World Electric Vehicle Journal

MDPI

Review

Large Language Models for Electrical Machines: Opportunities, Challenges, and Future Directions

Haotian Jiang [1], Zhaofeng Duan [1], Gang Lei [1,*], Youguang Guo [1] and Jianguo Zhu [2]

[1] School of Electrical, Mechanical, and Biomedical Engineering, University of Technology Sydney, Sydney, NSW 2007, Australia; haotian.jiang@student.uts.edu.au (H.J.); zhaofeng.duan-1@student.uts.edu.au (Z.D.); youguang.guo-1@uts.edu.au (Y.G.)
[2] School of Electrical and Computer Engineering, The University of Sydney, Sydney, NSW 2006, Australia; jianguo.zhu@sydney.edu.au
* Correspondence: gang.lei@uts.edu.au

Abstract

Artificial intelligence has become increasingly important in the modeling, design, optimization, control, and condition monitoring of electrical machines. Existing applications primarily employ task-specific machine-learning (ML) models for performance prediction, surrogate modeling, parameter estimation, control, and fault diagnosis. Recently, large language models (LLMs) have introduced complementary capabilities in knowledge integration, reasoning, multimodal understanding, code generation, tool interaction, and workflow coordination. Rather than replacing established physics-based models, ML models, numerical optimization algorithms, or real-time controllers, LLMs can connect engineering knowledge, computational models and tools, and human expertise. This paper reviews emerging LLM applications in electrical machines and discusses their opportunities in electromagnetic and multiphysics modeling, machine design and optimization, drive control and intelligent operation, and condition monitoring, fault diagnosis, and predictive maintenance. Early studies illustrate how LLMs can assist with design-space exploration, controller tuning, and diagnostic interpretation. These applications highlight the importance of engineering verification, reliable tool interaction, and human oversight. Overall, LLMs can add engineering intelligence to existing computational intelligence, opening opportunities for more knowledge-guided, integrated, and progressively autonomous electrical-machine engineering.

Keywords: electrical machines; machine learning; large language models; multiphysics; design optimization; control; condition monitoring

Check for updates

Academic Editor: Hui Yang

Received: 27 September 2026
Revised: 17 November 2026
Accepted: 23 November 2026
Published: 27 November 2026

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

1. Introduction

Electrical machines are widely employed in transportation, industrial automation, manufacturing, household appliances, renewable energy generation, aerospace systems, robotics, and many other applications. Many applications have imposed stringent requirements on electrical machines for efficiency, power density, reliability, cost, thermal performance, and operational flexibility. Consequently, the analysis, design, control, and condition monitoring of electrical machines have become increasingly multidisciplinary and computationally demanding, involving electromagnetic, thermal, mechanical, control, and system-level considerations [1,2].

The analysis and design of electrical machines have traditionally relied on physics-based modeling approaches at different levels of fidelity. Equivalent-circuit and analyti-

cal models provide efficient tools for understanding machine characteristics, estimating performance, and conducting preliminary design. Numerical techniques, particularly finite-element analysis (FEA), have become indispensable for accurately modeling complex electromagnetic devices and nonlinear magnetic materials [3–6]. Although these methods have established a rigorous foundation for electrical machine engineering, high-fidelity modeling and multidisciplinary design can involve substantial computational cost, while the overall engineering process still requires considerable expert knowledge and human intervention.

Machine learning (ML) and other data-driven techniques have therefore attracted increasing attention as complementary tools for electrical machine analysis, design, optimization, control, and condition monitoring. Artificial neural networks (ANNs), Gaussian processes, support vector machines (SVMs), deep learning (DL) models, reinforcement learning (RL), physics-informed neural networks (PINNs), and related techniques have been investigated for applications such as magnetic field computation and thermal calculation [7–12], surrogate modeling [13–15], design optimization [4–6,16–19], and motor drive control [20–23]. In machine design, ML-based surrogate models can approximate computationally expensive FEA or multiphysics simulations and can subsequently be integrated with optimization algorithms to accelerate the exploration of large design spaces [5,6]. Many of the ML applications discussed above address defined quantitative tasks, such as motor-temperature estimation [12] and efficiency-map prediction [13]. This task-specific formulation should be distinguished from predictive generalization to unseen operating conditions or machine designs. Interpreting engineering requirements, defining objectives and constraints, and coordinating simulation and optimization tools require additional engineering input and software integration beyond a predictive model alone. LLM-based systems offer a potential means of supporting these activities through knowledge interaction and workflow coordination, while their outputs and proposed actions remain subject to engineering verification.

The recent emergence of large language models (LLMs) introduces a fundamentally different paradigm for AI-assisted engineering. Beyond natural-language processing, modern LLMs demonstrate increasingly powerful capabilities in knowledge interaction, reasoning, code generation, workflow planning, and human–machine interaction. The application of LLMs has recently expanded rapidly into power and energy systems, with emerging studies covering power-system modeling, optimal power flow, forecasting, grid operation and control, fault diagnosis, energy management, and decision support [24–28]. More recent studies have further explored LLM-based agents for knowledge-grounded power-system operation and autonomous algorithm design, indicating a broader transition from task-specific LLM assistance toward tool-integrated and agent-based engineering intelligence [29,30].

Despite these rapid advances in power systems, the application of LLMs to electrical machines and drive systems remains at an earlier stage, with their potential roles in multidisciplinary modeling, design and optimization, drive control, and condition monitoring yet to be systematically investigated.

Building on previous reviews of electrical-machine design optimization [6] and EV drive-system optimization and AI-enabled design platforms [3], this paper reviews emerging LLM applications and develops a perspective on electrical-machine engineering. It focuses on the functional roles of LLMs, their interaction with established computational methods and human engineers, and the distinction between demonstrated applications and prospective extensions. The discussion covers electromagnetic and multiphysics modeling, machine design and optimization, drive control and intelligent operation, and condition monitoring and predictive maintenance. On this basis, the paper identifies technical chal-

lenges and research priorities toward knowledge-guided, integrated, and progressively autonomous engineering workflows.

The remainder of this paper is organized as follows. Section 2 reviews the fundamentals, emerging applications, and current research gaps of LLMs in electrical machines. Section 3 discusses their opportunities in four aspects, with particular emphasis on the complementary roles of physics-based models, ML/DL models, numerical methods, and LLMs. Section 4 discusses four key challenges. Section 5 explores future research directions toward autonomous engineering agents, followed by the conclusions in Section 6. Through this review and forward-looking discussion, the paper aims to establish a conceptual framework and research roadmap for LLM-enabled intelligent engineering of electrical machines and drive systems.

2. LLMs for Electrical Machines: Overview and Research Gaps

2.1. LLM-Enabled Engineering for Electrical Machines

The following terminology distinguishes model capabilities from workflow organization. An LLM denotes the underlying language model, while a multimodal LLM can also process supported non-text inputs, such as images. An LLM-based agent combines a model with task planning, tool interaction, and feedback to pursue a specified task. A multi-agent system coordinates agents assigned different roles. An autonomous engineering agent is delegated a defined range of engineering tasks with reduced step-by-step human intervention. Multimodality, multiple agents, and autonomy describe different system properties rather than successive levels of engineering capability. Here, "engineering intelligence" is used as a functional term encompassing problem formulation, workflow coordination, and decision support. These functions involve translating requirements and domain knowledge into objectives, constraints, and modeling assumptions; organizing interactions among suitable models and tools; and interpreting evidence and trade-offs to support human engineering decisions. LLMs may assist these functions alongside specialized computational methods and human expertise. Their formulations, computational plans, and recommendations remain subject to numerical and engineering verification.

Figure 1 illustrates a brief conceptual framework of LLM-enabled engineering for electrical machines. As shown, LLMs can potentially serve as an intelligent reasoning and coordination layer that connects engineering knowledge and data, human expertise, and specialized engineering tools.

With such an architecture, an LLM or an LLM-based agent may formulate computational tasks, generate or modify code, call numerical or multiphysics solvers, interact with optimization algorithms, interpret simulation or experimental results, and use the resulting information to guide subsequent engineering decisions. Engineers define the objectives and constraints, provide domain knowledge, and review the proposed assumptions and tool actions before execution. They check intermediate results for physical plausibility and consistency with the stated requirements. If a check fails, they can correct the inputs or request a revised script before rerunning the calculation. They make the final engineering decision based on the verified results. Such human–LLM–tool interaction provides a promising paradigm for complex, multidisciplinary engineering workflows, while current evidence suggests that LLMs are better regarded as engineering "co-pilots" than as fully autonomous designers. This paradigm is particularly relevant to electrical machines, where electromagnetic, thermal, mechanical, power-electronic, control, and data-driven models must often be integrated throughout design and operation. The emerging applications of LLMs in electrical machines are reviewed in the following subsection.

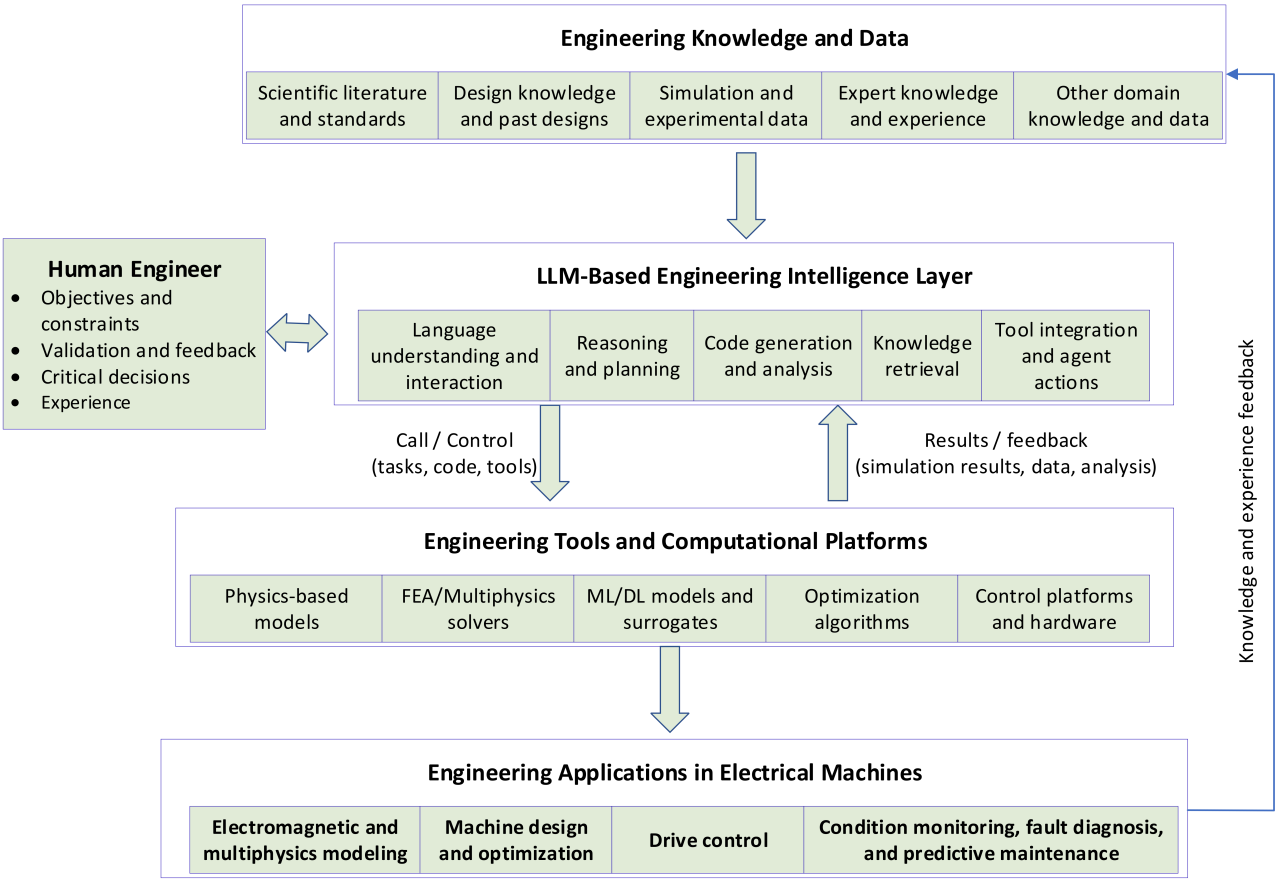


Figure 1. A conceptual framework of LLM-enabled engineering for electrical machines.

2.2. Emerging Applications of LLMs in Electrical Machines

We searched IEEE and MDPI publication platforms and Google Scholar for studies on applications of LLMs to electrical machines and associated drive systems. The main search terms included "LLM," "electrical machines," "design optimisation," "control," and "fault." No publication-year restriction was applied. The literature search was last updated around 20 September 2026.

Studies were included in Table 1 if they reported a direct application of an LLM to an electrical machine or its drive system. Studies outside this scope were excluded from the table. We aimed to include all directly relevant studies identified through these searches. Table 1 groups the studies by their primary engineering task into machine design and optimization, power electronics and motor control, and condition monitoring, fault diagnosis, and maintenance.

Table 1. Existing Applications of LLMs for Electrical Machines.

Ref.	Year	Application	Motor	LLM/AI Technique	Functional Role of LLM	Validation and Key Findings
[31]	2026	Machine design and optimization	IPMSM	Multimodal GPT-5.4 + CMA-ES	Multimodal design-space reasoning and dimensionality reduction, with FEM	IPMSM FEM study. LLM-guided CMA-ES achieved a better objective value with 4900 evaluations versus 6400 for full-space CMA-ES.

Table 1. *Cont.*

Ref.	Year	Application	Motor	LLM/AI Technique	Functional Role of LLM	Validation and Key Findings
[32]	2026	Machine design and optimization	IPMSM	LLM + RAG + Bayesian optimization	Knowledge-guided optimization agent, with FEM	MATLAB R2023b–Ansys Maxwell 2022 R2 simulations. RAG–BO used 182 iterations with language guidance and 242 without it. MO-AHA used 351.
[33]	2026	Power-electronic design	Motor drive	Generative AI + multi-agent LLM framework	Multi-agent workflow coordinator for power-electronic circuit design	Design checks on a 1.5 kW drive showed 100% schematic-to-PCB logical connectivity.
[34]	2024	Electromechanical design and control	DC motor	GPT-4o + prompt engineering	Motor modeling and controller-design assistant	Direct PID tuning met simulated DC motor targets in 0 of 200 trials. Guided analytical tuning matched the reference response on a separate model.
[35]	2025	Drive-control tuning	PMSM	Domain-trained LLM Tuning Assistant	Knowledge-trained LLM-based Tuning Assistant	MATLAB/Simulink tests. Peak-to-peak torque ripple was 14.2% (tuning assistant), 67.7% (ChatGPT), and 10.9% (Copilot) of rated torque.
[36]	2025	Real-time adaptive control	Servo motor	o3-mini + GPT-4.1-mini agentic AI	Closed-loop adaptive tuning agent	QUBE-Servo 2 bench tests demonstrated online LLM-assisted PID tuning, with reduced tracking errors in the illustrated runs.
[37]	2025	Intelligent drive control	PMSM	LLaMA3 + SVM + Q-learning	Supervisory control strategy selector	PLECS tests. LLaMA3 sometimes selected voltage droop. SVM and Q-learning generally selected FOC, which tracked current more accurately.
[38]	2025	PID control diagram generation	DC Motor	DeepSeek + multi-agent framework	Control diagram generation, validation, and correction	NCSLab simulation with manual fine-tuning. For DC motor position control, diagram generation succeeded in 80% of attempts with multiple agents versus 70% with the design agent alone.
[39]	2026	Condition monitoring and fault diagnosis	Induction motor	Multimodal LLM + structured prompting	Direct multimodal diagnostic reasoning assistant	Five published motor-current spectra with diagnostic labels withheld. Four conclusive responses and one inconclusive response.

Table 1. *Cont.*

Ref.	Year	Application	Motor	LLM/AI Technique	Functional Role of LLM	Validation and Key Findings
[40]	2026	Thermal condition monitoring and fault diagnosis	Induction motor	CNN + RAG + LLM	Knowledge-grounded diagnostic interpretation and prescriptive reporting	Thermal-image study. Semantic similarity between retrieved context and reports was 0.755 (Gemini Flash 2.0) and 0.783 (Zephyr-7B).
[41]	2025	Fault diagnosis and maintenance	EV drive motor	Fine-tuned Qwen2.5-7B + LoRA/QLoRA	Domain-adapted LLM fault diagnosis model	On bearing fault Dataset D, test accuracy was about 93.4% after multi-source pretraining and fine-tuning, versus 92.5% with training on D alone.
[42]	2026	Fault diagnosis	EV drive motor	Decision tree + multi-agent LLM	Explainability and diagnostic-reasoning layer	Ten participants rated explanations based on simulated drive data. Explanation accuracy averaged 3.71/5 (multi-agent) versus 3.43/5 (single agent).
[43]	2026	Fault diagnosis and maintenance	Motor	DCCNN-LTransformer + fault KG + Qwen3-7B	Knowledge-grounded maintenance decision agent	On 20 maintenance questions, the LLM with a knowledge graph achieved a mean entity-consistency score above 80.

(1) Machine Design and Optimization

The application of LLMs to electrical-machine design and optimization has only recently emerged. Sekiyama et al. integrated a multimodal LLM with conventional optimization and FEA for interior permanent magnet synchronous motor (IPMSM) topology and parameter optimization, using the LLM to identify significant design variables and reduce the dimensionality of the design space [31]. Guesmia et al. combined an LLM-based agent with Retrieval-Augmented Generation (RAG), Bayesian optimization, and FEA, enabling previous simulation results and engineering knowledge to guide the optimization process [32]. These two studies provide early proofs of concept for using LLMs to support design-space interpretation and knowledge-guided optimization alongside FEA and numerical optimizers. Their findings are specific to the investigated IPMSM configurations and evaluation settings; broader applicability across machine types and industrial conditions requires further validation.

(2) Power Electronics and Motor Control

LLMs have also been investigated for power-electronic design and motor control. Hetrich et al. employed a multi-agent LLM framework to coordinate multiple stages of power-electronic circuit design for motor drives, demonstrating the potential of LLMs to coordinate engineering workflows [33]. In control applications, existing studies range from general-purpose assistance for motor modeling and controller design, through domain-specific controller-tuning assistants, to agentic approaches for adaptive PID tuning [34–36]. LLMs have also been explored for selecting supervisory control strategies and for adapting different timescales in model-predictive controllers [37]. Ye and Liu used multiple LLM-based agents to generate PID control diagrams from natural language specifications and

evaluated a DC motor position-control example in NCSLab simulations [38]. Overall, these studies indicate that LLMs are particularly promising for engineering assistance, supervisory decision-making, parameter adaptation, and tool coordination, while conventional controllers remain responsible for high-bandwidth and safety-critical control actions.

(3) Condition Monitoring, Fault Diagnosis, and Maintenance

Condition monitoring and maintenance represent another emerging area for LLM applications. Multimodal LLMs have been directly applied to interpret motor-current spectra and generate diagnostic explanations [39], while hybrid architectures have combined conventional diagnostic models with RAG and LLMs, in which task-specific models perform fault classification, and LLMs provide knowledge-grounded interpretation and prescriptive maintenance reporting [40]. Other studies have investigated domain-adapted LLMs for fault identification, multi-agent LLMs for explainable diagnosis, and knowledge-graph-grounded LLMs for maintenance decision making [41–43]. Collectively, these studies demonstrate progression from fault recognition toward multimodal interpretation, explainable diagnosis, knowledge-grounded reasoning, and prescriptive maintenance.

2.3. Research Gaps

Within the literature summarized in this review, direct LLM applications focus primarily on design optimization, control assistance, fault diagnosis, and maintenance. Reported evaluations range from selected demonstrations and simulation studies to laboratory experiments, with substantial differences in tasks, datasets, operating conditions, and comparison protocols. These results provide evidence for particular implementations, while broader transferability across machine types and operating conditions remains insufficiently established. Direct evidence for LLM-assisted electromagnetic and multiphysics modeling of electrical machines is also limited, although related engineering applications provide useful methods for further investigation.

Several existing studies already integrate LLMs with retrieval systems, diagnostic models, optimizers, or numerical solvers. A central research gap is the reliable coordination of these capabilities across engineering stages, including consistent requirements and modeling assumptions, traceable information, verification of intermediate results, and effective human intervention. Evidence remains limited for such coordination across modeling, design optimization, control, and condition monitoring within integrated electrical-machine workflows. These gaps motivate the opportunities and research directions toward knowledge-guided, integrated, and progressively autonomous engineering discussed in the following sections.

3. Opportunities of LLMs for Electrical Machines

Figure 2 illustrates four major application areas of LLMs in electrical machines. As shown, LLMs offer new opportunities to enhance the modeling, design, control, and maintenance of electrical machines through their capabilities in knowledge integration, reasoning, multimodal understanding, tool interaction, and workflow coordination. These opportunities can be broadly classified into four major application areas:

(1) Electromagnetic and multiphysics modeling;
(2) Machine design and optimization;
(3) Drive control and intelligent operation;
(4) Condition monitoring, fault diagnosis, and predictive maintenance.

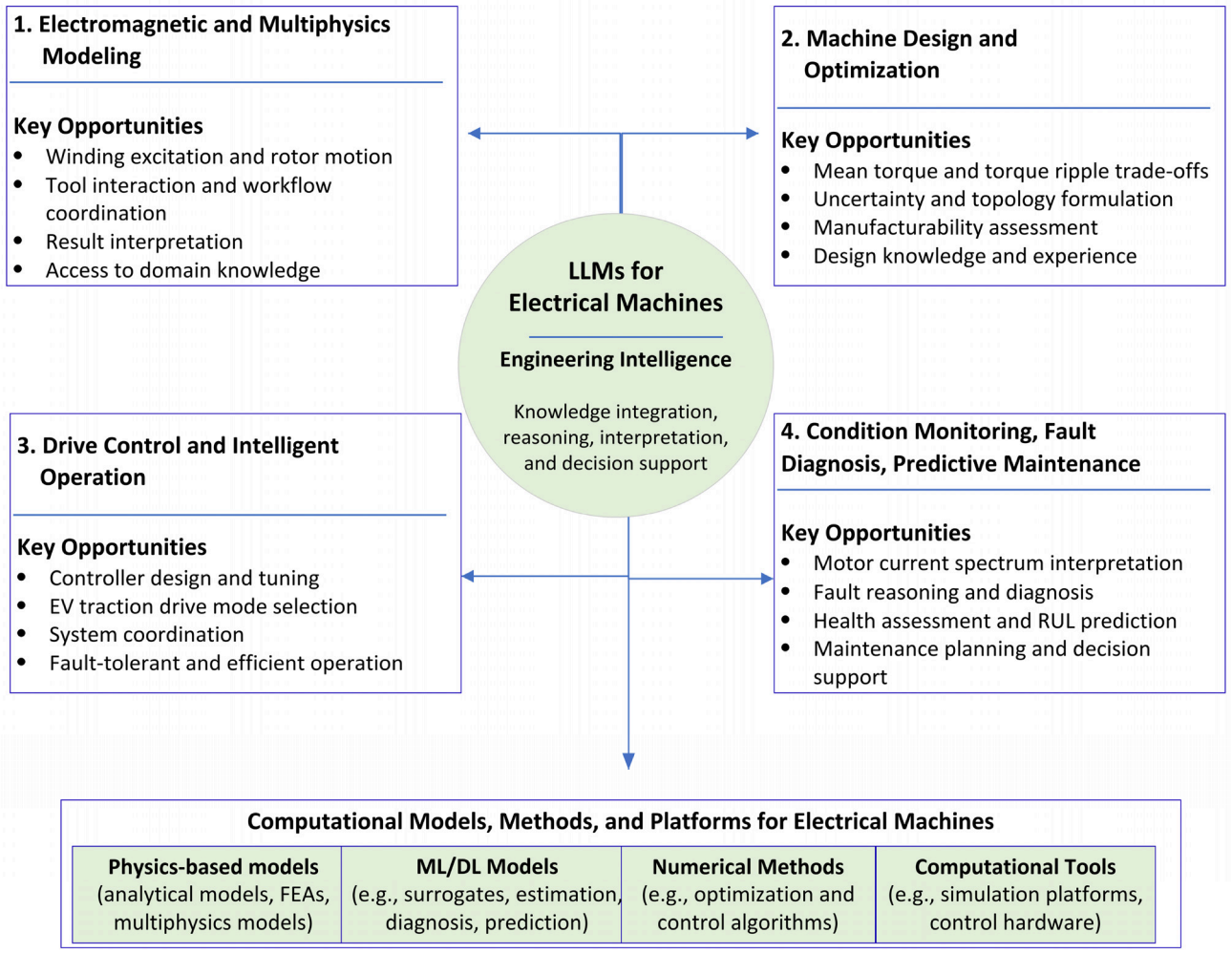


Figure 2. Overview of the four major application areas of LLMs in electrical machines.

The following discussion distinguishes direct LLM applications to electrical machines and drive systems, evidence from related engineering fields, and prospective extensions proposed in this review. Reported findings are interpreted within their validation settings, while established physics-based, ML/DL, and optimization methods provide the computational foundation for the proposed integrations.

3.1. Electromagnetic and Multiphysics Modeling

Electromagnetic and multiphysics modeling provides the quantitative foundation for analyzing and designing electrical machines. High-fidelity FEA is widely used to evaluate magnetic fields, torque, and losses, while coupled electromagnetic–thermal–mechanical models are increasingly required for many applications, such as the in-wheel motors for EVs. However, constructing these models remains strongly dependent on engineering expertise because the designer must define geometries, materials, excitations, boundary and motion conditions, meshes, solver settings, and post-processing procedures. LLMs, therefore, offer an opportunity to create a more intelligent interface between engineering requirements and established numerical modeling environments.

One promising direction is LLM-assisted FEA model construction and automation. An LLM-based agent could interpret natural-language requirements, retrieve relevant modeling knowledge, generate simulation code or scripts, and interact with FEA and multiphysics platforms through APIs or structured tool interfaces. For electrical machines, such capabilities could automate parts of the workflow, from geometry and material definition to

winding and excitation assignment, meshing, solver configuration, parametric analysis, and post-processing, and could also enable future integration with commercial platforms such as Ansys Maxwell. Recent frameworks such as AutoFEA and ALL-FEM have demonstrated the feasibility of using LLMs and agentic AI for FEA input/code generation and increasingly automated simulation workflows [44,45], while COMSOL-MCP has further demonstrated direct interaction between LLM agents and a commercial multiphysics environment through structured tool calls [46]. Nevertheless, current systems remain susceptible to hallucinations, incorrect model settings, and spatial-grounding errors. Notably, COMSOL-MCP reported a case in which incorrect boundary-condition entity assignment resulted in a solver-converged but physically invalid solution [46]. Therefore, LLMs should presently be regarded as assistants or coordination agents for constructing, operating, and interpreting FEA models rather than replacements for high-fidelity numerical solvers or expert engineering verification.

A more realistic future architecture is consequently the integration of LLM intelligence, task-specific ML/DL models, and high-fidelity physics-based solvers, with each providing complementary capabilities. An LLM-based agent could coordinate electromagnetic, thermal, mechanical, and acoustic models; manage data exchange among different solvers; select between high-fidelity FEA and faster ML/DL surrogate models according to the required accuracy; and interpret outputs such as field maps, temperature distributions, stress maps, waveforms, and numerical performance indices. A script-based interface can connect these components. The LLM helps translate engineering requirements into parameterized model inputs and drafts an Ansys Maxwell script. Before execution, the engineer checks dimensions and units and confirms that excitations, boundary conditions, and motion settings are assigned to the intended model regions. The engineer runs the script and returns the CSV results and run logs for analysis. High-fidelity FEA provides training data for the surrogate model. The surrogate screens candidates within a design domain where its prediction error has been assessed, and selected designs are re-evaluated using FEA.

In the case study in Section 3.2.4, the engineer asked the LLM to build a baseline Maxwell model using the parameters reported by Sun et al. [47] and to simulate one electrical cycle with 120 time steps. Listing 1 shows how the machine parameters and the requested time discretization were represented in the generated Maxwell script.

Listing 1. Abridged Ansys Maxwell script for the transient setup.

```
STATOR_SLOTS = 18
POLE_PAIRS = 10
STEPS = 120

local_variable(
    design, 'f_electric',
    'speed_rpm/60rpm*%d*1Hz' % POLE_PAIRS
)
local_variable(design, 'samples_per_cycle', str(STEPS))
local_variable(design, 'stop_time', '1/f_electric')
local_variable(design, 'time_step', 'stop_time/samples_per_cycle')
```

The script used the slot and pole counts to construct the machine geometry. It passed the calculated stop time and time step to the Maxwell transient setup. The engineer ran the script and compared the simulated back EMF and torque with the published prototype measurements.

Another promising application is magnetic-material modeling, where specialized ML/DL models can learn quantitative mappings of core loss and hysteresis behavior from large experimental datasets, especially in high-frequency regimes [48–51]. LLM-based agents could assist in integrating material knowledge, selecting and constructing appropriate models, interpreting their performance, and potentially generating and testing new physics-informed model formulations. Developing these models will benefit magnetic-field and multiphysics analysis of electrical machines.

3.2. Machine Design and Optimization

Electrical-machine design and optimization have remained important and active research areas for several decades, traditionally relying on physics-based models, FEA and multiphysics solvers, and various optimization models (such as deterministic, robust, and multiobjective models) and strategies (such as space reduction and multilevel strategies) [4–6,52–54]. These methods provide powerful quantitative capabilities for performance evaluation and design-space exploration, but the optimization formulation—including the selection of machine topology, materials, design variables, parameter ranges, objectives, constraints, and uncertainties—still depends heavily on engineering knowledge and experience. Such knowledge is often distributed across the scientific literature, standards, historical designs, simulation and experimental records, manufacturing documents, and expert experience, making it difficult to utilize systematically in computational design.

LLMs can complement existing design and optimization methods by extracting, integrating, and structuring engineering knowledge and using it to support design formulation and decision-making, as illustrated in Figure 3. Structured knowledge derived from previous designs and engineering experience can assist in initial design selection and design-space formulation, while manufacturing and quality information can support uncertainty characterization (including types and parameters) and manufacturability assessment. Physics-based models, task-specific ML/DL models, and numerical optimization algorithms remain responsible for quantitative performance prediction and optimization. The LLM's distinctive role is to provide engineering design support that helps determine what to optimize, how to formulate the design and uncertainty spaces, and how to interpret and evaluate the resulting designs. The following subsections discuss LLM support for deterministic, robust, and topology design optimization and present an illustrative case study.

3.2.1. Deterministic Design Optimization

Conventional deterministic design optimization generally begins with a predefined machine configuration and a set of design variables and parameter ranges. The quality and computational efficiency of the resulting optimization, therefore, depend strongly on how this design space is formulated. An excessively broad or poorly defined design space may substantially increase computational cost, whereas an inappropriate space may exclude promising designs. LLMs provide a potential mechanism for retrieving and structuring knowledge from the technical literature, standards, historical designs, material databases, simulation and experimental records, and expert experience. Information on machine topologies, materials, key dimensions, operating conditions, performance indices, and design constraints can be transformed into machine-accessible design knowledge to support topology and material selection, initial design generation, identification of significant variables, and definition of physically meaningful parameter ranges.

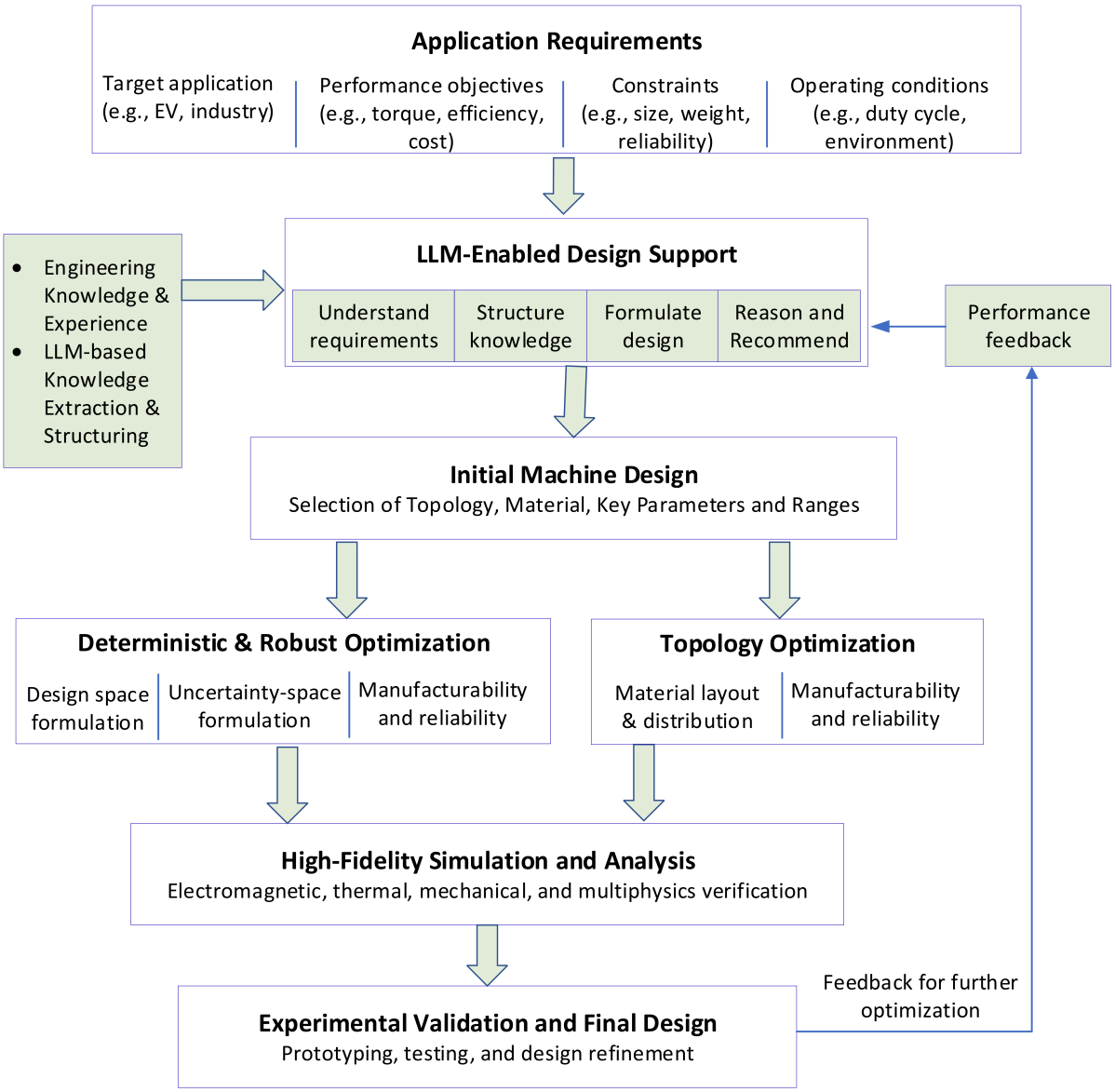


Figure 3. LLM-enabled knowledge-guided framework for design optimization of electrical machines.

Once the design space has been formulated, established physics-based models, FEA/multiphysics solvers, ML/DL surrogates, and numerical optimization algorithms can perform quantitative design exploration and optimization. LLMs may further assist in selecting appropriate models and tools, managing computational workflows, and interpreting optimization results and trade-offs. Thus, their principal contribution is not to replace numerical optimizers, but to make optimization more knowledge-guided, particularly in determining what to optimize and where promising regions of the design space may lie.

3.2.2. Robust Design Optimization

Robust design optimization (RDO) introduces an additional challenge because uncertainties in manufacturing tolerances, material properties, assembly variations, and operating conditions must be identified and appropriately represented. In practice, much of this uncertainty information is dispersed across tolerance specifications, material datasheets, manufacturing and quality-control records, inspection reports, historical production data, and expert or workshop experience [55–62]. LLMs could extract and structure such information into candidate uncertainty variables, parameter ranges, statistical characteristics, correla-

tions, and manufacturing constraints, thereby supporting the formulation of a more realistic uncertainty space. Physics-based models, ML/DL surrogates, uncertainty-quantification methods, and numerical RDO algorithms would then remain responsible for evaluating and optimizing design robustness.

Another promising role is manufacturability assessment. A numerically robust optimum is not necessarily practical to manufacture. By integrating manufacturing rules, process limitations, historical production experience, and known quality issues, LLMs could assist engineers in identifying candidate solutions that may be difficult or costly to manufacture and recommend them for further physics-based, statistical, or experimental verification. Consequently, LLMs could contribute to RDO at two complementary stages: before optimization, by supporting uncertainty-space formulation, and after optimization, by assisting the engineering assessment of manufacturability and practical robustness [4–6].

3.2.3. Topology Optimization

Topology optimization provides greater design freedom than conventional parameterized optimization by directly optimizing material distributions, but its large design space and complex electromagnetic, multiphysics, and manufacturing constraints also make problem formulation particularly challenging [63,64]. To improve the optimization efficiency, many machine learning models, especially deep learning models, have been employed [65–70]. LLMs could provide a higher-level knowledge and reasoning layer by integrating application requirements, previous designs, material information, manufacturing guidelines, standards, and engineering experience. Such knowledge could help define the design domain, candidate materials, objectives, constraints, operating conditions, and manufacturing rules before numerical optimization begins. The LLM would therefore help determine which topology-optimization problem to solve, while established topology-optimization algorithms and physics-based models remain responsible for quantitative material-layout optimization and performance evaluation.

A particularly promising opportunity is the use of multimodal LLMs for topology interpretation and adaptive search-space refinement. Topology optimization generates heterogeneous information, including material layouts, electromagnetic field maps, thermal and stress distributions, performance metrics, sensitivity information, and optimization histories. A multimodal LLM could interpret these outputs together to identify significant design regions and relate geometric features to physical behavior. An initial example demonstrated the integration of a multimodal LLM with simultaneous parameter and topology optimization of an IPMSM, in which the LLM interpreted rotor images and optimization information to identify active topology variables and reduce the effective search space [31]. Future approaches could combine such reasoning with ML/DL surrogate or field-prediction models to accelerate computation, while retaining high-fidelity FEA and multiphysics models for quantitative verification.

LLMs could also contribute to manufacturability assessment and iterative design refinement. Free-form optimized topologies may contain thin features, disconnected regions, complex material interfaces, or other structures that are difficult to manufacture and assemble. By combining manufacturing knowledge, process limitations, historical experience, and optimization results, an LLM could identify potential manufacturing issues and suggest modifications to the design domain or constraints for subsequent numerical verification.

3.2.4. Illustrative Case Study of LLM-Assisted Machine Design Optimization

This case study illustrates how an LLM can assist in the design optimization of a surface-mounted permanent magnet synchronous machine (SPMSM). Figure 4a shows

the script-based workflow and the division of responsibilities between the engineer and the LLM.

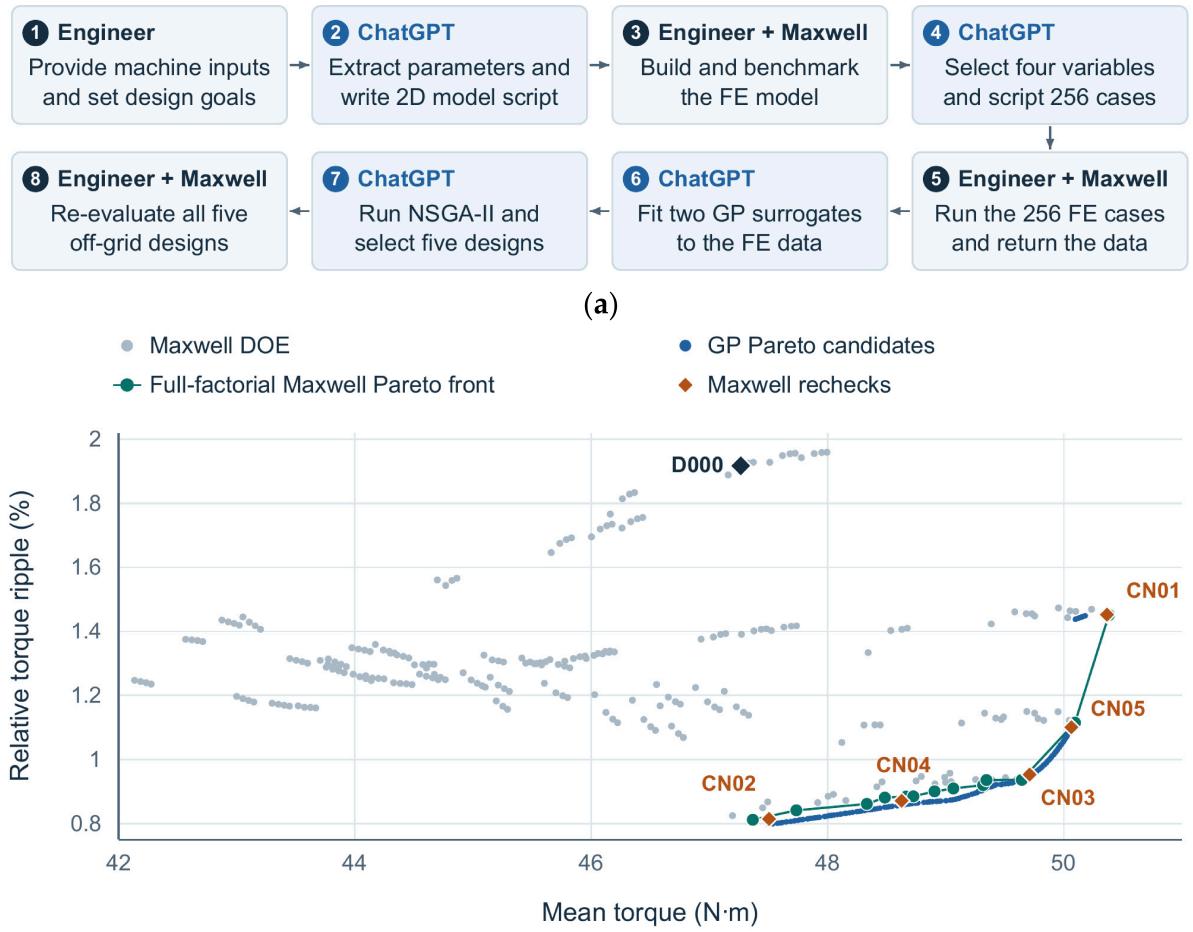


Figure 4. (**a**) Script-driven workflow and division of responsibilities. (**b**) Full-factorial Maxwell results (256 designs), GP Pareto candidates, and five Maxwell rechecks at 25 A peak phase current and 120 rpm. D000 is the baseline design.

We used OpenAI's Codex desktop application for the case study from 7 to 9 November 2026. The model designation recorded for the baseline reconstruction was "ChatGPT-6.1 Sol." The principal prompts and workflow instructions are provided in the Supplementary Materials.

First, Codex assisted in extracting the machine parameters from the open-access article by Sun et al. [47] and generated an Ansys Maxwell 2024 R1 script for a two-dimensional transient FEA model of the 18-slot, 20-pole SPMSM. We ran the model and compared the results with the published prototype measurements. At 120 rpm, the simulated and published fundamental peak line-to-line back EMFs were 27.93 and 27.73 V, respectively. At 25 A peak phase current, the corresponding mean torques were 47.27 and 46.07 N·m, and the relative torque ripples were 1.92% and 2.25%. The reconstruction used a back-EMF-calibrated equivalent winding. The loaded simulation assumed 120 rpm and zero direct-axis current.

Second, we specified the objectives of maximizing mean torque and minimizing relative torque ripple and requested four levels per variable. We adopted the four design variables proposed by Codex (magnet pole–arc ratio, magnet thickness, slot opening angle, and stator yoke thickness). Codex generated the batch script for all 256 full-factorial combinations.

We ran the cases at the same loaded operating point, using one electrical cycle and 120 time steps per case, and provided the FEA results to Codex.

Third, Codex generated code to fit Gaussian process surrogates [71], run NSGA-II [72] over separate domains for gapped and contacting magnets, and select five off-grid Pareto candidates for verification.

Fourth, we re-evaluated the five designs in Maxwell using the same settings. Prediction RMSEs were 0.0034 N·m for mean torque and 0.0099% for ripple. Four candidates entered the combined FE Pareto front, which grew from 13 to 16 designs (Figure 4b). This single case illustrates LLM-assisted motor design optimization from model reconstruction through engineer-run FEA verification. Estimating a statistical success rate for script generation would require repeated generation trials. Quantifying end-to-end computational speedup would require a matched comparison with a conventional workflow.

3.3. Drive Control and Intelligent Operation

Electrical drive control is fundamentally different from many other LLM applications because current, torque, speed, and position control require deterministic execution at short sampling intervals. Established approaches such as field-oriented control, direct torque control, model predictive control, and PID control are specifically designed for real-time operation, while ML/DL and reinforcement learning methods can provide additional capabilities for parameter estimation, nonlinear modeling, adaptive control, and operating-state prediction [73–77].

At the design stage, LLMs can assist with controller selection, parameter initialization, code generation, and tuning. Proposed controller changes should be checked against stability and dynamic performance requirements in simulation and, where appropriate, on a test bench before deployment. During operation, validated controllers execute the time-critical control loops. At the supervisory level, an LLM can process operating summaries asynchronously and propose bounded changes to references or controller parameters, or recommend a switch between approved operating modes.

This supervisory architecture can be extended from machine control to operation of the complete drive system. Drive performance depends on interactions among the machine, power converter, controller, thermal conditions, and operating requirements. In an EV traction drive, for example, an LLM could interpret drive-cycle demands, inverter loading, and motor temperature to recommend an operating mode for supervisory review.

A deterministic supervisor should apply an update only if it arrives within its validity window and satisfies predefined limits on operating conditions, parameter values, and rates of change. If a proposal is late, invalid, or missing, the drive should retain the last validated setting while it remains safe and otherwise enter a predefined fallback mode. Supervisory timing requirements should account for inference latency and jitter. The real-time control loops execute independently of LLM inference.

3.4. Condition Monitoring, Fault Diagnosis, and Predictive Maintenance

Condition monitoring, fault diagnosis, and predictive maintenance represent particularly promising applications of LLMs because electrical-machine health assessment increasingly involves heterogeneous information from multiple sources [39–43]. Figure 5 illustrates a framework for LLM-assisted condition monitoring and predictive maintenance for electrical machines.

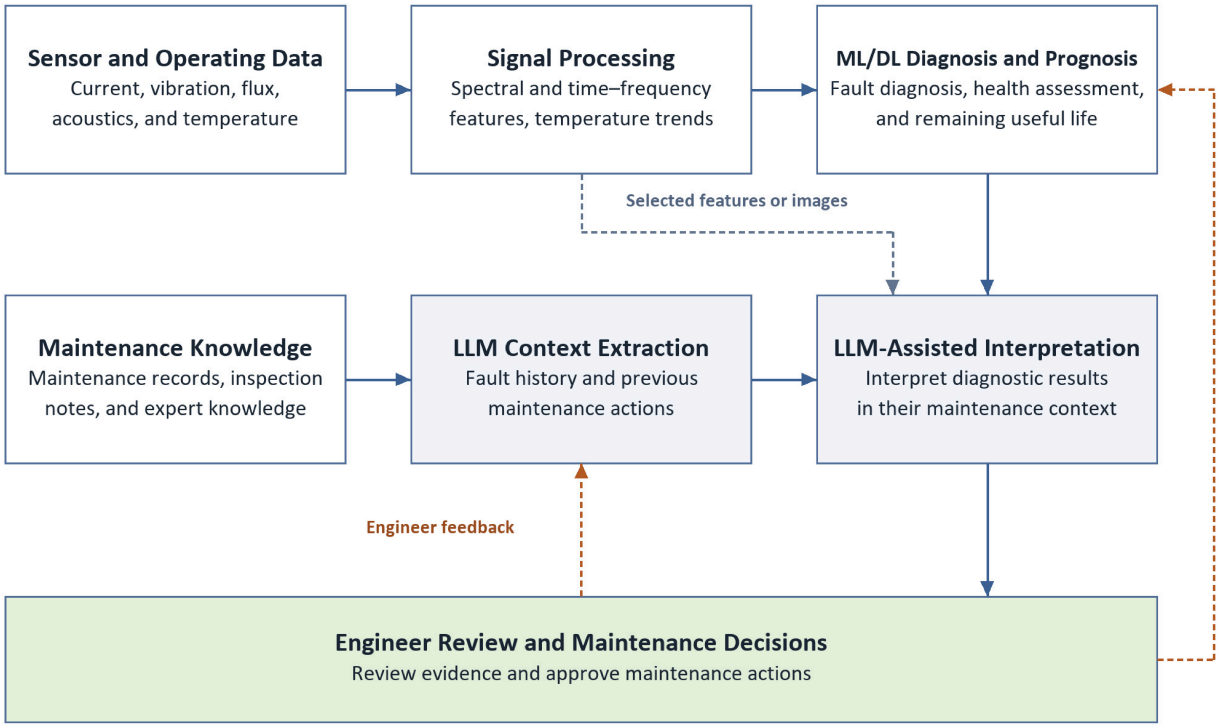


Figure 5. LLM-assisted condition monitoring and predictive maintenance framework for electrical machines.

Current, vibration, flux, and acoustic waveforms can be processed into spectral or time–frequency features [78,79]. Temperature measurements can be summarized as trends. Task-specific diagnostic and prognostic models can use these sensor-derived inputs to assess machine health. For LLM-assisted interpretation, selected features and model outputs can be provided as structured summaries that specify units, measurement intervals, and operating conditions such as speed and load. Spectra, spectrograms, and thermal images can also be supplied to a multimodal LLM.

Maintenance records, inspection notes, and expert knowledge provide a second source of information. An LLM can extract structured context, such as fault histories and prior maintenance actions. Diagnostic models can combine this context with sensor-derived features where appropriate. The LLM can then interpret diagnostic and prognostic outputs in context and suggest maintenance options for engineer review. Existing studies already demonstrate elements of this complementary paradigm [39–43]. For example, conventional diagnostic models can identify faults, while LLMs provide knowledge-grounded explanations and maintenance recommendations; domain-adapted LLMs may also participate more directly in fault classification when sufficient task-specific information is available. The appropriate division of roles should therefore depend on the type of available data, the required accuracy, the computational constraints, and the level of engineering validation.

A further opportunity is to extend fault recognition toward predictive and prescriptive maintenance. For example, an estimated RUL can be interpreted alongside machine loading, previous maintenance actions, known failure mechanisms, and operational requirements to inform decisions on inspection, repair, or replacement. In this sense, ML/DL and prognostic models primarily predict what is happening or likely to happen, while LLMs can help interpret the evidence and reason about what engineering action should follow.

Human expertise nevertheless remains essential, particularly because maintenance records and historical engineering information may be incomplete, inconsistent, or incorrectly interpreted by an LLM. As indicated by the feedback paths in Figure 5, engineers can

validate and correct extracted information, refine structured features and labels, evaluate diagnostic and prognostic results, and provide new operational and maintenance outcomes for subsequent model updating. Such a human-in-the-loop framework could progressively convert accumulated operational experience into reusable digital engineering knowledge, enabling condition monitoring to evolve from predominantly sensor-based fault recognition toward knowledge-enhanced diagnosis, prognosis, and maintenance decision support.

3.5. Summary of Complementary Roles of LLMs and Computational Methods

Based on the discussions in Sections 3.1–3.4, Table 2 summarizes the complementary roles of LLMs and established computational methods for electrical machines. These roles will provide complementary capabilities within a hierarchical intelligence framework. Physics-based models provide physical fidelity and verification; ML/DL models support efficient prediction, estimation, classification, and surrogate modeling; and numerical methods perform quantitative simulation, optimization, and control. LLMs can add a complementary layer of engineering intelligence through knowledge integration, reasoning, tool interaction, and interpretation of results. This support can help engineers determine what to model or optimize, coordinate the use of computational tools, and interpret the resulting designs and trade-offs. Together, these capabilities can support more knowledge-guided and integrated engineering workflows.

Table 2. Complementary Roles of LLMs and Established Computational Methods in Electrical Machine Engineering.

Application Area	Established Computational Intelligence	Primary Role of Established Methods	Role of LLMs	LLM-Enabled Engineering Intelligence
Electromagnetic and Multiphysics Modeling	FEA/multiphysics solvers; ML/DL surrogates; PINNs and reduced-order models	Field computation, multiphysics analysis, fast prediction, and high-fidelity verification	Assist with model setup and tool coordination before simulation. Interpret field and performance outputs after simulation.	Modeling intelligence and coordination
Machine Design and Optimization	FEA/multiphysics models; ML/DL surrogates; DOE and numerical optimization algorithms	Performance evaluation, design-space search, uncertainty quantification, and optimization	Help formulate design variables and uncertainties before the search. Coordinate evaluations during the search and interpret trade-offs for design review.	Design intelligence around optimization
Drive Control and Intelligent Operation	Conventional controllers; MPC; ML/DL/RL models; state estimators	Fast real-time control, estimation, prediction, and adaptation	Controller assistance and tuning; strategy selection; supervisory adaptation; system-level coordination	Supervisory intelligence above real-time control

Table 2. *Cont.*

Application Area	Established Computational Intelligence	Primary Role of Established Methods	Role of LLMs	LLM-Enabled Engineering Intelligence
Condition Monitoring, Fault Diagnosis, and Predictive Maintenance	Signal processing; physics-based models; ML/DL diagnostic and prognostic models	Anomaly detection, fault classification, health estimation, and RUL prediction	Structure maintenance context for diagnosis. Interpret diagnostic and prognostic outputs to support maintenance decisions.	Diagnostic reasoning and decision intelligence

4. Key Challenges

Despite the opportunities discussed in Section 3, the transition from proof-of-concept demonstrations to reliable engineering deployment of LLMs remains challenging. Complex physical laws govern the operation of electrical machines. Consequently, the usefulness of LLMs in this field cannot be evaluated solely by their ability to generate good responses or automate individual engineering tasks. Instead, their integration into practical engineering workflows raises four fundamental questions:

- Is the generated information technically correct?
- Does the model sufficiently understand the engineering domain?
- Can it interact reliably and efficiently with real engineering tools and systems?
- Can its decisions be trusted and safely deployed?

These questions lead to four major challenges, as discussed below.

4.1. Engineering Reliability and Physics Grounding—Is It Correct?

The first and most fundamental question is: Is the output generated or recommended by an LLM technically and physically correct? Currently, general-purpose LLMs are primarily trained to predict and generate information from large-scale data rather than to satisfy physical laws, numerical constraints, or engineering design rules. Consequently, an LLM may produce responses that appear technically convincing while containing inappropriate modeling assumptions, infeasible design parameters, or unsupported engineering conclusions. For example, a generated FEA model may be syntactically executable but contain inappropriate boundary conditions or material properties. Related FEA studies have tested retrieval, domain fine-tuning, and feedback from numerical tools. AutoFEA used retrieval to support input-file generation. In a beam example, the LLM corrected its displacement estimate after receiving results from a model that accounted for plastic deformation [44]. ALL-FEM combined domain fine-tuning with execution feedback and checked the generated code and numerical results against expert references. In one benchmark, a visually plausible result came from code that used plane strain in place of the specified plane stress [45]. Together with the boundary assignment error discussed in Section 3.1 [46], these examples motivate checks of both the physical formulation and the numerical results. For electrical machines, engineers can review material, winding, excitation, and boundary assignments and compare key outputs with an independently configured reference model or measurements.

4.2. Domain Knowledge, Data, and Multimodal Reasoning—Does It Understand the Engineering Domain?

The second question is: Does an LLM sufficiently understand electrical machine engineering and the information through which this knowledge is represented? Electrical machines constitute a highly specialized multidisciplinary domain involving electromagnetics, materials, thermal management, structural mechanics, power electronics, control, and manufacturing. Moreover, knowledge of electrical machines is not primarily represented in natural language. Engineering information is distributed across various forms, such as equations, material databases, CAD models, control diagrams, simulations, and experimental measurements. Understanding the relationships among these modalities is considerably more demanding than conventional text–image interpretation. An additional difficulty is the availability of suitable training and validation data, since high-quality industrial machine-design, manufacturing, operating, and failure data are often limited, proprietary, or unevenly distributed. Building reliable domain-specific and multimodal engineering datasets, therefore, remains an important prerequisite for more capable LLM-based systems.

4.3. Tool Integration, Real-Time Implementation, and Reliability—Can It Work Reliably with Real Engineering Systems?

The third question is: Can an LLM operate reliably, efficiently, and repeatedly within real electrical-machine engineering workflows? As shown in the literature review, the principal value of LLMs at this stage may lie in their ability to serve as an intelligent coordination layer connecting human engineers with specialized computational and experimental tools. However, this capability requires substantially more than generating engineering text or source code. A practical LLM-based agent may need to communicate with FEA and multiphysics solvers, CAD environments, MATLAB or Python programs, optimization algorithms, ML/DL models, databases, control platforms, laboratory equipment, and other engineering software through various interfaces.

At each tool handoff, inputs should be checked for format, units, and consistency with the current model. Returned files should be checked for completeness and linked to the corresponding run before their results are used downstream. A transient failure, such as a timeout, may justify a limited retry after checking the status of the previous run. Persistent failures or results that fail engineering checks should pause the workflow for engineer review.

For reproducibility, run records should capture the prompts, retrieved sources where used, model and API versions where available, and sampling settings. They should also preserve code and data versions, solver settings, outputs, and any manual corrections. When the exact version of a hosted model is unavailable, the provider and access date can be recorded. These records allow repeated runs to be assessed against the same engineering acceptance criteria.

4.4. Safety, Security, Explainability, and Trust—Can We Trust and Safely Deploy It?

The fourth question is: Can engineers trust LLM-enabled systems sufficiently to deploy them in critical applications? This question becomes increasingly important as LLMs move from passive assistants toward active agents capable of modifying simulation models, changing optimization settings, recommending controller parameters, interpreting faults, or proposing maintenance actions. Electrical drives are widely used in critical applications where an incorrect engineering decision can lead to equipment damage, operational interruptions, or safety risks. Therefore, increasing LLM autonomy must be accompanied by corresponding mechanisms for verification, constraint enforcement, and human oversight. Explainability and traceability are central to this requirement. Engineers should be

able to determine what information was used, which models or tools were invoked, what assumptions were made, and why a particular recommendation was produced. This is especially important when an LLM combines retrieved knowledge, simulation outputs, ML/DL predictions, and historical data within a multistep reasoning process. Engineering decisions should therefore be accompanied by traceable evidence rather than unsupported natural-language explanations. These challenges can compound across a workflow. For example, passing a specified RMS phase current to a solver as a peak value can underestimate winding losses. If the resulting data are used to train a surrogate, the bias can carry through to candidate screening and design selection. Recording the current convention at each tool handoff and re-evaluating shortlisted designs at the specified excitation can help detect the error before a design is accepted.

5. Future Directions and Research Roadmap

The challenges discussed in Section 4 are cross-cutting barriers that affect multiple LLM-enabled engineering applications rather than corresponding individually to specific research directions. Addressing these challenges will require advances in physics grounding and engineering reliability, domain knowledge and multimodal understanding, tool integration and reliability, and safety and trustworthiness. Against this background, three broader research directions are particularly important for the future development of LLM-enabled electrical-machine engineering: digitalized engineering knowledge, LLM-enabled system-level design and optimization, and autonomous engineering agents.

5.1. Digitalized Engineering Knowledge for Electrical Machines

A fundamental future direction for LLM-enabled electrical-machine engineering is the digitalization and computational utilization of engineering knowledge. Valuable knowledge is widely distributed across textbooks, the scientific literature, standards, historical designs, simulation and experimental results, manufacturing and quality records, maintenance reports, fault logs, and expert experience. Much of this information is heterogeneous and unstructured and therefore cannot be directly exploited by conventional computational models. LLMs, particularly when combined with multimodal understanding, RAG, and knowledge graphs, provide a promising means for extracting, integrating, and transforming such information into structured, machine-accessible engineering knowledge.

In the longer term, such capabilities may enable digital electrical-machine knowledge bases integrating structured databases, knowledge graphs, multimodal engineering information, and validated domain knowledge accumulated throughout design, manufacturing, operation, and maintenance. This knowledge infrastructure should be physically grounded, traceable to its original sources, and continuously validated using physics-based models, experimental evidence, and human expertise. Rather than simply developing domain-specific LLMs, the broader objective is to establish a knowledge-grounded engineering intelligence infrastructure that supports future intelligent digital twins, advanced design and optimization, and progressively autonomous electrical-machine engineering.

5.2. LLM-Enabled System-Level Design Optimization of Electrical Drive Systems

A particularly promising direction is extending LLM-enabled optimization from individual electrical machines to system-level design optimization of complete electrical drive systems. Electrical drive system design is inherently a multilevel and multidisciplinary process involving system requirements, machine topology, inverter and control design, material and dimensional selection, electromagnetic, mechanical, and thermal performance, and manufacturing considerations [80–82]. Figure 6 illustrates a framework of LLM-enabled system-level design optimization of electrical drive systems. The left-hand side of the

figure shows a typical system-level design and optimization flowchart for electrical drive systems [3,5]. On the right-hand side, some opportunities are listed for LLM-enabled multilevel design and optimization. Black arrows indicate the design workflow and interactions among design tasks; blue connectors link the design levels to the corresponding opportunities for LLM assistance.

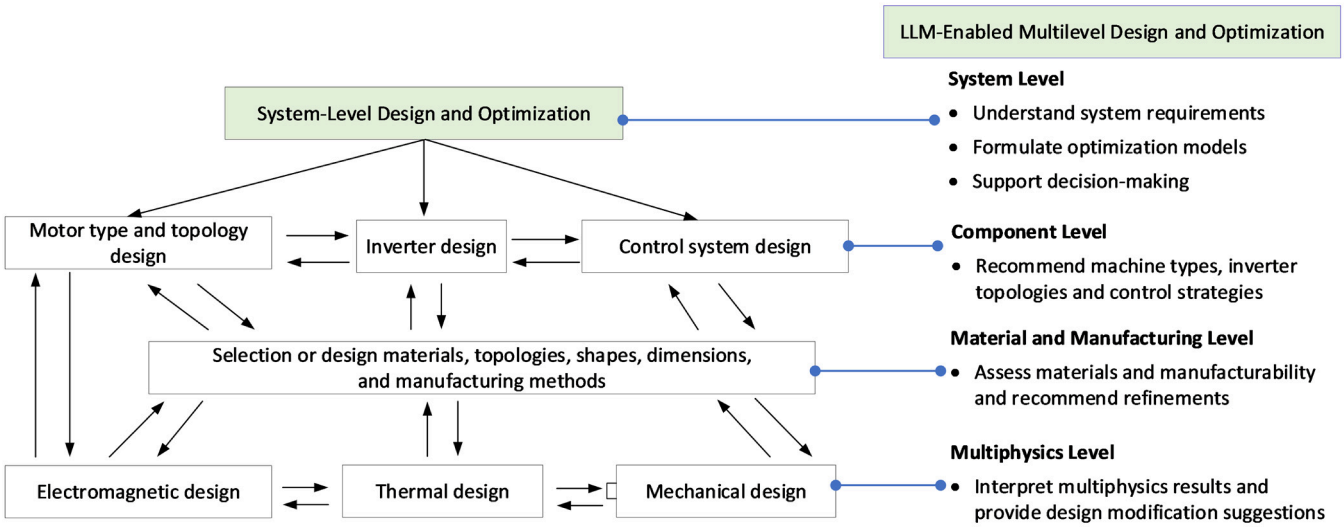


Figure 6. A framework of LLM-enabled system-level design optimization of electrical drive systems.

LLMs could provide knowledge integration and reasoning support throughout this process. At the system level, they could interpret application requirements, analyze trade-offs, and support the selection and coordination of machines, inverters, and control systems. At the detailed design level, engineering knowledge from previous designs, standards, materials, manufacturing records, and expert experience could support material and topology selection, definition of design variables, and manufacturability assessment. LLMs could also interpret electromagnetic, thermal, and mechanical simulation results and use multidisciplinary information to recommend design modifications, as discussed in Section 3.1. Importantly, this framework could integrate parameterized, robust, and topology optimization rather than treating them as separate design paradigms.

5.3. Towards Autonomous Electrical-Machine Engineering Agents

The longer-term evolution of LLM-enabled electrical machine engineering may lead to autonomous engineering agents. As illustrated in Figure 7, future agents could integrate engineering tasks and requirements with knowledge from the literature, standards, historical designs, simulations, experiments, operational data, and engineering experience. Rather than serving only as assistants for individual tasks, LLM-based agents could progressively understand engineering requirements, reason about possible solutions, formulate plans, select and use appropriate tools, analyze results, and support decisions across design and optimization, analysis and control, operation, diagnosis, and maintenance. In this sense, the digitized engineering knowledge discussed in Section 5.1 could provide the foundational knowledge, while the multilevel design and optimization framework in Section 5.2 represents an important application scenario for increasingly autonomous engineering.

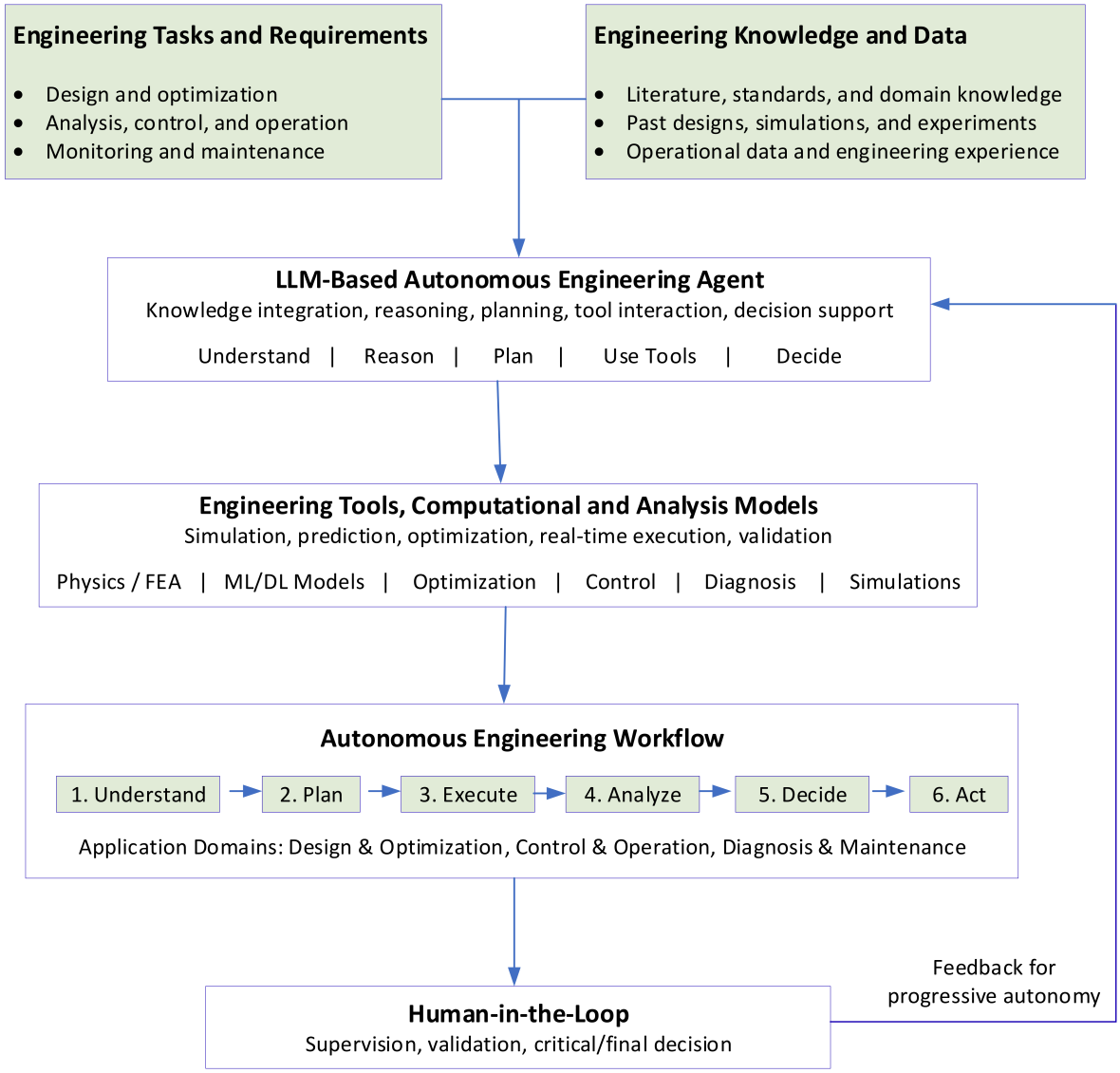


Figure 7. A framework of LLM-enabled autonomous electrical machine engineering agents.

Such autonomy should be achieved through the coordination of specialized engineering models and tools rather than through an LLM alone. The LLM-based agent can connect these capabilities through an iterative "Understand–Plan–Execute–Analyze–Decide–Act" workflow, selecting appropriate models and tools, interpreting their outputs, managing iterative processes, and adapting subsequent actions based on intermediate results. This architecture could progressively integrate currently separated modeling, simulation, optimization, control, diagnosis, and maintenance activities into more coordinated engineering workflows. Problem formulation can be evaluated by checking whether the objectives, constraints, and modeling assumptions match an engineer-reviewed specification. Workflow coordination can be evaluated by the proportion of tasks completed with verified outputs, the interventions required, and recovery from tool failures. Decision support can be evaluated by whether recommendations satisfy the stated constraints and are supported by verified evidence. Comparisons intended to assess the contribution of LLM assistance should use the same engineering tasks and underlying computational tools.

An agent may initially propose actions for engineer approval and later execute validated steps within a defined scope. Delegation should expand when repeated evaluations meet task-specific acceptance criteria and show that failures can be handled or referred to an

engineer. Engineers continue to define objectives and constraints, validate critical results, resolve ambiguous evidence, and approve decisions with significant consequences.

6. Conclusions

This paper reviewed emerging LLM applications in electrical machines and discussed opportunities in electromagnetic and multiphysics modeling, machine design and optimization, drive control and intelligent operation, and condition monitoring, fault diagnosis, and predictive maintenance. Early studies illustrate how LLMs can assist with design-space exploration, controller tuning, and the interpretation of diagnostic results. Related work in computational engineering also suggests ways to connect LLMs with modeling and simulation tools. In these workflows, LLMs can help bring engineering knowledge, computational methods, and human expertise together. Established solvers, optimizers, and validated controllers provide the capabilities needed for numerical evaluation, design search, and real-time control. Applying these workflows to new machines and operating conditions will require further engineering validation.

Future research should explore the digitalization of heterogeneous engineering knowledge, the integration of LLMs with established models and numerical methods for system-level design and optimization, and the development of progressively autonomous engineering agents. As these capabilities mature, LLMs may help coordinate increasingly complex workflows across modeling, design, operation, diagnosis, and maintenance. The longer-term vision is an integrated engineering environment that connects knowledge, physics, computational tools, and human expertise, with engineers guiding critical checks and decisions.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/wevj17120628/s1. The supporting information accompanying this article includes finite-element simulation data for the baseline model, 256 full-factorial designs, and five candidate re-evaluations; Ansys Maxwell scripts; Gaussian process and NSGA-II analysis code; and a README, software requirements, and principal prompts and workflow instructions.

Author Contributions: Conceptualization, G.L.; methodology, H.J. and G.L.; investigation, Z.D. and H.J.; data curation, Z.D. and H.J.; writing—original draft preparation, G.L., Z.D. and H.J.; writing—review and editing, H.J., G.L., Y.G. and J.Z.; supervision, G.L., Y.G. and J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The simulation data and code supporting the case study in Section 3.2.4 are provided in the Supplementary Materials. These materials include the baseline comparison with published measurements, the inputs and results for the 256 full-factorial Maxwell simulations and the five candidate re-evaluations, the Maxwell scripts, and the code and settings for Gaussian process modeling and NSGA-II optimization. The accompanying README describes the software environment and the steps for reproducing the analysis. Further inquiries can be directed to the corresponding author.

Acknowledgments: During manuscript preparation and revision, the authors used OpenAI's ChatGPT (recorded as versions 5.6 and 6.1) to assist with writing and language editing. For the case study in Section 3.2.4, the authors used OpenAI's Codex desktop application to assist with extracting published modeling inputs, generating and debugging scripts, and analyzing simulation results. The model designation recorded for baseline reconstruction was "ChatGPT-6.1 Sol." The authors reviewed and edited the AI-assisted output and take full responsibility for the content of this publication.

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

References

1. Liu, C.; Chau, K.T.; Lee, C.H.T.; Song, Z. A critical review of advanced electric machines and control strategies for electric vehicles. *Proc. IEEE* **2021**, *109*, 1004–1028. [CrossRef]
2. Husain, I.; Ozpineci, B.; Islam, S.; Gurpinar, E.; Su, G.-J.; Yu, W.; Chowdhury, S.; Xue, L.; Rahman, D.; Sahu, R. Electric drive technology trends, challenges, and opportunities for future electric vehicles. *Proc. IEEE* **2021**, *109*, 1039–1059. [CrossRef]
3. Jiang, H.; Wang, Y.; Lei, G.; Sun, X.; Zhu, J. Design optimization of EV drive systems: Building the next generation of automatic AI platforms. *World Electr. Veh. J.* **2026**, *17*, 35. [CrossRef]
4. Bramerdorfer, G.; Tapia, J.A.; Pyrhönen, J.J.; Cavagnino, A. Modern electrical machine design optimization: Techniques, trends, and best practices. *IEEE Trans. Ind. Electron.* **2018**, *65*, 7672–7684. [CrossRef]
5. Lei, G.; Zhu, J.; Guo, Y. *Multidisciplinary Design Optimization Methods for Electrical Machines and Drive Systems*; Springer: Berlin/Heidelberg, Germany, 2016.
6. Lei, G.; Zhu, J.; Guo, Y. A review of design optimization methods for electrical machines. *Energies* **2017**, *10*, 1962. [CrossRef]
7. Zhu, Y.; Guo, Z.; Lei, G.; Guo, Y.; Zhu, J. Self-adaptive physics-informed neural networks for solving 2-D magnetostatic fields in open boundaries. *IEEE Trans. Magn.* **2026**, *62*, 7202906. [CrossRef]
8. Gong, Z.; Chu, Y.; Yang, S. Physics-informed neural networks for solving 2-D magnetostatic fields. *IEEE Trans. Magn.* **2023**, *59*, 7002005. [CrossRef]
9. Khan, A.; Lowther, D.A. Physics-informed neural networks for electromagnetic analysis. *IEEE Trans. Magn.* **2022**, *58*, 7500404. [CrossRef]
10. Beltrán-Pulido, A.; Aliprantis, D.; Bilionis, I.; Munoz, A.R.; Chase, N. Physics-informed neural networks for parametric modeling of permanent magnet synchronous machines. *IEEE Trans. Energy Convers.* **2026**, *41*, 648–661. [CrossRef]
11. Wu, T.; Dai, P.; Xue, G.; Guo, Y.; Lei, G.; Zhu, J.; Wang, Y. A novel thermal analysis method for tubular PM linear motors based on transfer learning. *IEEE Trans. Transp. Electrif.* **2025**, *11*, 7379–7388. [CrossRef]
12. Kirchgässner, W.; Wallscheid, O.; Böcker, J. Data-driven permanent magnet temperature estimation in synchronous motors with supervised machine learning: A benchmark. *IEEE Trans. Energy Convers.* **2021**, *36*, 2059–2067. [CrossRef]
13. Khan, A.; Mohammadi, M.H.; Ghorbanian, V.; Lowther, D. Efficiency map prediction of motor drives using deep learning. *IEEE Trans. Magn.* **2020**, *56*, 7511504. [CrossRef]
14. Cheng, M.; Zhao, X.; Dhimish, M.; Qiu, W.; Niu, S. A review of data-driven surrogate models for design optimization of electric motors. *IEEE Trans. Transp. Electrif.* **2024**, *10*, 8413–8431. [CrossRef]
15. Qiu, W.; Zhao, X.; Tyrrell, A.; Perinpanayagam, S.; Niu, S.; Wen, G. Application of artificial intelligence-based technique in electric motors: A review. *IEEE Trans. Power Electron.* **2024**, *39*, 13543–13568. [CrossRef]
16. Asef, P.; Vagg, C. A physics-informed Bayesian optimization method for rapid development of electrical machines. *Sci. Rep.* **2024**, *14*, 4526. [CrossRef] [PubMed]
17. Lei, G.; Liu, C.; Zhu, J.; Guo, Y. Techniques for multilevel design optimization of permanent magnet motors. *IEEE Trans. Energy Convers.* **2015**, *30*, 1574–1584. [CrossRef]
18. Sun, X.; Shi, Z.; Lei, G.; Guo, Y.; Zhu, J. Multi-objective design optimization of an IPMSM based on multilevel strategy. *IEEE Trans. Ind. Electron.* **2021**, *68*, 139–148. [CrossRef]
19. Shi, Z.; Sun, X.; Yang, Z.; Cai, Y.; Lei, G.; Zhu, J.; Lee, C.H.T. Design optimization of a spoke-type axial-flux PM machine for in-wheel drive operation. *IEEE Trans. Transp. Electrif.* **2024**, *10*, 3770–3781. [CrossRef]
20. Farah, N.; Lei, G.; Bramerdorfer, G.; Zhu, J. A review of machine-learning-based control methods for permanent magnet synchronous machines. *IET Electr. Power Appl.* **2025**, e70126. [CrossRef]
21. Szoke, E.; Szabo, C.; Pintilie, L.-N. Artificial intelligence-based sensorless control of induction motors with dual-field orientation. *Appl. Sci.* **2025**, *15*, 8919. [CrossRef]
22. Jakobeit, D.; Schenke, M.; Wallscheid, O. Universal direct torque controller for permanent magnet synchronous motors via meta-reinforcement learning. *IEEE Trans. Power Electron.* **2026**, *41*, 7394–7406. [CrossRef]
23. Zhang, S.; Wallscheid, O.; Porrmann, M. Machine learning for the control and monitoring of electric machine drives: Advances and trends. *IEEE Open J. Ind. Appl.* **2023**, *4*, 188–214. [CrossRef]
24. Zhou, X.; Xu, Y.; Zhao, J.; Zhang, R. Large language model applications in power systems: A comprehensive review and outlook. *J. Mod. Power Syst. Clean Energy* **2026**, *14*, 773–790. [CrossRef]
25. Liang, M.; Luo, Q.; Yu, T.; Kuang, P.; Li, Z.; Pan, Z. LLM-based exploitation of edge data in modern power systems. *J. Mod. Power Syst. Clean Energy* **2026**, *14*, 63–81. [CrossRef]
26. Bernier, F.; Cao, J.; Cordy, M.; Ghamizi, S. PowerGraph-LLM: Novel power grid graph embedding and optimization with large language models. *IEEE Trans. Power Syst.* **2025**, *40*, 5483–5486. [CrossRef]
27. Naeem, Z.; Cirrincione, G.; Favuzza, S.; Musca, R.; Sanseverino, E.R.; Sciume, G.; Zizzo, G. Large language models in power systems: From grid operations to home energy management. *Energies* **2026**, *19*, 3769. [CrossRef]

28. Deng, K.; Zhou, Y.; Zeng, H.; Wang, Z.; Guo, Q. Power grid model generation based on the tool-augmented large language model. *IEEE Trans. Power Syst.* **2025**, *40*, 5487–5490. [CrossRef]
29. Miao, X.; Jia, M.; Yang, B.; Guan, X. AutoHPFL: Autonomous Physics-Data Hybrid Power Flow Linearization Design via LLM Agents. *IEEE Trans. Power Syst.* **2026**, *41*, 4867–4870. [CrossRef]
30. Li, B.; Zhang, H.; Cao, Y. Adaptive solving method for power system operation based on knowledge-driven LLM agents. *Electronics* **2026**, *15*, 478. [CrossRef]
31. Sekiyama, W.; Sasaki, H.; Sakamoto, H.; Takahashi, S. Parameter and topology optimization of IPMSMs with dimensionality reduction by multimodal large language models. *IEEE Trans. Magn.* **2026**, *in press*. [CrossRef]
32. Guesmia, M.A.; Pham, C.; Pan, Y.-J.; Nguyen, K.K.; Al-Haddad, K.; Wang, Q. AI-assisted Bayesian optimization of a permanent magnet synchronous motor for e-bike applications. *Machines* **2026**, *14*, 160. [CrossRef]
33. Hetrich, M.; Sahu, C.K.; Narasimhan, S.; Chilukuri, P.K. Power Circuit AI: Designing power electronic circuits for motor drives with generative artificial intelligence. In Proceedings of the 2026 IEEE International Conference on Intelligent Design and Control of Automation and Drive Systems (IDCD), Chicago, IL, USA, 23–24 April 2026; pp. 1–7.
34. Gashi, K.; Chacin, M.; Tang, G. Electromechanical design and control using a large language model. In Proceedings of the 2024 12th International Conference on Control, Mechatronics and Automation (ICCMA), London, UK, 11–13 November 2024; pp. 354–360.
35. Tarczewski, T.; Stojic, D.; Dzielinski, A. Large language model-based tuning assistant for variable speed PMSM drive with cascade control structure. *Electronics* **2025**, *14*, 232. [CrossRef]
36. Arif, T.M.; Rahim, M.A. Agentic AI for real-time adaptive PID control of a servo motor. *Actuators* **2025**, *14*, 459. [CrossRef]
37. Urquizo, J.; Pasmay, P.; Muñoz, L.; Galarza, L. Model-based and large language model meta artificial intelligence techniques for intelligent permanent magnet synchronous motor drive control. *IFAC J. Syst. Control* **2025**, *34*, 100341. [CrossRef]
38. Ye, S.; Liu, G.-P. On Generating PID Control Diagrams by Large Language Model-Based Multiple Agents. In Proceedings of the 2025 China Automation Congress (CAC), Harbin, China, 26–28 September 2025; pp. 3818–3823. [CrossRef]
39. Miljković, D. Large language models for automated motor current signature analysis. In Proceedings of the 2026 49th MIPRO ICT and Electronics Convention (MIPRO), Opatija, Croatia, 25–29 May 2026; pp. 958–963.
40. Santos, L.F.D.D.; Canuto, J.L.D.S.; Aylon, L.B.R.; Mariani, V.C.; de Souza, R.C.T. A multimodal model for fault diagnosis in induction motors using thermal images and large language models. *IEEE Access* **2026**, *14*, 89964–89976. [CrossRef]
41. Huang, K.; Chen, Q. Motor fault diagnosis and predictive maintenance based on a fine-tuned Qwen2.5-7B model. *Processes* **2025**, *13*, 2051. [CrossRef]
42. Lee, J.; Rew, J. Multi-agent large language model-based decision tree analysis for explainable electric vehicle drive motor fault diagnosis. *Comput. Mater. Contin.* **2026**, *87*, 100. [CrossRef]
43. Wu, S.; Wang, X.; Liu, Z.; Shi, G. Research on motor diagnosis and maintenance decision-making technology based on DCCNN-L transformer and fault knowledge graph. *Sci. Rep.* **2026**, *16*, 24220. [CrossRef] [PubMed]
44. Hou, S.; Johnson, R.; Makhija, R.; Chen, L.; Ye, Y. AutoFEA: Enhancing AI Copilot by integrating finite element analysis using large language models with graph neural networks. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 24078–24085. [CrossRef]
45. Deotale, R.; Srinivasan, A.; Golestanian, M.; Tian, Y.; Zhang, T.; Vlachos, P.; Gomez, H. ALL-FEM: Agentic large language models fine-tuned for finite element methods. *Comput. Methods Appl. Mech. Eng.* **2026**, *457*, 118985. [CrossRef]
46. Zhang, N.; Wang, J. COMSOL-MCP: An open-source model context protocol interface for AI-assisted multiphysics simulation. *Neurocomputing* **2026**, *703*, 134481. [CrossRef]
47. Sun, C.; Fang, Y.; Pfister, P.-D. Comparison of two finite-permeability subdomain models for surface-mounted permanent-magnet machines considering magnetic saturation. *IEEE Trans. Energy Convers.* **2024**, *39*, 2758–2768. [CrossRef]
48. Chen, M.; Li, H.; Wang, S.; Guillod, T.; Serrano, D.; Förster, N.; Kirchgässner, W.; Piepenbrock, T.; Schweins, O.; Wallscheid, O.; et al. MagNet Challenge for data-driven power magnetics modeling. *IEEE Open J. Power Electron.* **2025**, *6*, 883–898. [CrossRef]
49. Xiao, Y.; Li, C.; Zheng, Z. A magnetic core loss model based on physics-informed neural network with cross-attention. *IEEE Trans. Power Electron.* **2026**, *41*, 92–96. [CrossRef]
50. Huang, Q.; Li, Y.; Zhu, J.; Li, S. Magnetization mechanism-inspired neural networks for core loss estimation. *IEEE Trans. Power Electron.* **2024**, *39*, 16382–16390. [CrossRef]
51. Hu, Y.; Lv, Y.; Lu, H.; Gao, J.; Wang, J.; Xu, W. Physics-Aware Vision Transformer for Steady-State Ferrite Core Temperature Estimation via Time-Density Hysteresis Imaging. *IEEE Trans. Power Electron.* **2026**, *42*, 613–625. [CrossRef]
52. Xiang, D.; Zhu, Z.Q.; Liang, D.; Xu, F.; He, T. Taguchi-based robust design for minimising torque ripple in 6-slot/2-pole modular high-speed permanent magnet motor with manufacturing tolerances. *IET Electr. Power Appl.* **2024**, *18*, 1435–1447. [CrossRef]
53. Lei, G.; Bramerdorfer, G.; Liu, C.; Guo, Y.; Zhu, J. Robust design optimization of electrical machines: A comparative study and space reduction strategy. *IEEE Trans. Energy Convers.* **2021**, *36*, 300–313. [CrossRef]
54. Lei, G.; Bramerdorfer, G.; Ma, B.; Guo, Y.; Zhu, J. Robust design optimization of electrical machines: Multi-objective approach. *IEEE Trans. Energy Convers.* **2021**, *36*, 390–401. [CrossRef]

55. Song, W.S.; Min, S. Robust topology optimization of interior permanent magnet synchronous motor for torque ripple reduction under current uncertainty. *Appl. Math. Model.* **2025**, *140*, 115917. [CrossRef]
56. Guidotti, G.; Barri, D.; Soresini, F.; Gobbi, M. Optimal design of interior permanent magnet synchronous motor considering various sources of uncertainty. *World Electr. Veh. J.* **2025**, *16*, 79. [CrossRef]
57. Mahmouditabar, F.; Vahedi, A.; Takorabet, N. Robust design of BLDC motor considering driving cycle. *IEEE Trans. Transp. Electrif.* **2024**, *10*, 1414–1424. [CrossRef]
58. Wu, J.; Zhu, X.; Xiang, Z.; Fan, D.; Quan, L.; Xu, L. Robust optimization of a rare-earth-reduced high-torque-density PM motor for electric vehicles based on parameter sensitivity region. *IEEE Trans. Veh. Technol.* **2022**, *71*, 10269–10279. [CrossRef]
59. Xu, L.; Wu, W.; Zhao, W.; Liu, G.; Niu, S. Robust design and optimization for a permanent magnet vernier machine with hybrid stator. *IEEE Trans. Energy Convers.* **2020**, *35*, 2086–2094. [CrossRef]
60. Bramerdorfer, G. Tolerance analysis for electric machine design optimization: Classification, modeling and evaluation, and example. *IEEE Trans. Magn.* **2019**, *55*, 8106809. [CrossRef]
61. Yang, Y.; Bianchi, N.; Bramerdorfer, G.; Zhang, C.; Zhang, S. Methods to improve the cogging torque robustness under manufacturing tolerances for the permanent magnet synchronous machine. *IEEE Trans. Energy Convers.* **2021**, *36*, 2152–2162. [CrossRef]
62. Bramerdorfer, G. Effect of the manufacturing impact on the optimal electric machine design and performance. *IEEE Trans. Energy Convers.* **2020**, *35*, 1935–1943. [CrossRef]
63. Ma, B.; Zheng, J.; Lei, G.; Zhu, J.; Jin, P.; Guo, Y. Topology optimization of ferromagnetic components in electrical machines. *IEEE Trans. Energy Convers.* **2020**, *35*, 786–798. [CrossRef]
64. Lee, C.; Lee, J.; Jang, I.G. Topology optimization for the manufacturable and structurally safe synchronous reluctance motors with multiple iron webs and bridges. *IEEE Trans. Ind. Electron.* **2023**, *70*, 678–687. [CrossRef]
65. Sasaki, H.; Igarashi, H. Topology optimization accelerated by deep learning. *IEEE Trans. Magn.* **2019**, *55*, 7401305. [CrossRef]
66. Sasaki, H.; Yamamura, K. Topology optimization with Shapley additive explanations for permanent magnet synchronous motors. *IEEE Trans. Magn.* **2024**, *60*, 8201304. [CrossRef]
67. Asanuma, J.; Doi, S.; Igarashi, H. Transfer learning through deep learning: Application to topology optimization of electric motor. *IEEE Trans. Magn.* **2020**, *56*, 7512404. [CrossRef]
68. Khan, A.; Midha, C.; Lowther, D. Reinforcement learning for topology optimization of a synchronous reluctance motor. *IEEE Trans. Magn.* **2022**, *58*, 8206204. [CrossRef]
69. Hidaka, Y. Multi-material topology optimization method for hybrid-field motors utilizing design guidelines with explainable AI. *IEEE Trans. Magn.* **2026**, *in press*. [CrossRef]
70. Sato, T. A data-driven parameter-topology hybrid optimization of permanent magnet motors based on deep reinforcement learning. *IEEE Trans. Magn.* **2026**, *62*, 8101706. [CrossRef]
71. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
72. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
73. Rodriguez, J.; Garcia, C.; Mora, A.; Flores-Bahamonde, F.; Acuna, P.; Novak, M.; Zhang, Y.; Tarisciotti, L.; Davari, S.A.; Zhang, Z.; et al. Latest advances of model predictive control in electrical drives—Part I: Basic concepts and advanced strategies. *IEEE Trans. Power Electron.* **2022**, *37*, 3927–3942. [CrossRef]
74. Rodriguez, J.; Garcia, C.; Mora, A.; Davari, A.; Rodas, J.; Garcia, D.F.V.; Elmorshedy, M.F.; Wang, F.; Zuo, K.; Tarisciotti, L.; et al. Latest advances of model predictive control in electrical drives—Part II: Applications and benchmarking with classical control methods. *IEEE Trans. Power Electron.* **2022**, *37*, 5047–5061.
75. Zhang, Y.; Jin, J.; Huang, L. Model-free predictive current control of PMSM drives based on extended state observer using ultralocal model. *IEEE Trans. Ind. Electron.* **2021**, *68*, 993–1003. [CrossRef]
76. De Klerk, M.L.; Saha, A.K. A comprehensive review of advanced traction motor control techniques suitable for electric vehicle applications. *IEEE Access* **2021**, *9*, 125080–125108. [CrossRef]
77. Wang, B.; Jin, S.; Sun, P.; Liu, G. Comprehensive review of finite-control-set model predictive control methods for permanent-magnet synchronous motors. *Chin. J. Electr. Eng.* **2026**, *12*, 240–259. [CrossRef]
78. Nandi, S.; Toliyat, H.A.; Li, X. Condition monitoring and fault diagnosis of electrical motors—A review. *IEEE Trans. Energy Convers.* **2005**, *20*, 719–729. [CrossRef]
79. Shao, S.; McAleer, S.; Yan, R.; Baldi, P. Highly accurate machine fault diagnosis using deep transfer learning. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2446–2455. [CrossRef]
80. Bramerdorfer, G.; Lei, G.; Cavagnino, A.; Zhang, Y.; Sykulski, J.; Lowther, D.A. More robust and reliable optimized energy conversion facilitated through electric machines, power electronics and drives, and their control: State-of-the-art and trends. *IEEE Trans. Energy Convers.* **2020**, *35*, 1997–2012. [CrossRef]

81. Lei, G.; Wang, T.; Zhu, J.; Guo, Y.; Wang, S. System-level design optimization method for electrical drive systems—Robust approach. *IEEE Trans. Ind. Electron.* **2015**, *62*, 4702–4713. [CrossRef]
82. Lei, G.; Wang, T.; Guo, Y.; Zhu, J.; Wang, S. System-level design optimization methods for electrical drive systems: Deterministic approach. *IEEE Trans. Ind. Electron.* **2014**, *61*, 6591–6602. [CrossRef]

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