World Journal of Educational Studies
Print ISSN: 2959-9989
Online ISSN: 2959-9997
DOI: https://doi.org/10.61784/wjes3132

DUAL-ENGINE DRIVE OF DATA + MODEL ON OPTIMIZATION THEORY FOR COURSE FOR THE AI ERA

DanJu Lv*, YueYun Yu, Yan Zhang
College of Big Data and Intelligent Engineering, Southwest Forestry University, Kunming 650024, Yunnan, China.
**Corresponding Author: DanJu Lv*

Abstract: Addressing the shift in AI-era optimization algorithms from offline static to integrated perception, this paper proposes a three-tiered pyramid teaching framework. By introducing computational graph perspectives to reshape foundational theory, empowering advanced training through dual-layer programming, and driving end-to-end differentiable optimization applications drives application practice, effectively resolving the challenges of fragmented knowledge, outdated tools, and single-dimensional assessment in traditional curricula. Practice demonstrates that this framework significantly enhances graduate students' dual-habitat cross-disciplinary problem-solving capabilities, bridging mathematical rigor and AI intuition.
Keywords: Model-driven optimization; Integrated perception; Data-driven optimization; Bidirectional empowerment of AI and optimization; Prediction-decision integration

1 INTRODUCTION

The evolution from mathematical programming to intelligent decision-making. As the core of decision science, optimization algorithms have undergone a remarkable journey from classical linear programming to optimizing large-scale complex systems[1]. Since Dantzig introduced the simplex method, optimization theory has continuously matured within a rigorous mathematical programming framework, establishing a pedagogical paradigm anchored by KKT conditions, dual theory, and branch-and-bound methods. However, driven by surging demands from global supply chains and real-time industrial scheduling, optimization algorithms are evolving from offline, static computation into highly integrated, perception-aware intelligent decision engines. The evolution of neural network architectures is reshaping the very logic underlying our approach to combinatorial optimization problems[2-3].

Model-based traditional optimization education primarily relies on the model-driven paradigm, which assumes precisely known input parameters and relatively stable business environments. However, in the era of big data, decision systems face massive, heterogeneous, and dynamically changing sensory data[4]. This has rendered traditional two-stage approaches (predict first, then optimize) inadequate when addressing the mismatch between prediction errors and decision costs[5]. The Smart Predict-then-Optimize framework proposed by Elmachtoub and Grigas reveals the core of this paradigm shift from pursuing pure prediction accuracy to achieving optimality in final decisions[6]. This shift from deterministic mathematical programming to probabilistic AI induction constitutes the fundamental impetus for this curriculum reform.

The shortage of interdisciplinary dual-competence experts in cutting-edge industrial fields like smart logistics, autonomous driving, and energy dispatch, corporate demands for talent have shifted from single-dimensional algorithm implementation to complex system coordination[7].The current job market exhibits a pronounced talent gap: students with operations research backgrounds often lack the ability to handle unstructured data, while those with deep learning backgrounds frequently fail to make sound decisions due to a lack of respect for hard constraints. As demonstrated by Gasse et al. in their research on accelerating branch-bounding strategies using Graph Neural Networks (GNN) to accelerate branch-and-bound strategies, future algorithm specialists must possess the "amphibious" ability to seamlessly switch between mathematical rigor and AI intuition[8].

In response to the aforementioned challenges and needs, this paper aims to explore a systematic reform plan for the graduate course of optimization theory and methods. We propose a three-tiered pyramid course reconstruction framework, encompassing a comprehensive pathway from automatic differentiation to differentiable optimization. This paper also demonstrates practical implementation through dynamic path planning and provide quantitative evaluation of reform outcomes and offer an expert-driven outlook on future directions for explainable neural optimizers.

2 CHALLENGES IN TRADITIONAL OPTIMIZATION ALGORITHM COURSES

Against the backdrop of explosive growth in artificial intelligence technology, traditional optimization algorithm teaching models have revealed significant lag when handling high-dimensional unstructured data and real-time dynamic decision-making. Table 1 contrasts the generational gap between traditional teaching paradigms and the real-world challenges faced by industry in the era of AI. This paradigm mismatch has led to severe "high prediction accuracy but low decision-making benefits" misalignment. If traditional teaching systems persist, graduate students will struggle to bridge the technical gap between perception and decision-making, severely hindering the deployment of complex intelligent systems (e.g., smart logistics, autonomous driving). Therefore, the educational transition from model-driven to data+model dual-engine is now imperative.

Table 1 Comparison of Traditional Optimization Teaching Paradigms and AI-Era Challenges

Dimension	Traditional Paradigm	AI Era Challenges
Knowledge System	convex optimization theory; duality analysis; rigorous convergence proofs; lower bounds for decision optimality	unstructured perceptual data; minimized empirical risk under hard constraints
Tool chain	closed and isolated ecosystems: AMPL, GAMS, MATLAB	open and Integrated: AI frameworks Torch, TensorFlow, JAX
Evaluation Dimensions	single and static	multifaceted and dynamic:
Talent Development	operations research students lack data processing tools; deep learning students lack respect for mathematical constraints.	professionals capable of seamlessly switching between mathematical rigor and AI intuition, and the ability to handle complex system coordination.

3 THREE-TIERED PYRAMID CURRICULUM RECONSTRUCTION FRAMEWORK

Addressing knowledge fragmentation and tool generational gaps in traditional optimization curricula during the AI era, this study proposes a three-tiered pyramid curriculum restructuring system. This framework aims to establish a teaching paradigm deeply integrating model-driven and data-driven approaches through progressive tiers: foundational layer theory reshaping, advanced layer bidirectional empowerment, and application layer end-to-end decision-making (Figure 1).

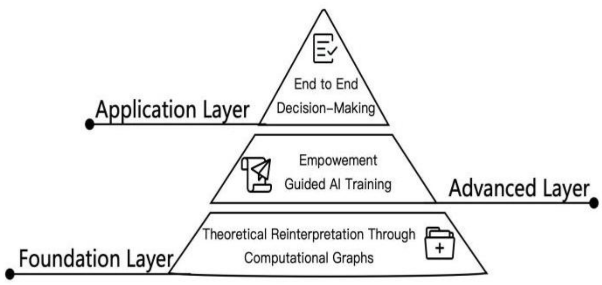

Figure 1 Data + Model Dual-Engine Driven

3.1 Foundation Layer: Theoretical Reinterpretation Through Computational Graphs

The core of foundational layer instruction lies in transcending traditional algebraic derivations by introducing a computational graph perspective to modernize classical optimization theory. (1) integrating auto-differentiation with convex optimization theory: Embedding classic algorithms like gradient descent and Newton's method into modern computational graph frameworks enables students to understand how automatic differentiation simplifies gradient computation for complex convex optimization problems. (2) applying Lagrange duality in deep learning regularization. This will lead students to reexamine loss functions and regularization terms in neural networks through a dual perspective, exploring the underlying mathematical logic of constrained optimization in weight decay and robust training.

3.2 Advanced Layer: Bidirectional Empowerment of AI and Optimization

This section serves as the eye of the storm within the entire curriculum, aiming to break down disciplinary barriers between AI and operations research. We no longer view them as isolated tools but explore how they deeply couple to form a symbiotic relationship of AI-driven, optimization-implemented.

3.2.1 Optimization algorithms guide AI training: from "alchemy" to mathematical formulation

In traditional deep learning, hyper-parameter tuning (HPO) is often derided as random mysticism. This section re-frames AI training through the rigorous lens of operations research, reconstructing it as a classic bi-level programming problem.

(1) Bi-level Programming Modeling: We treat the weight updates of neural networks as the inner-level problem (Lower -level), while the search for hyperparameters such as learning rate, regularization coefficient, and network depth is the outer-level problem (Upper-level).

(2) Learning Objectives: Guide students in understanding how to solve such two-layer models using KKT conditions, gradient approximations, or evolutionary algorithms. This not only enhances the efficiency of HPO, but also cultivates students' ability to examine AI training processes from a global perspective, equipping them with advanced engineering literacy to refine underlying algorithmic frameworks (Figure 2).

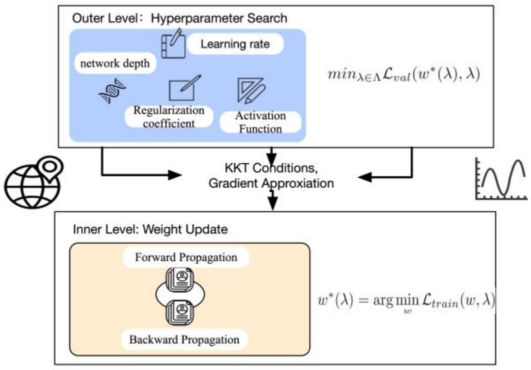

Figure 2 Bi-Level Programming Modeling

3.2.2 AI model-assisted optimization case study: overcoming performance bottlenecks of traditional algorithms

When tackling large-scale, nonlinear combinatorial optimization problems, traditional solvers often face the dimension disaster. This section highlights how AI's perceptual and learning capabilities can endow optimization algorithms with spirituality.

(1) Neural Combinatorial Optimization (NCO): Transcending Greedy Strategies

Traditional heuristic algorithms rely on expert knowledge, whereas NCO enables machines to autonomously learn solution paths. Incorporating attention mechanisms to address mismatches in input and output sequence lengths, particularly effective for Traveling Salesman Problems (TSP) and Vehicle Routing Problems (VRP). Models the solution process as a Markov Decision Process (MDP), iterating through reward mechanisms to enable the model to autonomously discover superior solution space search paths compared to traditional heuristics—without requiring labeled data.

(2) Learning to Branch (L2B): Equipping solvers with a brain

The core of Mixed-Integer Programming (MIP) lies in Branch-and-Bound, whose efficiency heavily depends on the quality of the branching rules[1]. Models MIP problems as bipartite graphs, using GNN to extract topological features between variables and constraints. Compared to traditional manual feature engineering, GNNs capture deeper structural information and train a lightweight AI model by mimicking computationally intensive expert rules. During actual solving, the trained GNN rapidly predicts branching decisions, achieving —expert-level decision accuracy with millisecond execution speed. The core challenge lies in converting abstract mathematical algebraic constraints into spatial topological structures perceivable by AI.

3.2.3. Case teaching process:

To foster students' intuitive understanding, the case design process incorporates decomposition, cultivating a dual-habitat capability that bridges mathematical rigor and AI intuition. The experimental design phase is illustrated in Table 2.

Table 2 Advanced-Level Data + Model Dual-Engine Driven Case Design

Stage	Task Content	Key Technical Points
Data Collection	Run solver, record node characteristics and expert decisions during branching processes	PyPSA or SCIP interface interface invocation
Implementation of the composition	Convert extracted matrix data into PyTorch Geometric graph objects	Heterogeneous data
Model Training	Employing cross-entropy loss function to mimic expert branch selection	Behavior cloning
Ensemble Testing	Feed GNN predictions back to the solver and compare the total number of branches	Solver callback function

3.3 Application Layer: End-to-End Decision Making and Differentiable Optimization

As the top layer of this course system, the application layer focuses on resolving the goal mismatch in the prediction-optimization process, achieving a closed-loop transition from unstructured data to optimal decisions. In the traditional two-stage paradigm of improvements to prediction models often fail to translate into enhanced decision benefits—a phenomenon known as high precision but low yield. Instruction guides students to employ differentiable optimization techniques, directly feeding decision losses back to the perception front-end.

3.3.1 Decision regret-driven loss function reconstruction

In application-level instruction, we shift the core evaluation metric from traditional Mean Squared Error (MSE) to Decision Regret, which lies on the target mismatch analysis (Figure 3).

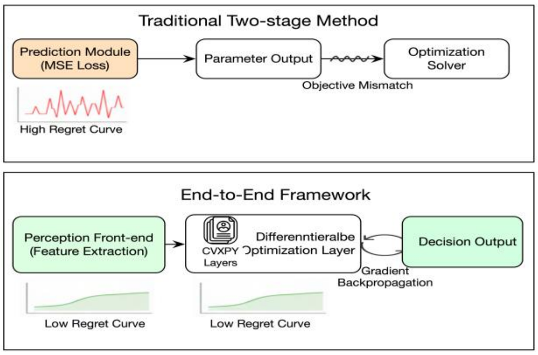

Figure 3 The Comparative Flow of Prediction-Decision Integration

We dissect the root cause of amplified prediction errors under nonlinear decision constraints in traditional two-stage methods, stemming from the mismatch between the prediction objective function and the final optimization objective function. Smart Predict-then-Optimize (SPO) loss function shifts from merely pursuing the convergence of predicted parameters $\hat{c}$ to true parameters C, but rather minimizes the loss incurred by decisions guided by $\hat{c}$ under the actual environment C. Teaching points is set to guide students to understand how to leverage the duality of linear programming to construct a surrogate loss function spanning prediction and decision-making.

3.3.2 Implementation of differentiable optimization layers based on the implicit function theorem

To achieve end-to-end joint training, the optimization process itself must be differentiable. Based on research by Amos and Kolter[9], use the Implicit Function Theorem to compute the Jacobian matrix of the solution x* to the optimization problem with respect to the input parameters θ. Guide students to embed convex optimization problems directly into PyTorch neural networks using CVXPY Layers. In asset allocation optimization or energy dispatch scenarios, deploy an optimization layer with hard physical constraints as the final network layer to achieve direct mapping from perceptual features to optimal operational commands.

3.3.3 End-to-end closed-loop evaluation system

This phase requires students to achieve cognitive elevation from mathematical exact solutions to business-optimal decisions. The evaluation model demonstrates whether decision outcomes obtained through end-to-end training exhibit greater stability than traditional methods when confronted with prediction inputs containing 5%-10% noise. Simultaneously, experimental comparisons enable students to deeply comprehend how AI endows algorithms with an intuition for handling uncertainty, while the KKT conditions of mathematical programming serve as the defensive line ensuring decisions satisfy physical constraints.

The evaluation model demonstrates whether decision outcomes obtained through end-to-end training exhibit greater stability than traditional methods when confronted with prediction inputs containing 5%-10% noise. Simultaneously, experimental comparisons enable students to deeply comprehend how AI endows algorithms with an "intuition" for handling uncertainty, while the KKT conditions of mathematical programming serve as the "defensive line" ensuring decisions satisfy physical constraints.

4 EVALUATION OF TEACHING REFORM OUTCOMES

As the culminating project of this curriculum, the final project aims to overcome the limitations of paper-based learning. It requires students to validate the effectiveness of the data + model dual-engine approach in complex industrial-scale scenarios. This project, set against the backdrop of real-time logistics scheduling in smart cities, requires students to construct an end-to-end decision-making loop, when confronted with high-dimensional perception data (e.g., traffic conditions, weather) and rigid physical constraints (e.g., vehicle load capacity, time windows).

4.1 Project Theme: Real-Time Dynamic Scheduling for Large-Scale Smart Logistics Distribution

Traditional VPR often assumes travel time as a fixed constant. However, in real-world scenarios, the randomness of road network congestion causes static solutions to fail easily during execution[10]. This project requires students to develop a real-time perception-enabled neural optimizer.

4.2 Task Breakdown and Technical Implementation Path

4.2.1 Perception phase: feature extraction and prediction from unstructured data

Students utilize historical road network data to construct spatial-temporal sequence models. The task objective is to predict average travel speeds and congestion probabilities for critical urban road segments within the next hour. The teaching focus is understanding how prediction error propagates through the coefficient matrix of the optimization problem to the final decision.

4.2.2 Decision phase: online inference of neural combinatorial optimization

The VRP is a typical NP-hard problem, students abandon traditional greedy algorithms and apply pointer networks. The task objective is to generate trajectory sequences for multiple delivery vehicles within milliseconds based on the dynamic time matrix output from the perception phase. The key technology is integrating reinforcement learning frameworks, training with total path cost, including delay penalties, as the reward function to enable autonomous avoidance of congested areas.

4.2.3 Closed-loop evaluation: from mathematical optimality to operational excellence

Students compare the performance of three paradigms within the dynamic simulation environment. The baseline group employs a traditional static VRP model, which is first predicting means, then computing with a solver. The improved approach utilizes a two-stage ensemble method, which is using predicted values as input to invoke classical heuristic algorithms. The final experimental group features a differentiable, end-to-end trained model.

4.3 Evaluation Criteria and Expert Review

The final evaluation transcends traditional focus on whether a route achieves mathematical global optimality. Instead, it emphasizes three industrial-grade dimensions: decision regret metric, constraint compliance and reasoning timeliness. The decision regret metric describes the gap between a student's initial decision and the post-event optimal solution after real-world traffic conditions unfold. The constraint compliance is measuring whether model-generated decisions fully satisfy hard constraints like vehicle load capacity and customer delivery time windows. Reasoning timeliness is to assess whether the algorithm can generate adjustment plans within seconds during real-time order surges.

5 CONCLUSION

Optimization theory forms the foundation of intelligent decision-making, while AI provides the wings of perception. This pedagogical reform represents not merely an upgrade of tools, but a fundamental reshaping of students' decision-making, cultivating both reverence for mathematical constraints and embracing data-driven flexibility. Future teaching priorities should shift from black-box models to explainable neural optimization, exploring how to translate AI's implicit insights into human-understandable explicit rules.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This work is supported by the Key Education Reform Project of Yunnan Provincial Society for Academic Degrees and Graduate Education (Grand NO: YYXH202318).

REFERENCES

[1] Watts J, Wyk F V, Khojandi M A. A dynamic deep reinforcement learning-bayesian framework for anomaly detection. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(12): 22884-22894.
[2] Wen J, Abeel T, De Weerdt M. Performance and interaction assessment of neural network architectures and bivariate smart predict-then-optimize. Machine Learning, 2025, 114(11).
[3] Qu K, Zhuang W. Scalable and dynamic cooperative perception: a data/model co-driven framework. IEEE Network, 2024(6): 38.
[4] Mandi J, Demirovic E, Stuckey P J, et al. Smart predict-and-optimize for hard combinatorial optimization problems. Proceedings of the AAAI Conference on Artificial Intelligence, 2020.
[5] Fan Z, Ghaddar B, Wang X, et al. Artificial intelligence for optimization: unleashing the potential of parameter generation, model formulation, and solution methods. European Journal of Operational Research, 2025,9(11).
[6] Balghiti O E, Grigas P, Elmachtoub A N, et al. Generalization bounds in the predict-then-optimize framework. Advances in Neural Information Processing Systems, 2020, 32: 3289-3299.
[7] Brahmi Z, Mezni H, Elmannai H, et al. Multi-modal vaas selection in smart mobility networks via spectral hyper-graph clustering and quantum-driven optimization. Concurrency & Computation: Practice & Experience, 2025, 37(23/24).
[8] Gasse M, Ferroni N, Chetelat D, et al. Exact combinatorial optimization with graph convolutional neural networks. Advances in Neural Information Processing Systems, 2020, 32: 15554-15566.
[9] Agrawal A, Amos B, Barratt S, et al. Differentiable convex optimization layers. Advances in Neural Information Processing Systems, 2019, 32: 9558-9570.
[10] Nandikolla V K, Ferman K, Barragan E, et al. Vision based obstacle detection and navigation of an autonomous vehicle. ASME International Mechanical Engineering Congress and Exposition. American Society of Mechanical Engineers, 2021, 85628: V07BT07A016.