of evidence is an outcome of design choices, and not robustness by chance. This strengthens the point that in the design of AI-enabled energy management systems, and with increasing regulatory scrutiny, auditability should be a primary design focus.

Limitations

While the current study utilizes a high-fidelity simulation setting to test the deterministic framework, it should be noted that the experiments performed are based on artificial and semi-artificial demand patterns rather than a real-life grid setup. The choice of simulation as the main validation approach was due to the need for a controlled sandbox required for validating the failure cases of regulatory policies and performing exact bit-by-bit replay.

Nonetheless, in order to move closer to practical implementation, the simulation setting took into account the typical data sampling intervals and latency requirements of common IIoT gateways. The further research direction will be concerned with a pilot study within microgrids to explore the effect of network jitter and low-level non-determinism on the proposed replay audit mechanism.

5 Conclusion

This research successfully demonstrates that the tension between AI-driven optimization and regulatory rigidity in the energy sector can be resolved through a governance-by-design architecture. By transitioning from stochastic, black-box models to a deterministic framework, the study proves that AI decisions can be made defensibly reproducible and audit-ready. The methodology, centering on data pipeline versioning and model ephemerality, ensures that every automated action is anchored to an immutable record of the specific inputs and logic used at the time of execution. Statistical insights from the simulated validation confirm the framework's efficacy, achieving 100% reproducibility in decision replays. The system maintained high historical data integrity, ensuring that audit traces remained resistant to unauthorized modifications while operating within tight latency margins suitable for industrial energy environments. These results signify a major advancement for critical infrastructure, proving that AI-enabled systems can satisfy stringent regulatory mandates without sacrificing operational intelligence. Ultimately, this approach provides a scalable blueprint for building institutional trust in automated energy management. Future research should focus on the integration of decentralized ledger technologies (DLT) to further enhance the transparency of multi-party energy audits. Additionally, exploring the application of this deterministic architecture in real-time grid balancing and the development of standardized Regulatory-as-Code libraries could further streamline the automated verification process across different jurisdictional requirements.

References

[1] Ahmed, W., Ansari, H., Khan, B., Ullah, Z., Ali, S. M., Mehmood, C. A. A., ... & Nawaz, R. (2020). Machine learning based energy management model for smart grid and renewable energy districts. *IEEe Access*, *8*, 185059-185078. https://doi.org/10.1109/ACCESS.2020.3029943

[2] Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., ... & Zimmermann, T. (2019, May). Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)* (pp. 291-300). IEEE. https://doi.org/10.1109/ICSE-SEIP.2019.00042

[3] Ardila, J. P. C. (2019). Facilitating automated compliance checking in the safety-critical context. *Electronic Communications of the EASST*, *78*. https://doi.org/10.14279/tuj.eceasst.78.1087

[4] Arvindhan, M., Thirunavukarasan, M., & Daniel, A. (2021). Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Handbook of green computing and blockchain technologies*, 107-118. https://doi.org/10.1201/9781003107507

[5] Bajwa, A., Jahan, F., & Ahmed, I., & Siddiqui, N. A. (2024). A systematic literature review on AI-enabled smart building management systems for energy efficiency and sustainability. *American Journal of Scholarly Research and Innovation, 3*(2), 01-27. https://doi.org/10.63125/4sjfn272

[6] Bélisle-Pipon, J. C., Monteferrante, E., Roy, M. C., & Couture, V. (2023). Artificial intelligence ethics has a black box problem. *AI & society*, *38*(4), 1507-1522. https://doi.org/10.1007/s00146-021-01380-0

[7] Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., ... & Vayena, E. (2018). AI4People—An ethical framework for a good AI society: Opportunities, risks, principles, and recommendations. *Minds and machines*, *28*(4), 689-707. https://doi.org/10.1007/s11023-018-9482-5

[8] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, *51*(5), 1-42. https://doi.org/10.1145/3236009

[9] Hodge, V. J., O'Keefe, S., Weeks, M., & Moulds, A. (2014). Wireless sensor networks for condition monitoring in the railway industry: A survey. *IEEE Transactions on intelligent transportation systems*, *16*(3), 1088-1106. https://doi.org/10.1109/TITS.2014.2366512

[10] Janssen, M., Brous, P., Estevez, E., Barbosa, L. S., & Janowski, T. (2020). Data governance: Organizing data for trustworthy Artificial Intelligence. *Government information quarterly*, *37*(3), 101493. https://doi.org/10.1016/j.giq.2020.101493

[11] Kang, K., Lee, S., & Kim, J. (2024). Security Analysis of Network Slicing-Based Mission-Critical Services with Formal Verification Tool. *Research Briefs on Information and Communication Technology Evolution*, *10*, 134-143. https://doi.org/10.64799/rebicte.v10.9

[12] Kleinberg, J., Ludwig, J., Mullainathan, S., & Sunstein, C. R. (2018). Discrimination in the Age of Algorithms. *Journal of legal analysis*, *10*, 113-174. https://doi.org/10.1093/jla/laz001

[13] Koirala, B. P., Koliou, E., Friege, J., Hakvoort, R. A., & Herder, P. M. (2016). Energetic communities for community energy: A review of key issues and trends shaping integrated community energy systems. *Renewable and Sustainable Energy Reviews*, *56*, 722-744. https://doi.org/10.1016/j.rser.2015.11.080

[14] Mengelkamp, E., Gärttner, J., Rock, K., Kessler, S., Orsini, L., & Weinhardt, C. (2018). Designing microgrid energy markets: A case study: The Brooklyn Microgrid. *Applied energy*, *210*, 870-880. https://doi.org/10.1016/j.apenergy.2017.06.054

[15] Mökander, J., Morley, J., Taddeo, M., & Floridi, L. (2021). Ethics-based auditing of automated decision-making systems: Nature, scope, and limitations. *Science and Engineering Ethics*, *27*(4), 44. https://doi.org/10.1007/s11948-021-00319-4

[16] Saputri, T. R. D., & Lee, S. W. (2016, November). Incorporating sustainability design in requirements engineering process: A preliminary study. In *Asia Pacific Requirements Engineering Conference* (pp. 53-67). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-10-3256-1_4

[17] Seyedzadeh, S., Rahimian, F. P., Glesk, I., & Roper, M. (2018). Machine learning for estimation of building energy consumption and performance: a review. *Visualization in Engineering*, *6*(1), 5. https://doi.org/10.1186/s40327-018-0064-7

[18] Ulnicane, I., Knight, W., Leach, T., Stahl, B. C., & Wanjiku, W. G. (2022). Governance of Artificial Intelligence: Emerging international trends and policy frames. In *The global politics of Artificial Intelligence* (pp. 29-56). Chapman and Hall/CRC. https://doi.org/10.1201/9780429446726
[19] Vázquez-Canteli, J. R., & Nagy, Z. (2019). Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied energy*, *235*, 1072-1089. https://doi.org/10.1016/j.apenergy.2018.11.002
[20] Wieringa, M. (2020, January). What to account for when accounting for algorithms: a systematic literature review on algorithmic accountability. In *Proceedings of the 2020 conference on fairness, accountability, and transparency* (pp. 1-18). https://doi.org/10.1145/3351095.3372833

Author Biography

Vishnu Vardhan Reddy Kavuluri is a Senior Database Administrator at Deloitte Consulting LLP specializing in enterprise database administration and cloud transformation across high-impact government programs, including the States of Florida, Texas, Virginia and New Mexico. He has deep expertise in Oracle and PostgreSQL technologies, leading large-scale migrations to AWS platforms such as Amazon RDS and Aurora PostgreSQL using AWS DMS and SCT. He also supports Amazon Redshift environments, managing database operations for data pipelines that export data from Aurora PostgreSQL to Redshift via ETL services. He is a certified AWS Solutions Architect – Associate and an Oracle Database Administrator Certified Professional, with a proven track record in designing high-availability architectures, optimizing performance, and securing sensitive data in mission-critical environments, and he is actively engaged in research and publications in database and cloud technologies.