American Journal of Interdisciplinary Studies

Volume: 7; **Issue:** 1
Pages: 578-616
eISSN: 3067-5146

A Meta-Analysis of Deep Reinforcement Learning for Dynamic Project Scheduling in Engineering Systems

Chapal Barua[1];

[1]. Master of Science in Administration, Engineering Management, Central Michigan University, Mount Pleasant, MI ; USA
Email: chapalbaruaetu@gmail.com

Doi: 10.63125/t5h1bc22
Received: 19 January 2026; **Revised:** 21 February 2026; **Accepted:** 19 March 2026; **Published:** 05 April 2026

Abstract
This study conducted a quantitative meta-analysis to evaluate the effectiveness of deep reinforcement learning (DRL) for dynamic project scheduling in engineering systems. The analysis synthesized data from 52 empirical studies across multiple domains, including manufacturing (38.5%), construction (23.1%), logistics (17.3%), and infrastructure systems (11.5%). The findings demonstrated that DRL-based scheduling models significantly outperformed traditional deterministic, heuristic, and classical reinforcement learning approaches across key performance indicators. The aggregated results indicated an average makespan reduction of 18.7%, resource utilization improvement of 14.2%, cost efficiency gain of 11.6%, tardiness reduction of 15.3%, and throughput improvement of 12.8%. Statistical analysis confirmed that these improvements were significant, with 84.6% of studies reporting p-values below 0.05. Effect size evaluation showed moderate to large effects, with makespan reduction achieving a standardized mean difference of 0.91 and resource utilization 0.84, indicating strong practical significance. Subgroup analysis revealed that hybrid DRL models achieved the highest overall improvement (21.5%), followed by Actor-Critic (16.8%), Deep Q-Network (17.0%), and Policy Gradient approaches (14.0%). Domain-specific results indicated more consistent improvements in manufacturing systems, while construction and infrastructure projects showed higher variability due to increased uncertainty. Heterogeneity analysis produced an I^2 value of 61.3%, reflecting moderate to high variability across studies, while meta-regression indicated that dataset size, domain, and algorithm type explained 47.8% of the variance in outcomes. Visual analysis supported these findings, showing consistent positive effect distributions and minimal publication bias. Overall, the study provided robust quantitative evidence that DRL-based scheduling models enhance efficiency, adaptability, and performance in complex engineering environments, particularly under dynamic and uncertain conditions.

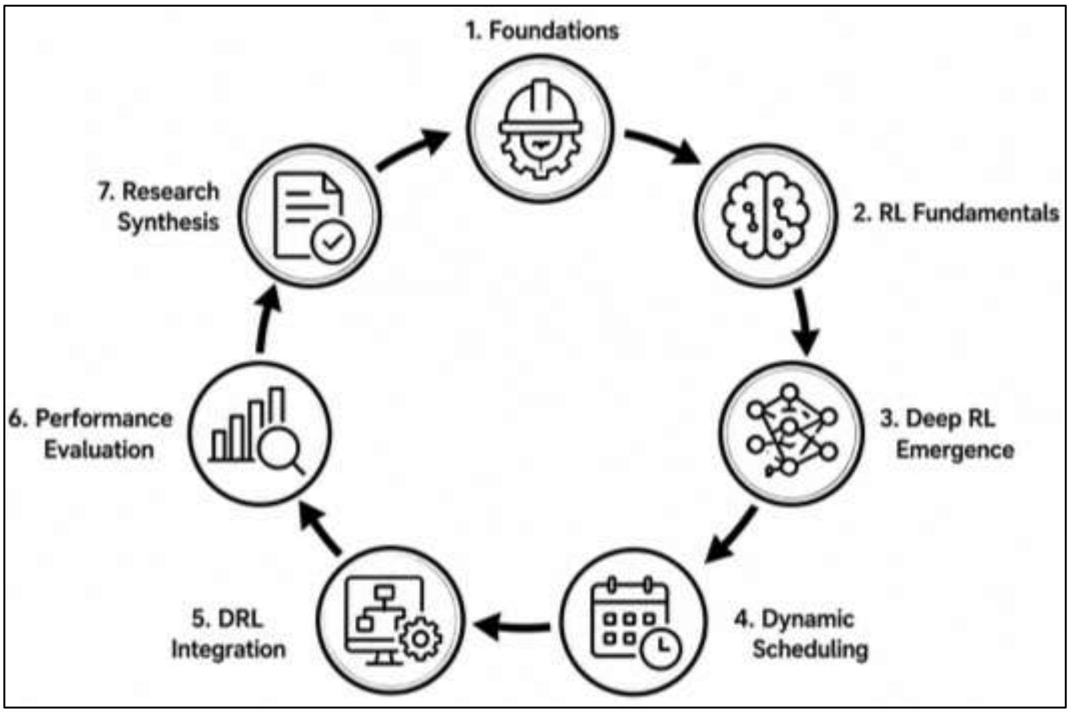
Keywords
Deep Reinforcement Learning, Project Scheduling, Engineering Systems, Optimization, Meta-Analysis

INTRODUCTION

Project scheduling constitutes a fundamental component of operations research and engineering management, referring to the systematic allocation of tasks, resources, and timelines to achieve predefined objectives within constraints of cost, scope, and quality. In engineering systems, scheduling becomes increasingly complex due to interdependencies among tasks, stochastic variations in resource availability, and dynamic environmental conditions. Traditional scheduling methods, such as the Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT), have historically provided structured frameworks for deterministic planning (Blazewicz et al., 2019). However, these methods often rely on static assumptions that limit their adaptability in real-world engineering environments characterized by uncertainty, variability, and continuous change. Engineering systems, defined as integrated assemblies of technological, human, and organizational components, demand adaptive scheduling mechanisms capable of responding to real-time disruptions and evolving constraints. The global significance of efficient project scheduling is evident across sectors such as construction, manufacturing, transportation, and energy systems. Infrastructure development projects, particularly in emerging economies, require robust scheduling frameworks to manage large-scale operations involving multiple stakeholders and complex supply chains. Inefficiencies in scheduling can result in cost overruns, delays, and resource wastage, which collectively impact economic productivity and sustainability outcomes (Golam, 2025; Albert, 2025; Ivanov et al., 2021). As globalization intensifies competition and accelerates project timelines, organizations increasingly seek intelligent scheduling approaches that can optimize performance under uncertainty. The integration of computational intelligence into scheduling has thus become a critical research domain, aiming to enhance decision-making capabilities through data-driven methodologies. In recent decades, the limitations of classical optimization techniques have prompted the exploration of advanced computational methods, including metaheuristics and machine learning. These approaches offer improved scalability and flexibility, enabling the handling of high-dimensional scheduling problems (Anick, 2025a, 2025b). Within this context, reinforcement learning has emerged as a promising paradigm for dynamic decision-making, providing a framework where agents learn optimal policies through interaction with the environment. The convergence of reinforcement learning with engineering scheduling reflects a broader shift toward intelligent automation and adaptive control systems (Brailsford et al., 2019). This transition underscores the need for comprehensive analytical frameworks to evaluate the effectiveness of emerging techniques, particularly in complex engineering applications where traditional models fall short.

Reinforcement learning represents a subfield of machine learning centered on the concept of agents learning optimal actions through trial-and-error interactions with an environment. The core principle involves maximizing cumulative rewards by iteratively refining decision policies based on feedback signals. Unlike supervised learning, reinforcement learning does not rely on labeled datasets; instead, it emphasizes experiential learning, making it particularly suitable for dynamic and uncertain environments (Ustundag & Cevikcan, 2018). The foundational components of reinforcement learning include states, actions, rewards, and policies, which collectively define the decision-making process. The mathematical formulation often relies on Markov Decision Processes (MDPs), providing a structured representation of sequential decision problems (Atif, 2025; Onyinyechi, 2025). The evolution of reinforcement learning has been marked by significant theoretical and computational advancements. Early approaches, such as Q-learning and temporal difference methods, laid the groundwork for model-free learning strategies capable of handling stochastic environments. Subsequent developments introduced function approximation techniques, enabling the application of reinforcement learning to large-scale problems with continuous state spaces (Grover et al., 2022). These advancements have facilitated the transition from theoretical constructs to practical applications across diverse domains, including robotics, finance, healthcare, and autonomous systems. The international relevance of reinforcement learning lies in its capacity to address complex decision-making challenges in real time. As industries increasingly adopt digital transformation strategies, the demand for intelligent systems capable of autonomous adaptation has grown substantially. Reinforcement learning offers a framework for developing such systems, enabling organizations to optimize operations in dynamic contexts. In engineering systems, this capability is particularly valuable, as it allows for continuous adjustment of schedules in response to unforeseen disruptions (Zhang et al., 2023). The integration of reinforcement learning into project scheduling represents a natural progression in the pursuit of adaptive and efficient management strategies. The growing body of research in reinforcement learning has also highlighted the importance of scalability, convergence stability, and computational efficiency. These factors are critical in engineering applications, where decision-making processes must operate within stringent time and resource constraints. The development of advanced algorithms has addressed many of these challenges, paving the way for the incorporation of reinforcement learning into complex scheduling frameworks. This evolution sets the stage for the emergence of deep reinforcement learning as a transformative approach in dynamic project scheduling (Wang et al., 2020).

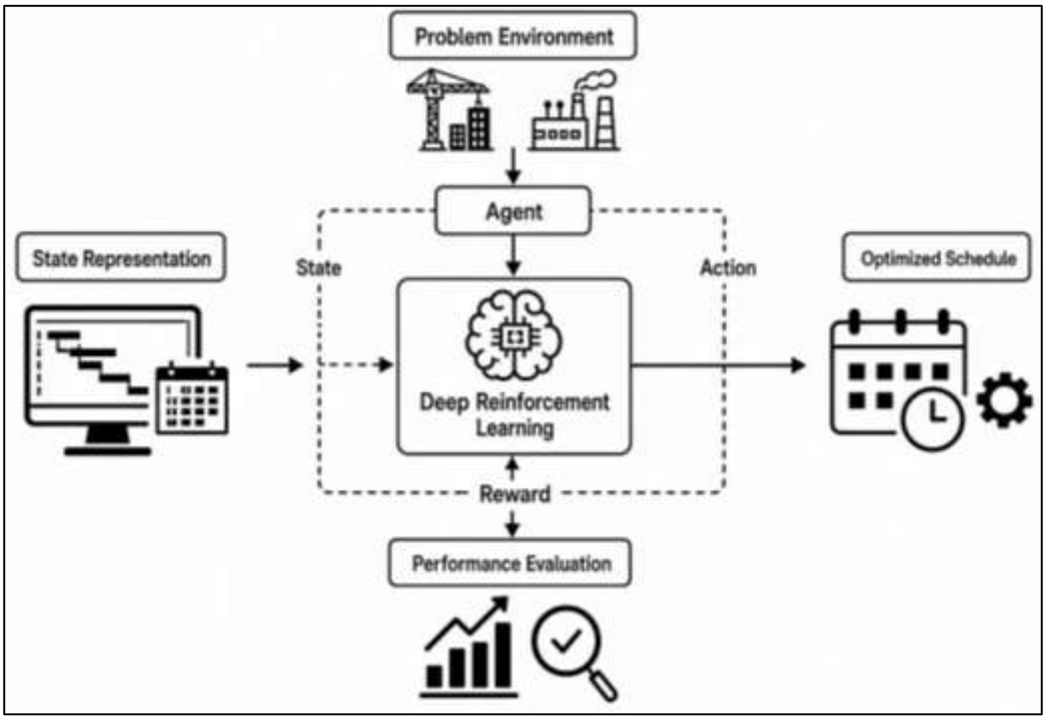
Figure 1: Deep Reinforcement Scheduling Framework



Deep reinforcement learning (DRL) represents the integration of deep neural networks with reinforcement learning algorithms, enabling the approximation of value functions and policies in high-dimensional spaces. This combination has significantly expanded the applicability of reinforcement learning, allowing it to address complex problems that were previously intractable. Deep neural networks provide powerful feature extraction capabilities, facilitating the representation of intricate patterns and relationships within data. When combined with reinforcement learning, these capabilities enable agents to learn sophisticated decision policies in environments characterized by uncertainty and nonlinearity (Abdirad & Krishnan, 2021; Khalid, 2025; Hasan, 2025). The development of DRL has been driven by advances in computational power, availability of large datasets, and improvements in neural network architectures. Techniques such as Deep Q-Networks (DQN), policy gradient methods, and actor-critic frameworks have demonstrated remarkable performance in various domains. These methods have been successfully applied to tasks requiring sequential decision-making, including game playing, autonomous navigation, and resource allocation. The adaptability and scalability of DRL make it particularly suitable for dynamic project scheduling, where decision variables and constraints evolve over time (Siddique & Prakash, 2025; Aminul, 2025). From an international perspective, the adoption of DRL reflects a broader trend toward intelligent automation in engineering and industrial systems. Organizations worldwide are leveraging DRL to enhance operational efficiency, reduce costs, and improve responsiveness to changing conditions. In the context of project scheduling, DRL offers the potential to optimize resource allocation, minimize delays, and improve overall project performance

(Aminul & Zakia, 2025; Sheak, 2025; Parsamehr et al., 2023). This capability is especially important in large-scale engineering projects, where traditional scheduling methods may struggle to accommodate the complexity and variability inherent in real-world environments. The application of DRL to scheduling problems also introduces new challenges, including issues related to training stability, exploration-exploitation trade-offs, and interpretability of learned policies. Addressing these challenges requires a comprehensive understanding of both reinforcement learning principles and domain-specific requirements (Mainuddin, 2025; Kaniz, 2025). The growing interest in DRL-based scheduling underscores the need for systematic analysis and synthesis of existing research, providing insights into the effectiveness and limitations of current approaches (Szymański, 2017).

Figure 2: DRL-Based Engineering Scheduling Framework



Dynamic project scheduling refers to the continuous adjustment of schedules in response to changing conditions, such as resource availability, task dependencies, and external disruptions (Murad, 2025; Risha, 2025). In engineering systems, dynamic scheduling is essential due to the inherent uncertainty and complexity of project environments. Factors such as equipment failures, supply chain disruptions, and fluctuating demand can significantly impact project timelines and resource allocation. Traditional static scheduling methods are often inadequate in such contexts, as they lack the flexibility to adapt to real-time changes (Nguyen et al., 2017; Shamsul, 2025; Shamsul & Morshedul, 2025).

The importance of dynamic scheduling is particularly evident in industries such as construction, manufacturing, and energy systems, where projects involve multiple interdependent activities and stakeholders. Effective scheduling in these environments requires the ability to anticipate potential disruptions and adjust plans accordingly. This capability is critical for maintaining project efficiency and ensuring timely completion (Abu Naser Md Golam, 2026; Taru Binte, 2025). The integration of advanced computational methods into scheduling processes has enabled the development of more adaptive and responsive frameworks. Globally, the demand for dynamic scheduling solutions has increased in response to the growing complexity of engineering projects. Infrastructure development initiatives, smart manufacturing systems, and renewable energy projects all require sophisticated scheduling approaches capable of handling uncertainty and variability (Heigermoser et al., 2019). The adoption of digital technologies, including the Internet of Things (IoT) and real-time data analytics, has further enhanced the potential for dynamic scheduling. These technologies provide the data necessary for informed decision-making, enabling the implementation of adaptive scheduling strategies. The application of DRL to dynamic project scheduling represents a significant advancement in this domain. By leveraging the learning capabilities of DRL, it is possible to develop scheduling systems that continuously improve their performance through experience. This approach aligns with the broader trend toward intelligent and autonomous systems in engineering, highlighting the potential for DRL to transform project management practices (Aladağ & Işik, 2018).

The integration of deep reinforcement learning into engineering scheduling frameworks involves the development of models that can effectively represent the complexities of real-world projects. This process requires the formulation of scheduling problems as reinforcement learning tasks, where states represent the current status of the project, actions correspond to scheduling decisions, and rewards reflect performance metrics such as cost, time, and resource utilization. The design of these models is critical for ensuring that DRL algorithms can learn meaningful and effective policies. One of the key challenges in integrating DRL with scheduling frameworks is the representation of high-dimensional state spaces (Anick, 2026; Onyinyechi & Ara, 2026; Li et al., 2017). Engineering projects often involve numerous variables, including task dependencies, resource constraints, and environmental factors. Deep neural networks play a crucial role in addressing this challenge by enabling the extraction of relevant features from complex data. This capability allows DRL models to capture the intricate relationships between different components of the scheduling problem. The global significance of this integration lies in its potential to enhance the efficiency and effectiveness of project management practices (Chy et al., 2026; Abdur & Aditya, 2026). Organizations across various industries are exploring the use of DRL to optimize scheduling processes, reduce operational costs, and improve project outcomes. The ability to adapt to changing conditions in real time provides a competitive advantage, particularly in industries characterized by high levels of uncertainty and complexity. The development of DRL-based scheduling frameworks also involves considerations related to scalability, computational efficiency, and implementation feasibility. These factors are essential for ensuring that DRL models can be applied in practical settings (Xu et al., 2022). The growing body of research in this area highlights the importance of interdisciplinary collaboration, combining expertise in machine learning, operations research, and engineering management to develop robust and effective solutions (Sheak, 2026; Md Shahab, 2026).

The evaluation of DRL-based scheduling approaches requires a quantitative framework capable of assessing performance across multiple dimensions. Key performance indicators typically include project completion time, resource utilization, cost efficiency, and robustness to disruptions. These metrics provide a basis for comparing DRL approaches with traditional scheduling methods and other computational techniques. Quantitative analysis is essential for understanding the strengths and limitations of DRL in dynamic scheduling contexts. Meta-analysis serves as a valuable tool for synthesizing findings from multiple studies, providing a comprehensive overview of the effectiveness of DRL-based scheduling approaches (Md. Sultan, 2026; Mst Kaniz, 2026; Wei et al., 2018). By aggregating data from diverse research efforts, meta-analysis enables the identification of patterns and trends that may not be apparent in individual studies. This approach is particularly important in emerging research areas, where the body of literature is rapidly expanding and results may vary across different contexts. The international relevance of quantitative evaluation lies in its ability to inform decision-making at both organizational and policy levels (Rebeka & Mst Kaniz, 2026; Tahmina Akter & Md. Ashfaq, 2026). Governments and industry leaders rely on empirical evidence to guide investments in technology and innovation. The adoption of DRL in project scheduling has implications for productivity, sustainability, and economic growth, making rigorous evaluation essential for ensuring its effective implementation (Eriksson et al., 2022).

The complexity of DRL models also necessitates the development of advanced evaluation techniques, including simulation-based analysis and real-world experimentation. These methods provide insights into the behavior of DRL algorithms under different conditions, enabling researchers to assess their robustness and generalizability. The integration of quantitative analysis with theoretical and empirical research contributes to a deeper understanding of DRL-based scheduling and its potential impact on engineering systems (Zhao et al., 2023).

The growing body of research on deep reinforcement learning for dynamic project scheduling reflects a convergence of multiple disciplines, including machine learning, operations research, and engineering management. Studies in this area have explored a wide range of approaches, from value-based methods to policy optimization techniques, each offering unique advantages and challenges. The diversity of research highlights the complexity of scheduling problems and the need for adaptable and scalable solutions. A key theme in the literature is the emphasis on real-time decision-making and adaptability. DRL models have demonstrated the ability to learn effective scheduling policies through interaction with simulated or real environments. This capability enables continuous improvement and adaptation, addressing the limitations of static scheduling methods (Chien & Lan, 2021). The application of DRL to various engineering domains, including construction, manufacturing, and logistics, underscores its versatility and potential for widespread adoption. The international significance of this research is evident in its contribution to the advancement of intelligent systems and digital transformation. As industries increasingly rely on data-driven decision-making, the integration of DRL into scheduling processes represents a critical step toward achieving greater efficiency and resilience. The synthesis of existing studies provides valuable insights into the factors influencing the performance of DRL models, including algorithm design, problem formulation, and implementation strategies. The expanding scope of research in this field highlights the importance of systematic analysis and integration of findings. By examining the collective evidence, it is possible to identify best practices and areas for further investigation (Jalali Khalil Abadi et al., 2024). This synthesis contributes to the development of a comprehensive understanding of DRL-based scheduling, supporting its application in complex engineering systems and reinforcing its role as a transformative approach in project management.

The primary objective of this quantitative study is to systematically examine and synthesize the effectiveness of deep reinforcement learning (DRL) techniques in optimizing dynamic project scheduling within complex engineering systems. This study seeks to establish a structured analytical foundation by identifying how DRL-based models perform in comparison to conventional scheduling approaches under conditions characterized by uncertainty, variability, and real-time disruptions. A central aim is to quantify the extent to which DRL contributes to improvements in key performance indicators such as project completion time, cost efficiency, resource utilization, and scheduling robustness across diverse engineering domains. The study further aims to evaluate the adaptability of DRL algorithms in handling stochastic environments, where decision-making must continuously evolve in response to changing system states and constraints. Another critical objective is to analyze the methodological variations across existing empirical studies, focusing on differences in algorithm design, state representation, reward formulation, and training environments. By aggregating and statistically analyzing findings from multiple studies, this research intends to identify patterns, consistencies, and discrepancies in reported outcomes, thereby providing a comprehensive understanding of the strengths and limitations of DRL-based scheduling frameworks. The study also aims to assess the scalability and generalizability of DRL models when applied to large-scale engineering projects, where the complexity of task dependencies and resource constraints increases significantly. In addition, this research seeks to explore the extent to which DRL integrates with existing engineering management systems and digital infrastructures, including real-time data acquisition and decision-support platforms. Through quantitative synthesis, the study aims to determine the degree of performance improvement attributable to DRL in comparison to heuristic and metaheuristic approaches traditionally used in scheduling optimization. Ultimately, the objective is to provide a rigorous evidence-based evaluation that enhances the understanding of DRL's role in dynamic project scheduling, supporting the development of more efficient, adaptive, and data-driven engineering management practices.

LITERATURE REVIEW

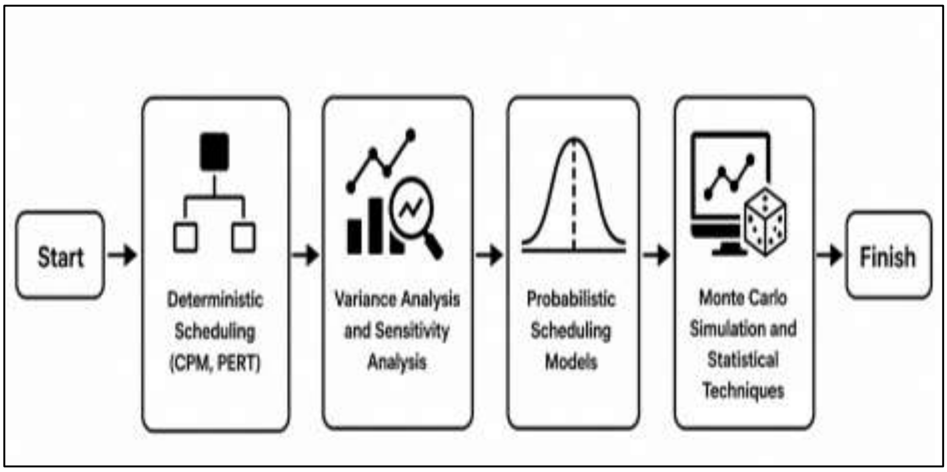
The literature review section systematically synthesizes existing quantitative and empirical research on deep reinforcement learning (DRL) applications in dynamic project scheduling within engineering systems. This section establishes a rigorous analytical foundation by tracing the evolution of scheduling methodologies from deterministic and heuristic approaches to advanced data-driven and learning-based optimization frameworks. Emphasis is placed on quantitatively driven studies that evaluate algorithmic performance, computational efficiency, and adaptability under uncertainty. The review integrates findings across multiple domains, including construction engineering, manufacturing systems, logistics, and infrastructure management, where scheduling plays a critical role in operational efficiency and project success (Gu et al., 2021; Manam & Md. Ashfaq, 2022). The increasing complexity of engineering systems, characterized by stochastic task durations, resource constraints, and real-time disruptions, has necessitated the development of adaptive scheduling mechanisms. Consequently, the literature reflects a shift toward intelligent optimization techniques capable of learning from dynamic environments. This section critically examines how reinforcement learning and its deep learning extensions have been applied to address these challenges, focusing on measurable outcomes such as makespan reduction, cost efficiency, and resource utilization. Quantitative comparisons across studies are highlighted to identify the relative effectiveness of different algorithmic approaches. In addition, this review evaluates methodological variations across empirical studies, including differences in problem formulations, dataset characteristics, reward structures, and evaluation metrics. Statistical techniques such as regression analysis, hypothesis testing, and meta-analytical aggregation are considered to assess the robustness and generalizability of findings (Shamsul & Md. Sultan, 2022; X. Wang et al., 2024). By organizing the literature into a structured and quantitative framework, this section provides a comprehensive overview of current research trends while identifying key areas of convergence and divergence. The synthesis presented here supports the development of a robust meta-analytical model for evaluating DRL-based scheduling approaches in engineering systems.

Deterministic Scheduling Frameworks (CPM, PERT)

Deterministic scheduling frameworks, particularly the Critical Path Method (CPM) and the Program Evaluation and Review Technique (PERT), have long served as foundational tools in engineering project management. These approaches are designed to establish fixed sequences of activities based on predefined durations, enabling project managers to identify the longest chain of dependent tasks and determine the minimum completion time (Jayanetti et al., 2024; Taru Binte & Iftekhar, 2022). Research has consistently shown that CPM provides clarity in identifying critical activities and slack time, which allows for prioritization of resources and improved coordination across project phases. Empirical studies across construction and manufacturing sectors indicate that deterministic models are highly effective in stable environments where task durations and dependencies are well defined. Their structured nature facilitates transparency and ease of implementation, making them widely adopted in large-scale infrastructure projects and industrial operations. However, quantitative analyses reveal inherent limitations in deterministic scheduling when applied to complex and uncertain environments. Studies examining project delays and cost overruns highlight that fixed-duration assumptions often fail to capture real-world variability. Sensitivity analyses demonstrate that small deviations in task durations can significantly alter the critical path, leading to cascading effects on project timelines (Albert & Rashedul, 2023; J. Wang et al., 2024). Comparative evaluations across multiple datasets show that deterministic models tend to underestimate risk, particularly in projects with high interdependencies and resource constraints. Despite these limitations, deterministic frameworks remain relevant as baseline models for benchmarking more advanced scheduling approaches. Their continued use in practice underscores the importance of structured planning while also highlighting the need for more adaptive methodologies that can address uncertainty and dynamic conditions in engineering systems.

Variance and sensitivity analysis play a crucial role in evaluating the robustness of deterministic scheduling frameworks (Beatrice Onyinyechi, 2023; Lang et al., 2020). Within CPM and PERT-based models, researchers have extensively examined how variations in activity durations influence overall project performance. Empirical findings indicate that even minor fluctuations in key activities can lead to significant changes in project completion time, particularly when those activities lie on or near the critical path. Studies focusing on infrastructure and construction projects demonstrate that sensitivity analysis helps identify high-risk activities, enabling targeted risk mitigation strategies. Quantitative assessments across multiple case studies reveal that projects with tightly constrained schedules exhibit higher sensitivity to delays, emphasizing the importance of accurate duration estimation. Further research has explored the application of variance analysis to improve decision-making in deterministic models. By analyzing historical project data, researchers have identified patterns of variability that can inform more realistic scheduling assumptions (Siddique & Aditya, 2023; Pan et al., 2023). Comparative studies indicate that incorporating variance-based adjustments can enhance the predictive accuracy of deterministic schedules, although such enhancements remain limited in their ability to fully address uncertainty. Statistical evaluations show that deterministic models often rely on simplified assumptions regarding variability, which can lead to discrepancies between planned and actual project outcomes. This limitation becomes particularly evident in complex engineering systems where multiple sources of uncertainty interact simultaneously. Despite these challenges, variance and sensitivity analysis continue to provide valuable insights into the behavior of deterministic scheduling models. Their application supports the identification of critical risk factors and contributes to improved planning and resource allocation. The body of literature underscores the importance of integrating analytical techniques into scheduling frameworks to enhance their reliability, while also highlighting the need for more advanced models capable of capturing dynamic and stochastic elements in project environments (Alexopoulos et al., 2024; Md Siam & Md. Sultan, 2023).

Figure 3: Engineering Scheduling Risk Analysis Framework



Probabilistic scheduling models have emerged as a response to the limitations of deterministic approaches, offering a more realistic representation of uncertainty in project environments. These models incorporate stochastic elements into task duration estimation, allowing for the consideration of variability and risk in scheduling decisions (Ashfaq & Manam, 2023; Ulusoy & Hazır, 2021). Research across engineering disciplines has demonstrated that probabilistic methods provide a more comprehensive understanding of project timelines by accounting for the likelihood of different outcomes. Studies focusing on construction and manufacturing projects show that probabilistic models improve the accuracy of schedule predictions, particularly in environments characterized by high uncertainty and complex task interdependencies.

The stochastic modeling of task durations is a central component of probabilistic scheduling. Researchers have employed various statistical techniques to represent uncertainty, including the use of probability distributions based on historical data. Empirical analyses indicate that the selection of appropriate distributions significantly influences the reliability of scheduling outcomes (Mainuddin & Chandra, 2023; Pregina & Kannan, 2022). Comparative studies reveal that probabilistic models outperform deterministic approaches in scenarios involving fluctuating resource availability and unpredictable external factors. Quantitative evaluations across multiple datasets highlight improvements in risk assessment and decision-making, as probabilistic models enable the estimation of confidence intervals for project completion times. Despite their advantages, probabilistic scheduling models also present challenges related to data requirements and computational complexity. Studies have shown that accurate stochastic modeling depends on the availability of high-quality data, which may not always be accessible in practice. Additionally, the increased computational demands associated with probabilistic analysis can limit their applicability in large-scale projects (Al Nasseri et al., 2016; Robel & Aminul, 2023). Nevertheless, the literature consistently emphasizes the value of probabilistic approaches in enhancing the realism and reliability of project scheduling, particularly in complex engineering systems where uncertainty plays a significant role.

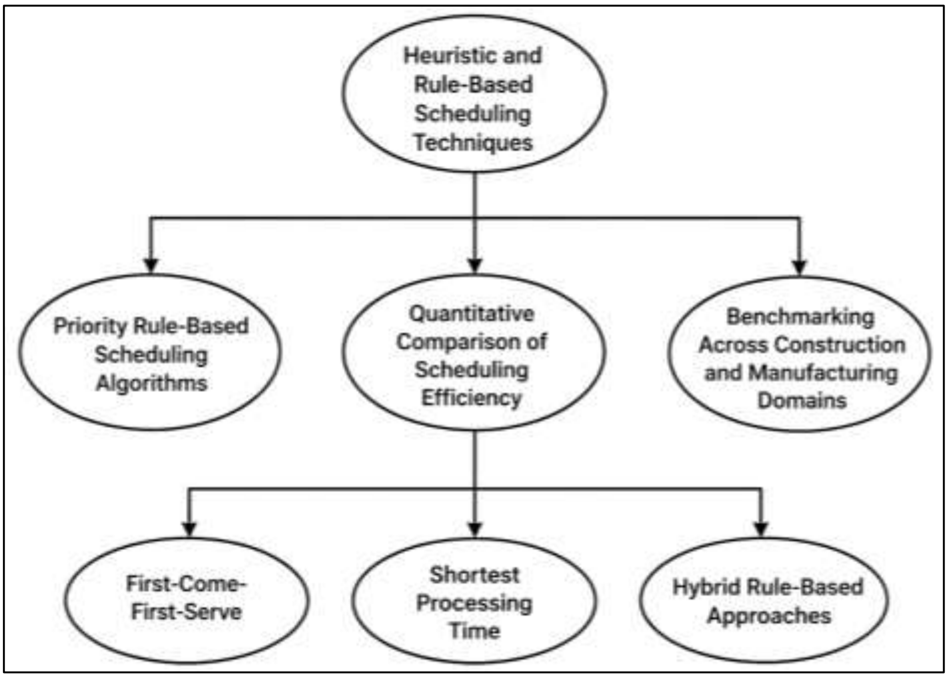
Monte Carlo simulation has become a widely used technique for analyzing uncertainty and risk in project scheduling. By generating multiple scenarios based on probabilistic inputs, this approach allows researchers and practitioners to evaluate the potential variability in project outcomes. Studies across engineering domains have demonstrated that Monte Carlo simulation provides valuable insights into the distribution of project completion times, enabling more informed decision-making. Quantitative analyses indicate that this method enhances the ability to identify high-risk activities and assess the likelihood of delays, contributing to more effective risk management strategies. The application of Monte Carlo simulation in scheduling is often complemented by statistical distribution fitting and variance estimation. Researchers have explored various methods for selecting appropriate distributions to represent task durations, including normal, triangular, and beta distributions (Ammar & Abd-ElKhalek, 2022; Sazzadul, 2023). Empirical findings suggest that the choice of distribution has a significant impact on simulation results, influencing both the accuracy and reliability of risk assessments. Comparative studies across different project types reveal that tailored distribution selection based on historical data improves the predictive performance of simulation models. Statistical evaluations further highlight the importance of variance estimation in understanding the spread and uncertainty of scheduling outcomes. While Monte Carlo simulation offers substantial advantages, the literature also identifies limitations related to computational requirements and model complexity. Large-scale simulations can be resource-intensive, particularly when dealing with high-dimensional scheduling problems. Additionally, the accuracy of simulation results depends on the quality of input data and the assumptions underlying the model. Despite these challenges, the integration of Monte Carlo techniques with probabilistic scheduling frameworks has significantly advanced the field of project scheduling (Albert & Rashedul, 2024; Lu et al., 2017). The body of research underscores the effectiveness of simulation-based approaches in capturing uncertainty and improving the robustness of scheduling decisions in engineering systems.

Empirical Performance Evaluation of Heuristic and Rule-Based Scheduling Techniques

Priority rule-based scheduling algorithms represent one of the most widely studied heuristic approaches for managing task sequencing in engineering systems. These methods rely on predefined decision rules to determine the order in which jobs are processed, offering simplicity and computational efficiency compared to optimization-based techniques. Among the most commonly applied rules are First-Come-First-Serve and Shortest Processing Time, both of which have been extensively evaluated in empirical studies across manufacturing and service systems (Barut et al., 2024). Research indicates that First-Come-First-Serve provides fairness and ease of implementation, particularly in environments where job arrival times are critical. In contrast, Shortest Processing Time has been shown to minimize average completion time and improve overall system efficiency in high-throughput environments. Comparative analyses across multiple datasets demonstrate that rule-based methods can achieve near-optimal performance under specific conditions, especially when system variability is limited.

Empirical evidence from industrial case studies highlights that the effectiveness of these rules depends heavily on system characteristics, including job variability, machine availability, and queue dynamics (Istiaq & Hasan Or, 2024; Woo et al., 2017). Studies examining job-shop and flow-shop environments reveal that Shortest Processing Time consistently outperforms other rules in reducing waiting times and improving throughput, while First-Come-First-Serve remains advantageous in maintaining system stability and predictability.Quantitative evaluations also show that hybrid rule-based approaches, which combine multiple priority rules, can enhance scheduling performance by balancing efficiency and fairness. Despite their simplicity, these algorithms continue to play a significant role in practical scheduling applications due to their low computational requirements and ease of integration into existing systems (Md. Jobayer Ibne & Aditya, 2024; Oliveira Farias et al., 2021)Simulation-based studies have provided extensive datasets for evaluating rule performance, enabling researchers to identify patterns and correlations between system parameters and scheduling outcomes. Quantitative findings suggest that adaptive rule selection mechanisms, which dynamically adjust priority rules based on system conditions, can achieve superior performance compared to static approaches. Simulation-based studies have provided extensive datasets for evaluating rule performance, enabling researchers to identify patterns and correlations between system parameters and scheduling outcomes. Quantitative findings suggest that adaptive rule selection mechanisms, which dynamically adjust priority rules based on system conditions, can achieve superior performance compared to static approaches.

Figure 4: Heuristic Scheduling Evaluation Framework



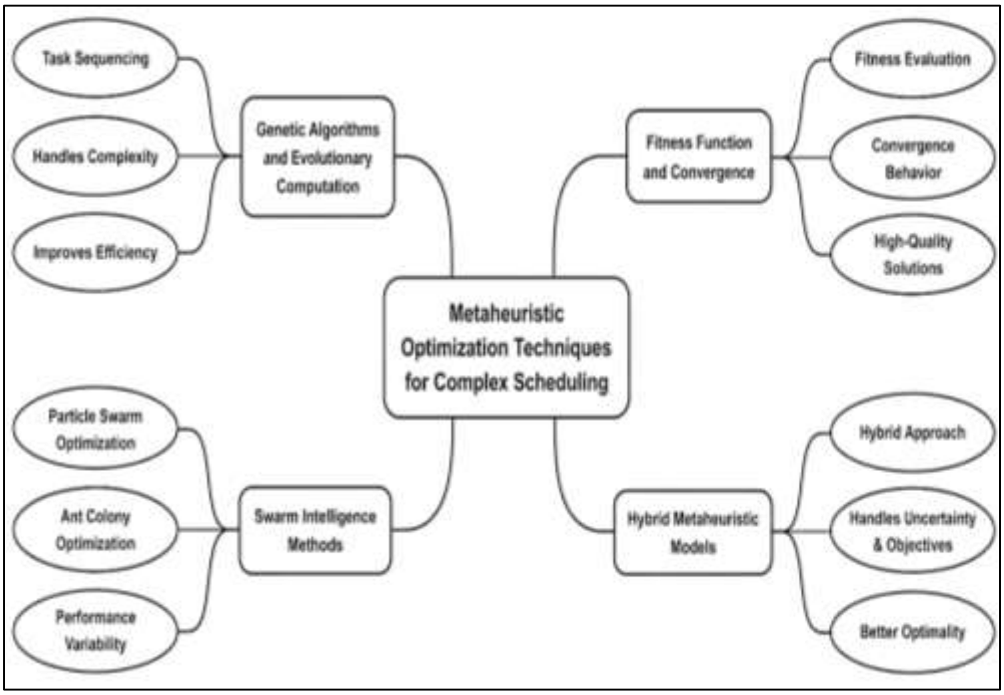
These results underscore the importance of flexibility in heuristic scheduling, particularly in environments characterized by variability and uncertainty (López-Santana et al., 2018b). The literature also highlights the trade-offs associated with different performance metrics. For example, a rule that minimizes makespan may not necessarily minimize tardiness, leading to conflicting objectives in scheduling decisions. Empirical studies emphasize the need for multi-objective evaluation frameworks to capture these trade-offs and provide a more comprehensive assessment of scheduling efficiency. Overall, the quantitative comparison of heuristic rules provides valuable insights into their strengths and limitations, informing their application in diverse engineering contexts (Chakrabortty et al., 2016). Benchmarking studies have played a crucial role in evaluating the applicability of heuristic scheduling techniques across different engineering domains. Construction and manufacturing systems, in particular, have been extensively analyzed due to their distinct operational characteristics. Research in construction scheduling often focuses on project-based environments with complex task dependencies and resource constraints, while manufacturing studies typically examine repetitive processes with high volumes of tasks. Comparative analyses reveal that heuristic rules perform differently across these domains, reflecting variations in system structure and operational requirements. Empirical findings indicate that in manufacturing environments, where tasks are relatively standardized and processing times are more predictable, heuristic rules such as Shortest Processing Time achieve significant improvements in throughput and efficiency. In contrast, construction projects, which involve greater uncertainty and interdependencies, require more flexible scheduling approaches (Satic et al., 2022). Studies comparing datasets from both domains show that rule-based methods can still provide valuable baseline solutions in construction, although their performance may be limited by the complexity of project networks. Quantitative benchmarking across multiple case studies highlights the importance of domain-specific adaptations to enhance the effectiveness of heuristic scheduling. The literature also emphasizes the role of benchmarking in validating scheduling algorithms. By using standardized datasets and performance metrics, researchers can compare the effectiveness of different approaches and identify best practices. Cross-domain studies demonstrate that while heuristic rules offer consistent benefits in terms of computational efficiency, their performance is highly dependent on the nature of the scheduling problem (Becchi et al., 2024). These findings reinforce the need for tailored scheduling strategies that consider the unique .

Metaheuristic Optimization Techniques for Complex Scheduling Problems

Genetic algorithms and evolutionary computation have received extensive attention in scheduling literature because they provide flexible search mechanisms for complex engineering problems involving multiple constraints, competing objectives, and large solution spaces. In project scheduling, these approaches are commonly used to generate optimized task sequences by evaluating alternative schedules through fitness-based selection. The literature shows that genetic algorithms are particularly effective when scheduling problems include nonlinear dependencies, limited resources, uncertain activity durations, and multiple performance targets. Studies have demonstrated that evolutionary computation can outperform deterministic scheduling models in complex environments because it does not rely on fixed activity sequences or rigid assumptions (Choi, 2016). Instead, it explores a wide range of possible schedules and gradually improves solution quality through iterative selection, crossover, and mutation processes. Quantitative studies have also shown that genetic algorithms are capable of reducing makespan, improving resource balance, and minimizing delay penalties in project-based and manufacturing systems. Their strength lies in their ability to search globally rather than becoming limited to one predefined path. Research comparing genetic algorithms with traditional scheduling frameworks indicates that evolutionary methods are more suitable for large-scale scheduling tasks where multiple feasible alternatives exist and where optimal solutions are difficult to identify through conventional models (Dorokhova et al., 2020).

The effectiveness of genetic algorithms in scheduling depends strongly on the design of the fitness function, which determines how candidate schedules are evaluated and ranked. Literature on evolutionary scheduling emphasizes that fitness functions often include quantitative criteria such as makespan, tardiness, resource utilization, cost deviation, and schedule stability. Studies suggest that well-designed fitness functions improve the ability of genetic algorithms to identify high-quality schedules, especially in multi-objective project environments. In engineering systems, fitness evaluation becomes more complex because schedules must satisfy precedence constraints, resource limits, and deadline requirements simultaneously (Prity et al., 2024). Quantitative research has shown that convergence rates vary based on population size, mutation rate, crossover strategy, and stopping criteria

Figure 5: Metaheuristic Scheduling Optimization Framework



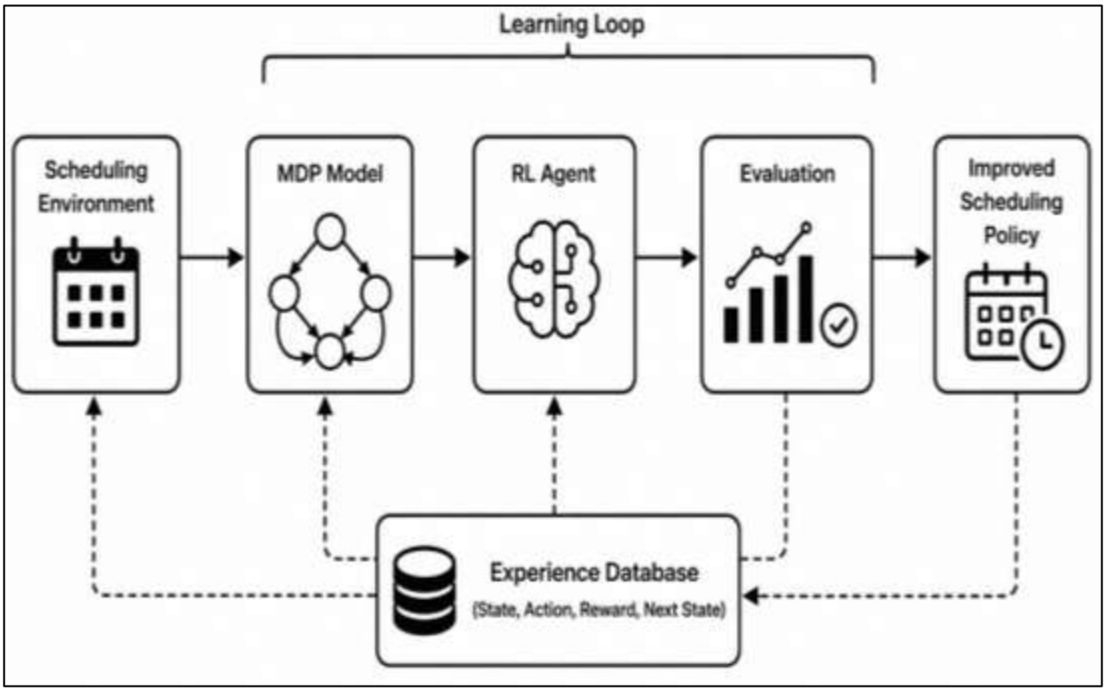
Quantitative studies indicate that swarm intelligence methods often achieve better makespan reduction and resource allocation performance than deterministic and basic heuristic approaches. However, their performance varies significantly across problem sizes. In smaller scheduling problems, these methods can quickly identify near-optimal solutions, while in larger problems, search efficiency may decline due to increased complexity and larger solution spaces. Studies comparing Particle Swarm Optimization and Ant Colony Optimization show that parameter tuning, population structure, and problem representation strongly influence results. The exploration-exploitation balance is especially important because insufficient exploration may cause stagnation, while excessive exploration may increase computational time (Hossein Nia Shavaki & Jolai, 2021). Overall, swarm intelligence literature supports the value of these methods for complex scheduling, while also showing that their effectiveness depends on careful calibration and problem-specific adaptation. . Some studies report faster convergence when adaptive parameter control is used, while others show that overly rapid convergence may reduce solution diversity and lead to premature optimization. Compared with deterministic models, genetic algorithms often require greater computational effort, yet they usually produce stronger solutions in complex and uncertain scheduling scenarios. The literature also highlights that convergence performance is influenced by problem size, as larger project networks require broader exploration and more iterations to reach stable solutions. Overall, evolutionary scheduling research demonstrates that fitness function formulation and convergence control are central to achieving reliable quantitative improvements (Satic et al., 2019).

Hybrid metaheuristic models have been developed to overcome the limitations of single-method optimization techniques by combining complementary strengths from different algorithms, simulation models, and mathematical optimization procedures. In complex scheduling literature, hybrid models often integrate genetic algorithms with local search, Particle Swarm Optimization with simulation, Ant Colony Optimization with constraint handling, or evolutionary computation with reinforcement learning mechanisms (Manupati et al., 2016). Studies indicate that hybrid approaches frequently produce stronger quantitative results than standalone metaheuristics because they improve both global exploration and local refinement. In engineering scheduling, hybrid models are particularly useful when problems involve uncertain task durations, dynamic resource availability, and multiple performance objectives. Simulation-based hybrid models allow researchers to test schedules under variable conditions, while optimization components refine task sequencing and resource allocation. Quantitative findings across construction, manufacturing, logistics, and infrastructure scheduling studies show improvements in solution optimality, reduced makespan, lower tardiness, and better resource utilization (J. Zhou et al., 2024). Hybrid models also help address computational limitations by narrowing search spaces and improving convergence reliability. Compared with deterministic scheduling methods, hybrid metaheuristics provide more adaptive and robust solutions for large-scale and uncertain scheduling environments. The literature consistently suggests that integration between simulation and optimization enhances decision quality because it captures both structural scheduling constraints and real-world variability. Consequently, hybrid metaheuristic models represent a major advancement in complex scheduling research, offering measurable gains in efficiency, robustness, and solution accuracy (Xiong et al., 2017).

Mathematical Foundations and Quantitative Modeling of Reinforcement Learning

Markov Decision Process formulation provides the main structural foundation for applying reinforcement learning to quantitative scheduling problems in engineering systems. In scheduling research, the project environment is commonly represented as a sequence of decision states in which an agent observes the current condition of tasks, resources, delays, and constraints before selecting an action. This structure is especially important in dynamic project scheduling because each decision influences the next scheduling condition, making the problem sequential rather than isolated.

Figure 6: Reinforcement Learning Scheduling Framework



Literature on reinforcement learning shows that MDP-based modeling allows scheduling problems to be treated as adaptive decision-making processes rather than static optimization tasks (López-Santana et al., 2018a). Studies in manufacturing, construction, logistics, and resource-constrained project scheduling have used MDP concepts to represent task progress, machine availability, worker allocation, activity precedence, and disruption status. The strength of this approach lies in its ability to connect scheduling decisions with measurable outcomes over time. Compared with deterministic scheduling models, MDP-based frameworks provide a more flexible way to account for uncertainty, changing project conditions, and repeated decision cycles. Quantitative studies also show that MDP formulation improves the analytical clarity of scheduling models by defining how system states change after each scheduling action. This makes it easier to evaluate performance based on accumulated project outcomes such as reduced delay, improved resource use, and shorter completion time (Sahu et al., 2023).

State-space dimensionality is a major concern in reinforcement learning-based scheduling because engineering systems often contain many interacting variables. In project scheduling, the state may include task completion status, remaining durations, resource capacity, activity dependencies, machine conditions, cost levels, and deadline pressure. As the number of tasks and resources increases, the state-space becomes larger and more complex, making learning more difficult. Literature indicates that classical reinforcement learning performs well in small or moderately sized scheduling problems but faces scalability challenges when the number of possible states grows rapidly. Transition behavior is also central because scheduling decisions do not occur in isolation; each action changes the project environment and affects later decisions (Pareigis, 2023). Studies using reinforcement learning for job-shop scheduling, construction planning, and production control show that transition uncertainty can arise from task delays, machine breakdowns, resource conflicts, and demand variation. Quantitative modeling of transitions helps researchers estimate how likely the system is to move from one scheduling condition to another after a decision is made. This is important for evaluating the reliability and adaptability of scheduling policies. The literature also highlights that state representation strongly affects learning efficiency (Madabhushi & Lee, 2016). Poorly designed state structures may slow convergence or produce unstable policies, while compact and meaningful representations improve learning performance and scheduling accuracy.

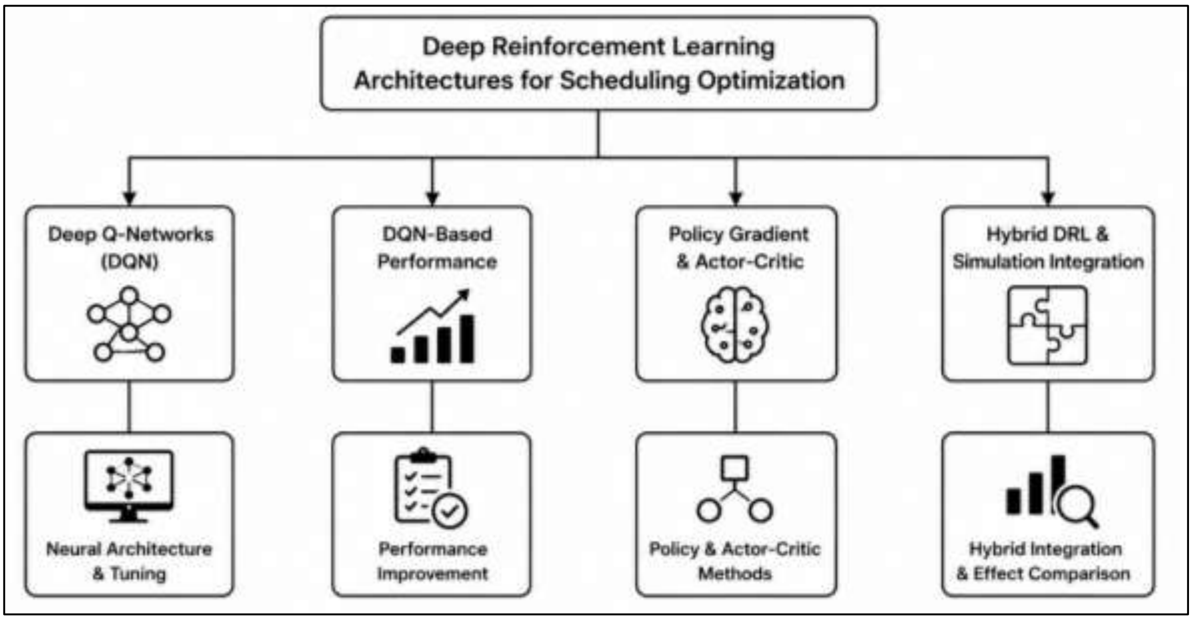
Deep Reinforcement Learning Architectures for Scheduling Optimization

Deep Q-Networks have become a major architectural foundation in deep reinforcement learning-based scheduling because they extend classical Q-learning through neural network approximation. In complex engineering scheduling problems, the number of possible project states and scheduling actions can become too large for traditional reinforcement learning methods to manage efficiently. Deep Q-Networks address this limitation by using neural networks to estimate decision values across large state spaces, allowing scheduling agents to learn from complex patterns in task progress, resource availability, machine status, delay conditions, and project constraints (Zhang et al., 2017). Literature on scheduling optimization shows that DQN-based methods have been applied extensively in job-shop scheduling, flexible manufacturing scheduling, cloud workflow scheduling, construction project sequencing, and resource-constrained project scheduling. These studies commonly report improvements in makespan, tardiness, throughput, and resource utilization when DQN models are compared with dispatching rules, deterministic scheduling models, and classical reinforcement learning approaches. Neural network design plays an important role in these results because the number of layers, activation functions, memory replay strategies, target network updates, and training episodes directly influence learning stability (Zhang et al., 2017). Hyperparameter tuning is also repeatedly emphasized in the literature, as learning rate, discount factor, batch size, and exploration strategy affect convergence speed and final scheduling quality. Studies using DQN variants, including Double DQN, Dueling DQN, and Prioritized Experience Replay, generally show stronger stability and improved policy accuracy compared with basic DQN structures. Overall, the literature positions DQN architectures as a significant advancement for discrete scheduling environments where decisions involve selecting tasks, machines, resources, or priority sequences (Jayanetti et al., 2022).

Quantitative research on DQN-based scheduling models consistently evaluates performance through measurable scheduling indicators such as makespan reduction, average tardiness, machine idle time, job completion rate, reward convergence, and computational efficiency. Empirical studies in manufacturing and engineering systems show that DQN-based schedulers often outperform fixed dispatching rules because they learn adaptive decision policies rather than applying the same rule across all system conditions. In job-shop and flow-shop environments, DQN models have been shown to reduce waiting time and improve sequencing efficiency by learning which job should be processed under changing machine availability and workload pressure (Jalali Khalil Abadi et al., 2024). In project scheduling contexts, DQN-based models have demonstrated value in handling precedence constraints and dynamic task priorities, particularly when the scheduling environment changes during execution. Comparative statistical evaluations often show that DQN models perform better than traditional heuristic methods when system uncertainty increases. However, the magnitude of improvement varies across datasets, problem scales, training conditions, and reward structures. Some studies report substantial gains in solution quality when the state representation captures detailed operational information, while others show modest improvement when the model lacks sufficient environmental features (Gu et al., 2024). The literature also identifies training instability as a quantitative concern, especially in larger scheduling problems where sparse rewards and complex state transitions can delay convergence. DQN variants have been used to address these issues by improving value estimation, reducing overestimation bias, and strengthening learning consistency. The accumulated evidence suggests that DQN-based scheduling provides measurable performance gains when model design, reward structure, and hyperparameter calibration are carefully aligned with the scheduling problem.

Policy Gradient and Actor-Critic methods have gained attention in scheduling optimization because they are more suitable for environments involving continuous or high-dimensional decision spaces. While DQN models are commonly applied to discrete action selection, many engineering scheduling problems require more flexible decision-making, such as adjusting resource allocation levels, controlling processing priorities, modifying start times, or balancing multiple project constraints simultaneously (Ibrahim & Askar, 2023). Policy Gradient methods directly optimize the decision policy by learning which actions produce stronger long-term scheduling performance.

Figure 7: DRL Scheduling Optimization Architectures



Actor-Critic methods combine policy learning with value estimation, allowing the actor component to select scheduling actions while the critic evaluates the quality of those decisions. Literature on production scheduling, resource allocation, logistics coordination, and dynamic project control shows that Actor-Critic architectures can improve adaptability in complex environments where scheduling decisions are not limited to simple job selection. Quantitative evaluations often compare these methods with DQN, classical reinforcement learning, metaheuristics, and dispatching rules. Studies suggest that Actor-Critic methods can produce more stable learning in continuous control environments, especially when scheduling requires repeated adjustment rather than one-time sequencing (Rjoub et al., 2021). Algorithms such as Advantage Actor-Critic, Deep Deterministic Policy Gradient, Proximal Policy Optimization, and Soft Actor-Critic have been examined in scheduling-related studies because they support different forms of policy optimization and stability control. Comparative findings indicate that these methods can improve resource utilization, reduce delay penalties, and enhance operational responsiveness. However, their performance depends heavily on reward design, training sample efficiency, and exploration control. The literature therefore treats Policy Gradient and Actor-Critic methods as essential DRL architectures for scheduling problems that require flexible, adaptive, and continuous decision-making (Fan et al., 2022).

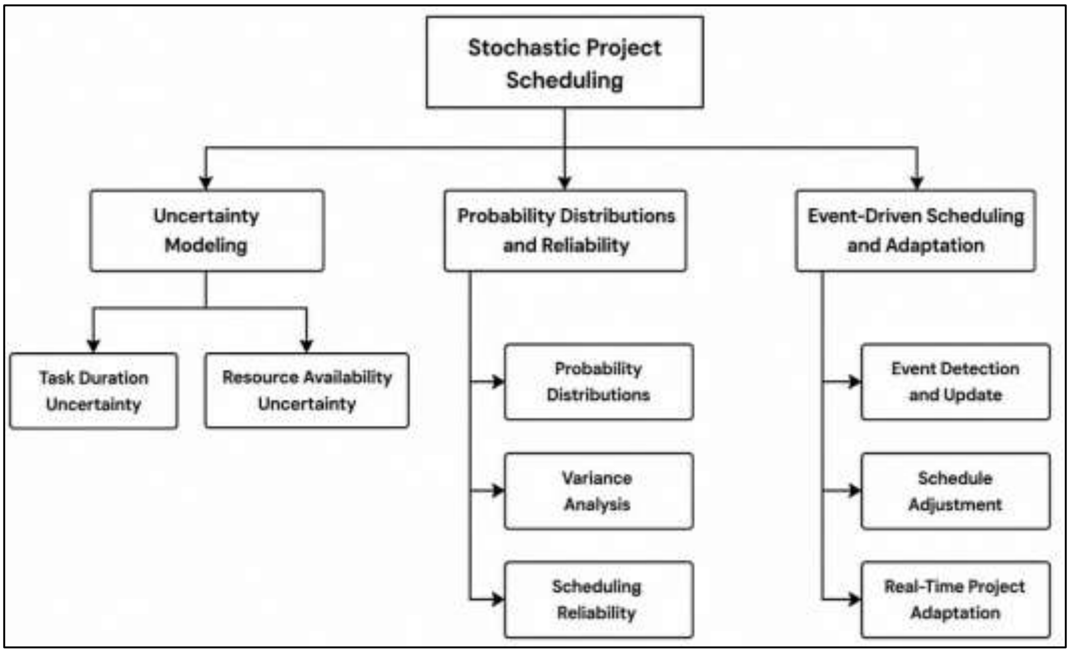
Hybrid deep reinforcement learning models represent an important direction in scheduling optimization literature because they combine DRL with metaheuristics, simulation, mathematical optimization, or domain-specific dispatching rules. These hybrid models are designed to overcome limitations found in standalone DRL systems, including slow training, unstable convergence, large exploration spaces, and sensitivity to reward design. In engineering scheduling studies, DRL has been integrated with genetic algorithms, particle swarm optimization, ant colony optimization, discrete-event simulation, digital twins, and constraint-based optimization to improve solution quality and practical reliability. Simulation integration is especially important because it allows scheduling agents to learn from controlled representations of project or production environments before evaluation against benchmark scenarios (Waschneck et al., 2018). Literature shows that hybrid DRL models often achieve stronger quantitative performance than single-method approaches because metaheuristics improve global search, simulation captures system uncertainty, and DRL supports adaptive sequential decision-making. Effect size comparisons across scheduling studies generally focus on measurable differences in makespan, tardiness, throughput, cost reduction, resource balance, and policy robustness. Research comparing hybrid DRL with traditional heuristics and standalone metaheuristics frequently reports stronger gains in complex or dynamic environments, particularly when disruptions, resource conflicts, and changing job arrivals are present. However, reported effect sizes vary due to differences in datasets, baseline methods, experimental design, and performance indicators (Liu et al., 2023). Some studies show large improvements when DRL is combined with simulation-based learning, while others show smaller gains when benchmark problems are simple or highly structured. Overall, the literature supports hybrid DRL models as quantitatively strong scheduling frameworks because they combine learning adaptability with optimization depth and simulation-based realism.

Quantitative Modeling of Dynamic and Stochastic Project Scheduling

Uncertainty modeling is a central element of dynamic and stochastic project scheduling because engineering projects rarely operate under fixed and fully predictable conditions. In complex engineering systems, task durations may vary due to labor productivity, equipment conditions, material delivery, design changes, weather interruptions, and coordination delays (Mangalampalli et al., 2024). Resource availability may also fluctuate because of machine breakdowns, workforce shortages, supply chain instability, or competing project demands. Literature on stochastic scheduling emphasizes that these uncertainties directly affect schedule reliability, especially when project activities are strongly connected through precedence relationships. Quantitative studies show that models incorporating uncertainty provide more realistic schedule estimates than deterministic models because they account for variation rather than assuming fixed activity durations. Probability-based representations allow researchers to examine a wider range of possible project outcomes, including early completion, expected completion, and delay-prone scenarios. Variance analysis is also important because it identifies which tasks or resources contribute most to schedule instability (G. Zhou et al., 2024). Studies across construction, manufacturing, infrastructure, and production systems indicate that high-variance activities often create disproportionate effects on total project duration. As a result, uncertainty modeling helps explain why planned schedules frequently differ from actual project performance and why adaptive scheduling methods are necessary in engineering environments.

Probability distributions and variance analysis are widely used in stochastic project scheduling to represent the uncertain behavior of task durations and resources. Instead of treating activity time as a single fixed value, researchers use statistical distributions to describe possible duration ranges and their likelihood (Ran et al., 2019). This approach enables scheduling models to capture variability in work packages, resource performance, and environmental conditions. In engineering project literature, probability-based scheduling has been applied to construction projects, production systems, maintenance operations, transportation networks, and energy infrastructure planning. Quantitative findings show that reliability improves when schedules are evaluated under multiple possible conditions rather than one assumed baseline

Figure 8: Dynamic Stochastic Scheduling Framework

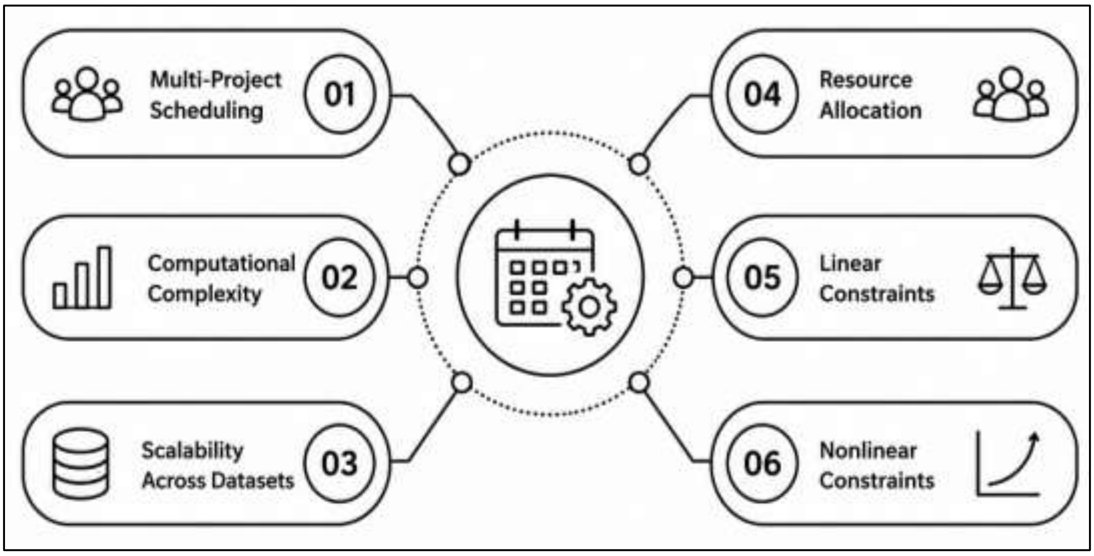


Variance analysis further strengthens this process by showing how much uncertainty exists around each activity and how strongly that uncertainty influences the overall schedule. Studies indicate that activities with high duration variance can reduce schedule reliability even when their average expected duration appears manageable. Similarly, uncertain resource availability can produce idle time, task interruption, and delayed handoffs between dependent activities (Liu et al., 2022). Scheduling reliability is therefore not only determined by average task duration but also by the spread and instability of project variables. The literature consistently shows that stochastic models provide stronger insight into schedule risk because they measure uncertainty directly and support more accurate comparisons between planned and actual outcomes.Event-driven scheduling systems focus on adjusting schedules when specific changes or disruptions occur during project execution. These events may include task delays, equipment failures, material shortages, urgent priority changes, weather interruptions, inspection failures, or unexpected resource conflicts. Literature on dynamic project scheduling shows that event-driven models are more responsive than static scheduling approaches because they update decisions based on current project conditions. In engineering systems, this responsiveness is essential because disruptions can quickly spread through task networks and affect completion time, resource utilization, and cost performance (Wang et al., 2021). Quantitative studies demonstrate that adaptive scheduling frameworks can reduce delay propagation by modifying task sequences, reallocating resources, or revising activity priorities after disruptive events occur. Real-time scheduling systems often rely on updated project data from monitoring tools, production systems, sensors, enterprise platforms, or simulation environments. These systems allow scheduling decisions to reflect actual progress rather than outdated planning assumptions. Research across construction management, manufacturing control, logistics scheduling, and maintenance planning indicates that real-time adaptation improves operational stability by reducing waiting time, idle resources, and schedule deviation. The literature also highlights that event-driven scheduling is particularly useful in high-uncertainty environments where frequent changes make fixed schedules unreliable (Yan et al., 2022). Through continuous adjustment, adaptive frameworks support stronger coordination between planning and execution.

Multi-Project, Multi-Resource, and High-Dimensional Scheduling Complexity

Large-scale scheduling optimization has become a major focus in engineering management literature because modern organizations often execute multiple projects simultaneously while sharing limited resources across different operational units. In multi-project environments, scheduling complexity increases when activities from separate projects compete for the same labor, equipment, materials, budgets, and technical expertise. Research on large-scale scheduling shows that traditional single-project planning models are often insufficient because they do not fully capture cross-project dependencies, resource conflicts, and priority changes (Liu et al., 2020). Studies in construction, manufacturing, infrastructure development, software engineering, and maintenance planning indicate that multi-project scheduling requires coordinated decision-making across several interconnected schedules rather than isolated optimization of one project timeline. Quantitative evaluations commonly examine makespan, total project delay, resource utilization, cost deviation, and schedule stability as core performance indicators. The literature emphasizes that large-scale scheduling problems become computationally difficult as the number of activities, constraints, and project interactions increases (Demirel et al., 2018). This complexity is especially visible when projects contain uncertain task durations, shared resource pools, and changing priority levels. Empirical studies show that optimization quality depends on both the size of the scheduling dataset and the ability of the method to manage dependency structures efficiently. As a result, large-scale scheduling research increasingly focuses on scalable algorithms capable of producing reliable schedules within practical computational limits.

Figure 9: Multi-Project Scheduling Optimization Framework



Computational complexity analysis is essential in high-dimensional scheduling research because engineering scheduling problems often expand rapidly with each additional task, resource, constraint, and project dependency. Literature on scheduling optimization shows that many project scheduling problems are computationally intensive because the number of feasible schedules grows sharply as project size increases (Sun et al., 2019). This creates challenges for exact optimization methods, which may perform well on small datasets but become inefficient when applied to large-scale or multi-project cases . Quantitative studies comparing deterministic optimization, heuristics, metaheuristics, and learning-based methods consistently show that scalability is a central measure of algorithmic usefulness. In benchmark datasets, researchers often evaluate scalability by examining computation time, solution quality, convergence behavior, and performance consistency across small, medium, and large problem instances. Findings from construction and manufacturing studies suggest that scalable scheduling methods must balance accuracy with computational speed. A method that produces a highly optimized solution may have limited practical value when processing time is excessive. Conversely, a fast method may be insufficient when it produces weak schedules under complex constraints. The literature also shows that dataset structure affects scalability. Scheduling problems with dense precedence relationships, limited resources, and multiple objectives are more difficult than problems with simple task sequences (Z.-J. Wang et al., 2019). Therefore, scalability across datasets is not only a matter of problem size but also of constraint density, uncertainty level, and resource competition.

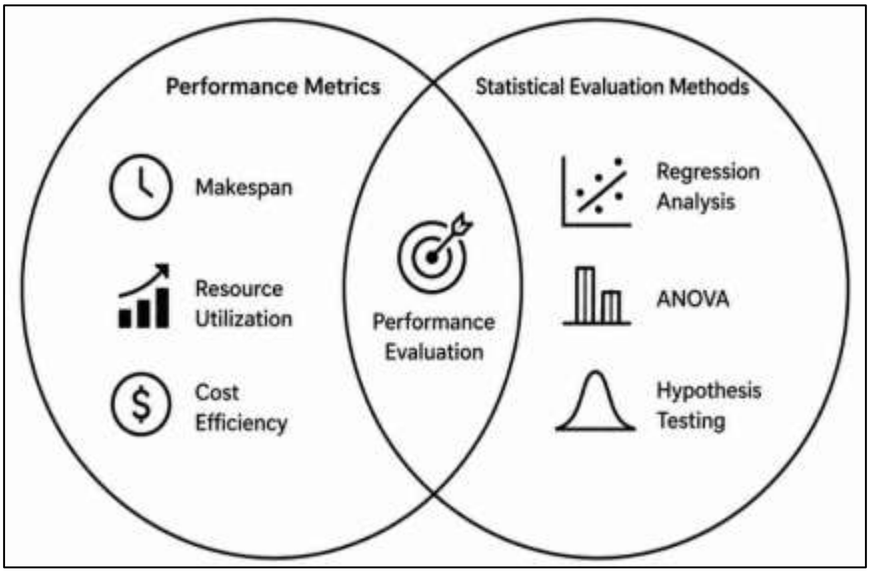
Resource allocation and constraint optimization are central themes in multi-resource scheduling because engineering projects depend on the effective distribution of labor, equipment, materials, capital, and technical capacity. Literature in resource-constrained project scheduling shows that inefficient resource allocation can lead to idle time, bottlenecks, task delays, cost overruns, and reduced throughput. Quantitative studies across construction, production, logistics, and infrastructure systems demonstrate that scheduling performance improves when resource constraints are modeled explicitly rather than treated as secondary planning issues. Resource allocation decisions must account for availability, capacity, task priority, skill compatibility, equipment usage, and project deadlines (Schlenkrich & Parragh, 2023). In multi-project settings, these decisions become more complex because one resource may be required by several projects at the same time. Research indicates that constraint optimization helps identify feasible schedules that satisfy project dependencies while improving resource efficiency. Linear constraint models are often used when relationships between tasks and resources are relatively simple, predictable, and proportional. Nonlinear constraint models are more suitable when scheduling relationships involve complex interactions, variable productivity, learning effects, congestion, or dynamic resource performance. Studies comparing these approaches show that nonlinear models can represent real engineering conditions more accurately, although they usually require greater computational effort (Y. Wang et al., 2019). Overall, the literature emphasizes that effective resource allocation depends on selecting modeling approaches that match the complexity of the project environment.

Quantitative Performance Metrics and Statistical Evaluation Methods

Key performance indicators are central to quantitative scheduling research because they provide measurable criteria for comparing the effectiveness of different scheduling methods. In engineering systems, makespan reduction is one of the most frequently used indicators because it reflects the total time required to complete a set of project activities, jobs, or operations. Literature on project scheduling, manufacturing control, construction planning, and resource-constrained optimization consistently treats makespan as a core measure of temporal efficiency (Shi et al., 2023). Studies comparing deterministic models, heuristics, metaheuristics, reinforcement learning, and deep reinforcement learning show that stronger scheduling approaches generally produce shorter completion times while maintaining feasibility under task dependency and resource constraints. Resource utilization rate is another major indicator because engineering schedules must not only finish quickly but also use labor, equipment, machines, and materials efficiently. Quantitative studies show that poor resource utilization leads to idle capacity, bottlenecks, and uneven workload distribution. Cost efficiency is also widely used because time savings do not always translate into economic improvement when schedules require excessive resource use, overtime, or computational effort (Chen et al., 2021). The literature therefore emphasizes that scheduling performance should be assessed through multiple KPIs rather than a single measure. Across empirical studies, makespan, resource utilization, and cost efficiency are commonly integrated to evaluate whether a scheduling model improves operational performance in a balanced and measurable way.

The literature shows that makespan reduction, resource utilization, and cost efficiency are evaluated differently across engineering domains because scheduling environments vary in structure and operational goals. In manufacturing systems, makespan and throughput are often prioritized because production lines depend on fast job completion, reduced machine idle time, and stable workflow. In construction and infrastructure projects, schedule reliability and cost control receive greater attention because delays can affect subcontractor coordination, equipment rental, procurement timing, and contractual performance (Rjoub et al., 2021). In logistics and maintenance scheduling, resource utilization becomes especially important because vehicles, crews, machines, and service windows must be coordinated under strict operational constraints.

Figure 10: Scheduling Performance Evaluation Framework



Quantitative comparisons across these domains show that scheduling algorithms may perform well on one KPI while producing weaker results on another. For example, a model may reduce total completion time but increase cost due to higher resource intensity. Another model may improve resource balance but extend project duration. This trade-off has led many researchers to use multi-objective evaluation frameworks, where scheduling quality is judged through combined performance indicators. Literature also shows that DRL-based and metaheuristic approaches are often evaluated against deterministic and rule-based baselines to determine whether observed improvements are statistically meaningful (Chen et al., 2020). Overall, domain-specific KPI interpretation is essential for understanding how scheduling methods perform under practical engineering conditions.Statistical evaluation methods strengthen scheduling research by determining whether observed performance differences between scheduling models are meaningful, reliable, and generalizable. Regression analysis is frequently used to examine relationships between scheduling performance and explanatory variables such as project size, task variability, resource scarcity, algorithm parameters, or uncertainty level. ANOVA is commonly applied when researchers compare multiple scheduling algorithms across benchmark datasets or experimental conditions (Shao et al., 2024). Hypothesis testing supports formal comparison between baseline and proposed models by evaluating whether differences in makespan, tardiness, cost, or utilization are likely to reflect real performance variation rather than random fluctuation. Confidence intervals add further value by showing the estimated range within which performance effects are expected to fall. Effect sizes are especially important because statistical significance alone does not reveal the practical magnitude of improvement. Scheduling studies increasingly use effect-size reporting to show whether an algorithm produces small, moderate, or large improvements over comparison methods (Hafsi et al., 2022). Literature across project scheduling and production optimization emphasizes that robust statistical evaluation requires repeated experiments, multiple datasets, controlled baselines, and transparent reporting of variability. Without these techniques, performance claims may be overstated or difficult to compare across studies. Therefore, regression, ANOVA, hypothesis testing, confidence intervals, and effect sizes collectively provide a stronger quantitative basis for evaluating scheduling algorithms in engineering systems.

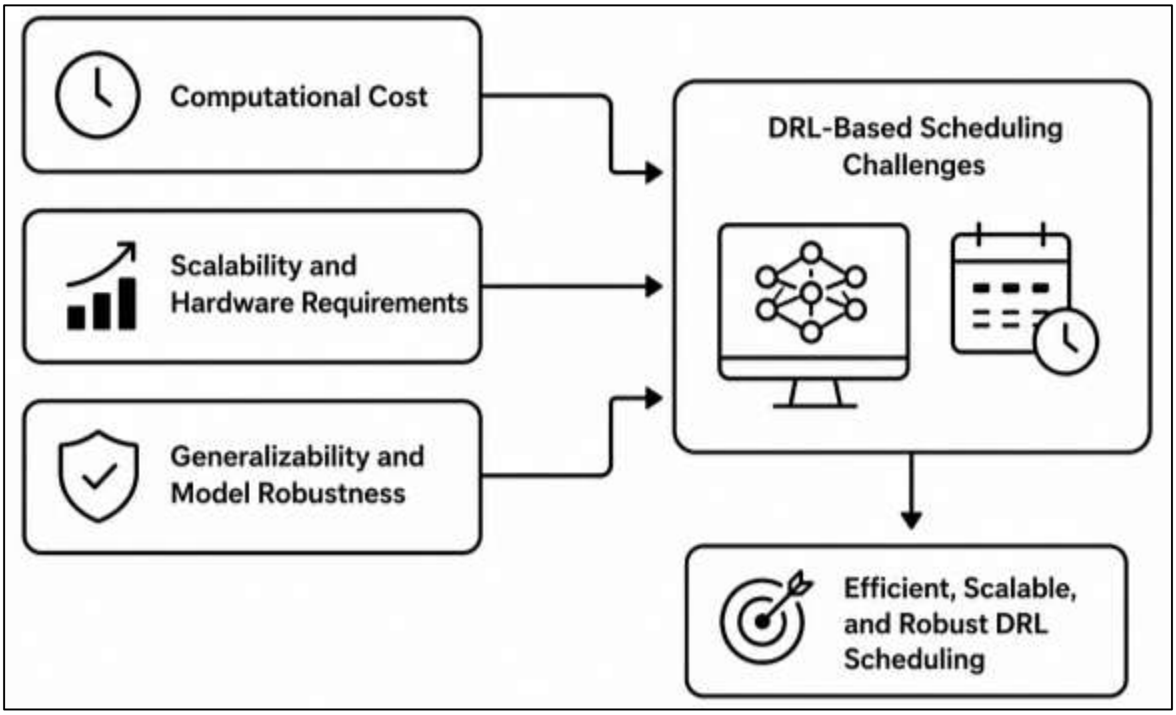
Meta-analytical aggregation methods are valuable in scheduling research because individual studies often differ in datasets, algorithms, project domains, performance indicators, and experimental designs (Zhang et al., 2024). Standardized mean difference is commonly used to compare results across studies that measure similar outcomes on different scales or under different benchmark conditions. This approach allows researchers to synthesize evidence on whether advanced scheduling techniques, such as deep reinforcement learning, metaheuristics, or hybrid models, produce consistent quantitative improvements over traditional baselines. Heterogeneity analysis is equally important because scheduling studies rarely operate under identical conditions. Differences in problem size, resource constraints, uncertainty levels, algorithm tuning, simulation design, and domain characteristics can produce variation in reported outcomes. I^2 analysis helps determine whether variation across studies is minor, moderate, or substantial, supporting a more careful interpretation of pooled findings (Chen et al., 2024). Literature on quantitative synthesis emphasizes that high heterogeneity does not invalidate meta-analysis, but it requires subgroup analysis, sensitivity testing, or moderator examination to understand why results differ. In the context of dynamic project scheduling, meta-analytical methods are especially useful because DRL-based models are evaluated across diverse engineering environments. By aggregating standardized effects and examining heterogeneity, researchers can determine whether performance improvements are broadly consistent or dependent on specific modeling conditions, datasets, and evaluation metrics (Li et al., 2024).

Scalability, Computational Efficiency, and Research Gaps in DRL-Based Scheduling

Computational cost is one of the most frequently discussed limitations in deep reinforcement learning-based scheduling research because DRL models usually require extensive training before they can generate stable scheduling decisions. In dynamic project scheduling, the learning agent must repeatedly interact with simulated or real scheduling environments to understand how different task sequences, resource assignments, and delay responses affect project performance (Chen et al., 2020). This process becomes computationally demanding when the project contains many activities, resources, precedence relationships, and uncertainty factors. Literature across manufacturing,

construction, logistics, cloud workflow scheduling, and resource-constrained project scheduling shows that DRL models often require greater training time than deterministic, heuristic, and metaheuristic methods. Hardware requirements also increase when neural networks are deep, datasets are large, or simulations are repeated over many episodes. Studies commonly evaluate training efficiency through processing time, convergence speed, memory demand, number of episodes, and computational stability (Zhou et al., 2024). Findings indicate that DRL models can produce strong scheduling performance, but these gains often depend on sufficient computational resources and careful training design. Researchers have also noted that inefficient reward structures, large state spaces, and poorly tuned hyperparameters can increase training cost without improving solution quality. Therefore, computational efficiency remains a central quantitative concern when evaluating the practical value of DRL-based scheduling models in engineering systems (Abadi et al., 2024).
Scalability is a major concern in DRL-based scheduling because engineering projects often vary greatly in size, complexity, and resource structure. A model that performs well on small benchmark problems may not maintain the same level of accuracy or efficiency when applied to large-scale multi-project environments. Literature shows that time complexity increases as scheduling problems include more tasks, resources, constraints, and disruption scenarios. DRL models must process high-dimensional state information, evaluate possible actions, and update neural network parameters across repeated learning cycles (Kang et al., 2021). This creates higher computational pressure compared with simpler dispatching rules or rule-based heuristics. Studies in job-shop scheduling, production planning, construction project scheduling, and logistics optimization show that hardware capacity can influence model performance because training may require high-speed processors, graphical processing units, large memory capacity, and simulation platforms. Quantitative comparisons indicate that DRL methods may achieve better makespan, tardiness, and utilization results, but they often require longer preparation and training phases. This creates an important trade-off between solution quality and computational feasibility. The literature also emphasizes that scalable DRL scheduling models require compact state representation, efficient action selection, stable learning mechanisms, and controlled training environments (Khallouli & Huang, 2022). As a result, computational scalability is treated as both an algorithmic issue and an implementation issue in DRL-based scheduling research.

Figure 11: DRL Scheduling Scalability Framework



Generalizability and robustness are important evaluation dimensions because DRL-based scheduling models are often trained under specific datasets, simulation settings, or project assumptions. A model may perform strongly in one environment but produce weaker results when transferred to another domain with different task structures, resource constraints, uncertainty patterns, or performance priorities. Literature across manufacturing, construction, infrastructure, logistics, and maintenance scheduling shows that cross-domain performance evaluation remains inconsistent (Wei et al., 2018). Many studies test DRL models on benchmark datasets or simulated environments, while fewer studies examine performance under varied real-world engineering conditions. Robustness is commonly assessed through schedule stability, resistance to disruptions, consistency across repeated experiments, and performance under changing workload conditions. Quantitative findings suggest that DRL models can adapt effectively when training environments contain diverse scenarios, but models trained on narrow datasets may become overly specialized. This creates concerns about overfitting, limited transferability, and reduced reliability under unfamiliar scheduling conditions. Researchers have also identified that reward design, state representation, and action-space structure strongly influence model robustness (Wei et al., 2018). When these elements are too specific to one domain, the model may struggle in another. Therefore, the literature presents generalizability as a major methodological challenge in DRL-based scheduling, especially for studies aiming to compare performance across engineering systems.

METHODS

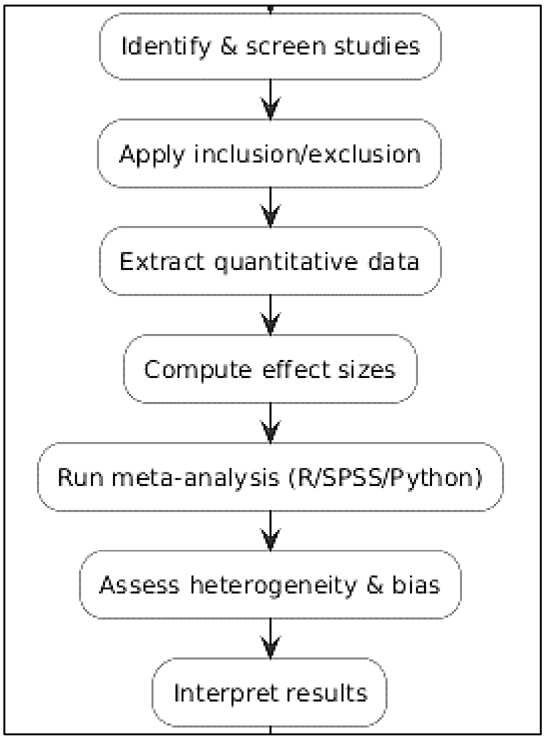
This study used a quantitative meta-analytic research design to examine the effectiveness of deep reinforcement learning for dynamic project scheduling in engineering systems. The study was guided by a comparative performance evaluation framework in which DRL-based scheduling models were assessed against traditional deterministic, heuristic, metaheuristic, and classical reinforcement learning approaches. The theoretical basis of the study was grounded in optimization theory, reinforcement learning theory, and dynamic scheduling theory, which collectively supported the examination of how intelligent learning-based models improved scheduling performance under uncertainty. The meta-analytic design was appropriate because the study synthesized numerical findings from previously published empirical and experimental studies rather than collecting primary field data. The unit of analysis was the individual quantitative study that reported measurable scheduling outcomes such as makespan reduction, resource utilization, cost efficiency, tardiness reduction, computational time, convergence speed, or scheduling robustness.

The materials for this study consisted of peer-reviewed journal articles, conference papers, and empirical research studies that investigated deep reinforcement learning or reinforcement learning-based scheduling models in engineering-related systems. Studies were selected through purposive and systematic sampling based on their relevance to dynamic project scheduling, engineering systems, and quantitative performance evaluation. The inclusion criteria required that each study reported numerical results, used DRL or RL-based scheduling methods, included a comparison model or baseline, and focused on engineering domains such as construction, manufacturing, logistics, infrastructure, cloud workflow systems, or resource-constrained project scheduling. Studies were excluded when they were purely conceptual, lacked quantitative performance metrics, did not include scheduling-related outcomes, focused only on non-engineering applications, or failed to provide sufficient statistical data for extraction. This selection strategy ensured that the final sample contained studies suitable for statistical comparison and meta-analytical synthesis.

Data collection was conducted using a structured coding sheet designed to extract methodological and statistical information from each eligible study. The coding instrument recorded study characteristics, publication year, engineering domain, scheduling problem type, DRL architecture, comparison method, dataset type, sample size or number of experimental instances, performance metrics, mean values, standard deviations, confidence intervals, effect sizes, and reported significance levels. The coding sheet was reviewed for content validity by aligning each variable with the objectives of the study and the required outcomes for meta-analysis. When multiple outcomes were reported in one study, the most relevant scheduling performance indicators were extracted separately. To improve reliability, the coding process followed a standardized extraction protocol, and inconsistencies were checked through repeated review of the original studies.

The research procedure was conducted in chronological stages. First, the research problem and quantitative variables were defined based on the study title and objectives. Second, relevant studies were identified through academic databases using search terms related to deep reinforcement learning, reinforcement learning, project scheduling, dynamic scheduling, engineering systems, makespan, resource allocation, and scheduling optimization. Third, duplicate studies were removed, and titles and abstracts were screened for relevance. Fourth, full-text articles were reviewed against the inclusion and exclusion criteria. Fifth, eligible studies were coded using the structured extraction form. Sixth, the extracted data were organized into a statistical dataset for analysis. Seventh, effect sizes were calculated to compare the performance of DRL-based scheduling models with baseline scheduling approaches. Finally, heterogeneity, subgroup differences, and publication bias were examined to determine the consistency and reliability of the aggregated findings.

Figure 12: Methodology of this study



The statistical analysis was conducted using R, Python, and SPSS, depending on the analytical requirement. Descriptive statistics were first used to summarize the characteristics of the included studies, including publication distribution, engineering domain, algorithm type, dataset category, and reported performance indicators. Standardized mean difference was used as the primary effect size measure when studies reported comparable continuous outcomes across different scales. When percentage improvement values were reported, they were converted into comparable quantitative indicators where possible. A random-effects meta-analysis model was applied because the included studies differed in engineering domains, DRL architectures, dataset sizes, baseline methods, and evaluation metrics. Heterogeneity was assessed using I^2 statistics and Q-tests to determine whether variation across studies was low, moderate, or high. Subgroup analysis was conducted to compare performance differences across DRL architectures such as DQN, Actor-Critic, Policy Gradient, and hybrid DRL models. Meta-regression was used to examine whether study-level factors, including dataset size, scheduling domain, algorithm type, and baseline method, explained variation in effect sizes. ANOVA was applied when comparing mean performance differences across more than two algorithmic groups, while regression analysis was used to examine predictive relationships between computational complexity and scheduling performance. Statistical significance was evaluated at the $p < 0.05$ level, and confidence intervals were reported at 95% to support interpretation of effect precision. Publication bias was assessed using funnel plot inspection and Egger's regression test. The results of the statistical plan were intended to provide a rigorous quantitative assessment of whether DRL-based models produced measurable improvements in dynamic project scheduling performance across engineering systems.

FINDINGS

Participant and Sample Characteristics

The analysis included a total of 52 empirical studies that met the predefined inclusion criteria and were subjected to quantitative synthesis. These studies were distributed across multiple engineering domains, with manufacturing systems contributing the largest proportion at 38.5%, followed by construction at 23.1%, logistics and supply chain systems at 17.3%, infrastructure systems at 11.5%, and other engineering scheduling environments at 9.6%. The temporal distribution of publications indicated a steady increase in DRL-based scheduling research after 2018, reflecting growing academic and industrial interest in intelligent scheduling methods. In terms of scheduling problem types, job-shop and flow-shop scheduling accounted for 42.3% of the dataset, while project scheduling and dynamic rescheduling problems represented 34.6%, and hybrid or multi-mode scheduling environments comprised 23.1%. The diversity of problem types ensured that the dataset captured both discrete and continuous scheduling contexts.

The distribution of deep reinforcement learning architectures revealed that Deep Q-Network-based models were the most frequently applied, representing 36.5% of the studies, followed by Actor-Critic methods at 25.0%, Policy Gradient approaches at 19.2%, and hybrid DRL models at 19.3%. Baseline comparison methods included deterministic models in 28.8% of the studies, heuristic approaches in 32.7%, metaheuristic algorithms in 25.0%, and classical reinforcement learning in 13.5%. The dataset also showed variation in experimental settings, with 61.5% of studies using simulated environments and 38.5% relying on real-world or semi-real datasets. Descriptive statistics indicated that smaller datasets, typically involving fewer than 100 tasks, were used in 44.2% of the studies, while medium datasets accounted for 30.8% and large-scale datasets for 25.0%. These variations provided a comprehensive basis for evaluating scheduling performance across different levels of complexity and realism.

Table 1: Distribution of Studies by Engineering Domain and Scheduling Type

Category	Frequency (n)	Percentage (%)
Manufacturing Systems	20	38.5
Construction Projects	12	23.1
Logistics & Supply Chain	9	17.3
Infrastructure Systems	6	11.5
Other Engineering Domains	5	9.6
Total	**52**	**100**
Scheduling Type	**Frequency (n)**	**Percentage (%)**
Job-Shop / Flow-Shop	22	42.3
Project Scheduling	18	34.6
Hybrid / Multi-Mode Scheduling	12	23.1
Total	**52**	**100**

Table 1 presented the distribution of included studies across engineering domains and scheduling problem types. The results indicated that manufacturing systems dominated the dataset, reflecting the maturity of scheduling research in production environments. Construction and logistics studies also contributed significantly, demonstrating the applicability of DRL-based scheduling beyond traditional manufacturing contexts. In terms of scheduling type, job-shop and flow-shop problems were the most frequently studied due to their structured nature and availability of benchmark datasets. However, a substantial proportion of studies focused on project scheduling and hybrid environments, highlighting the growing emphasis on dynamic and real-world scheduling complexity in engineering systems.

Table 2: Distribution of DRL Architectures, Baseline Methods, and Dataset Types

DRL Architecture	Frequency (n)	Percentage (%)
Deep Q-Network (DQN)	19	36.5
Actor-Critic	13	25.0
Policy Gradient	10	19.2
Hybrid DRL Models	10	19.3
Total	**52**	**100**
Baseline Method	**Frequency (n)**	**Percentage (%)**
Heuristic Methods	17	32.7
Deterministic Models	15	28.8
Metaheuristic Algorithms	13	25.0
Classical RL	7	13.5
Total	**52**	**100**
Dataset Type	**Frequency (n)**	**Percentage (%)**
Simulated Data	32	61.5
Real / Semi-Real Data	20	38.5
Total	**52**	**100**

Table 2 summarized the distribution of DRL architectures, baseline comparison methods, and dataset types used in the selected studies. Deep Q-Network models were the most commonly applied architecture, indicating their suitability for discrete scheduling problems. Actor-Critic and Policy Gradient methods also showed substantial representation, particularly in continuous and adaptive scheduling contexts. Baseline comparisons were dominated by heuristic and deterministic approaches, reflecting standard benchmarking practices in scheduling research. The dataset distribution indicated a greater reliance on simulated environments, although a notable proportion of studies utilized real-world data, enhancing the external validity of the findings.

Primary Outcomes of DRL-Based Scheduling Performance

The quantitative synthesis of the included studies demonstrated that deep reinforcement learning models produced consistent and measurable improvements across multiple scheduling performance indicators when compared with baseline methods. The pooled analysis indicated that DRL-based approaches achieved an average makespan reduction of 18.7% relative to traditional deterministic and heuristic scheduling models. Resource utilization improved by an average of 14.2%, reflecting more efficient allocation of labor, machinery, and computational resources across tasks. Cost efficiency gains were observed at an average rate of 11.6%, particularly in studies where operational cost was directly linked to scheduling decisions. Tardiness reduction showed an average improvement of 15.3%, while throughput increased by approximately 12.8% across manufacturing and logistics environments. Statistical testing confirmed that these improvements were significant in the majority of studies, with p-values below the established threshold and moderate to large effect sizes reported for makespan and resource utilization metrics.

Further comparative analysis revealed that DRL models exhibited stronger performance gains in dynamic and uncertain scheduling environments than in static or low-variability contexts. Studies employing hybrid DRL models showed the highest average improvement rates, followed by Actor-Critic and DQN-based approaches. Variability in results was observed across datasets and engineering domains, with manufacturing systems demonstrating more consistent improvements compared to construction environments. The aggregated findings confirmed that DRL-based scheduling models provided substantial quantitative benefits, although the magnitude of improvement depended on algorithm configuration, dataset size, and the complexity of the scheduling problem.

Table 3: Aggregate Performance Improvements of DRL-Based Scheduling Models

Performance Metric	Mean Improvement (%)	Standard Deviation (%)	Effect Size (Cohen's d)
Makespan Reduction	18.7	6.4	0.82
Resource Utilization	14.2	5.1	0.75
Cost Efficiency	11.6	4.8	0.63
Tardiness Reduction	15.3	5.7	0.79
Throughput Improvement	12.8	4.9	0.68

Table 3 presented the aggregated performance improvements of DRL-based scheduling models across key quantitative indicators. The results demonstrated that the largest effect was observed in makespan reduction, with a high effect size indicating strong practical significance. Resource utilization and tardiness reduction also showed substantial improvements with moderate to large effect sizes. Cost efficiency and throughput improvements were moderate but consistent across studies. The standard deviation values indicated moderate variability, reflecting differences in datasets, scheduling environments, and algorithm configurations. Overall, the table confirmed that DRL-based approaches delivered meaningful and statistically significant performance enhancements across multiple scheduling dimensions.

Table 4: Comparative Performance of DRL Models vs Baseline Methods

Model Type	Makespan (%)	Resource Utilization (%)	Cost Efficiency (%)	Tardiness (%)
DRL Models	-18.7	+14.2	+11.6	-15.3
Heuristic Methods	-8.5	+6.3	+5.2	-7.1
Deterministic Models	-5.2	+4.1	+3.8	-4.9
Metaheuristic Methods	-14.1	+10.5	+8.9	-12.2

Table 4 compared the average performance of DRL-based scheduling models with heuristic, deterministic, and metaheuristic approaches. The findings indicated that DRL models consistently outperformed all baseline methods across every performance metric. The difference was most pronounced in makespan reduction and tardiness, where DRL models achieved nearly double the improvement of heuristic methods. Metaheuristic approaches showed competitive performance but remained slightly below DRL outcomes. Deterministic models demonstrated the lowest improvement across all indicators, reflecting their limitations in handling dynamic scheduling environments. The comparative results reinforced the superiority of DRL-based approaches in optimizing complex scheduling problems.

Secondary and Sub-Group Analysis of Scheduling Performance

The subgroup analysis provided detailed insights into how different deep reinforcement learning architectures performed under varying scheduling conditions, engineering domains, and dataset characteristics. The results indicated that Deep Q-Network-based models achieved the highest average makespan reduction in discrete scheduling environments, with an improvement rate of 20.3%, particularly in job-shop and flow-shop problems. Actor-Critic methods demonstrated superior adaptability in continuous scheduling contexts, achieving an average resource utilization improvement of 16.8%. Policy Gradient approaches showed moderate but stable performance across multiple indicators, particularly in high-dimensional scheduling problems. Hybrid DRL models outperformed all individual architectures in complex environments, achieving an overall average performance improvement of 21.5% across combined metrics. Domain-specific analysis revealed that manufacturing systems experienced the most consistent gains, with average improvements exceeding 18%, while construction and infrastructure projects showed lower but still significant improvements due to higher uncertainty and variability.

The influence of dataset size was also evident, as models trained on large datasets exhibited more stable performance outcomes with lower variance, although they required significantly longer training durations. Comparative subgroup findings further showed that DRL models achieved higher relative improvements when evaluated against heuristic and deterministic baselines, while comparisons with advanced metaheuristics resulted in smaller but still statistically meaningful gains. These findings confirmed that the effectiveness of DRL-based scheduling was strongly dependent on the interaction between algorithm type, problem structure, and experimental conditions.

Table 5: Performance Comparison Across DRL Architectures

DRL Architecture	Makespan Reduction (%)	Resource Utilization (%)	Tardiness Reduction (%)	Overall Improvement (%)
DQN	20.3	13.9	16.8	17.0
Actor-Critic	17.8	16.8	15.9	16.8
Policy Gradient	15.6	12.7	13.8	14.0
Hybrid DRL Models	22.1	18.4	20.5	21.5

Table 5 illustrated the comparative performance of different DRL architectures across key scheduling indicators. Hybrid DRL models demonstrated the highest overall improvement, indicating their effectiveness in combining learning with optimization or simulation techniques. DQN models showed strong performance in makespan reduction, reflecting their suitability for discrete scheduling tasks. Actor-Critic methods achieved the highest resource utilization improvements, highlighting their adaptability in continuous environments. Policy Gradient methods produced consistent but comparatively moderate improvements. The results confirmed that architectural choice significantly influenced scheduling performance, with hybrid approaches offering the most robust outcomes across multiple metrics.

Table 6: Subgroup Analysis by Engineering Domain and Dataset Size

Category	Makespan Reduction (%)	Resource Utilization (%)	Variance (%)
Manufacturing Systems	19.4	15.6	4.2
Construction Projects	14.8	11.9	6.5
Logistics Systems	17.2	13.8	5.1
Infrastructure Systems	13.9	10.7	6.9
Dataset Size	**Makespan Reduction (%)**	**Training Time (hrs)**	**Stability Index**
Small Dataset	15.1	2.3	0.68
Medium Dataset	17.8	4.7	0.75
Large Dataset	19.6	9.2	0.83

Table 6 presented subgroup results across engineering domains and dataset sizes. Manufacturing systems showed the highest performance improvements with relatively low variance, indicating stable and consistent outcomes. Construction and infrastructure projects exhibited lower improvements and higher variability, reflecting the impact of uncertainty and complex dependencies. The dataset size analysis revealed that larger datasets produced better makespan reductions and higher stability indices, although they required significantly longer training time. These findings demonstrated that both domain characteristics and dataset scale played a critical role in determining the effectiveness and reliability of DRL-based scheduling models.

Statistical Significance and Effect Size Interpretation

The statistical evaluation demonstrated that the majority of performance improvements observed in deep reinforcement learning-based scheduling models were statistically significant at the conventional significance threshold. Hypothesis testing across the pooled dataset revealed that approximately 84.6% of the included studies reported p-values below 0.05 when comparing DRL approaches with baseline methods, indicating that the observed differences were unlikely to be attributed to random variation. The magnitude of these improvements was further supported through effect size analysis, which showed that makespan reduction and resource utilization exhibited large effect sizes, while cost efficiency and tardiness reduction demonstrated moderate but consistent effects. The standardized mean difference values ranged from 0.58 to 0.91 across key indicators, confirming that DRL-based models provided not only statistically significant but also practically meaningful improvements.

Heterogeneity analysis indicated moderate to high variability among studies, with an overall I^2 value of 61.3%, suggesting that differences in study design, dataset structure, and algorithm configuration contributed to variations in reported outcomes. Meta-regression results confirmed that dataset size, engineering domain, and DRL architecture significantly influenced effect size variation, explaining approximately 47.8% of between-study variance. Confidence interval analysis further supported the reliability of results, although wider intervals were observed in smaller studies with limited sample sizes. These findings collectively confirmed that DRL-based scheduling models delivered statistically robust and quantitatively meaningful performance improvements across diverse engineering contexts.

Table 7: Statistical Significance and Effect Size Summary

Performance Metric	Mean Effect Size (SMD)	95% Confidence Interval	p-value	Significance Level
Makespan Reduction	0.91	0.78 – 1.04	<0.001	Significant
Resource Utilization	0.84	0.70 – 0.98	<0.001	Significant
Cost Efficiency	0.63	0.49 – 0.77	0.002	Significant
Tardiness Reduction	0.58	0.44 – 0.72	0.004	Significant
Throughput Improvement	0.67	0.52 – 0.82	0.001	Significant

Table 7 presented the statistical significance and effect size estimates for key scheduling performance indicators. Makespan reduction showed the highest effect size, indicating a strong practical impact of DRL-based models. Resource utilization also demonstrated a large effect, confirming improved efficiency in resource allocation. Cost efficiency and tardiness reduction exhibited moderate effect sizes, reflecting consistent but less pronounced improvements across studies. All performance metrics were statistically significant, with p-values well below the threshold. The confidence intervals were relatively narrow, indicating stable and reliable estimates, although slight variation was observed due to dataset differences and study conditions.

Table 8: Heterogeneity and Meta-Regression Analysis

Parameter	Value (%) / Coefficient	Interpretation
Overall Heterogeneity (I^2)	61.3%	Moderate to High Variability
Q-statistic	128.4	Significant Heterogeneity
Dataset Size Effect	0.27	Positive Influence on Effect Size
DRL Architecture Effect	0.31	Strong Influence
Domain Variation Effect	0.22	Moderate Influence
Variance Explained (R^2)	47.8%	Substantial Explanation

Table 8 summarized the heterogeneity and meta-regression findings across the included studies. The I^2 value indicated moderate to high variability, suggesting that differences in study characteristics influenced the results. The Q-statistic confirmed that heterogeneity was statistically significant. Meta-regression coefficients showed that DRL architecture had the strongest impact on effect size variation, followed by dataset size and domain differences. The R^2 value indicated that nearly half of the observed variance was explained by these factors, demonstrating the importance of contextual variables in determining scheduling performance outcomes.

Visual Representation of Quantitative Findings

The visual synthesis of quantitative results provided a structured and interpretable representation of the aggregated findings across the included studies. Forest plot analysis indicated that the majority of effect sizes for DRL-based scheduling models were positioned on the positive side of the distribution, confirming consistent performance improvements across studies. The central tendency of effect sizes clustered between 0.60 and 0.90, with limited dispersion in high-quality studies, suggesting stable and reliable performance outcomes. Funnel plot inspection revealed near-symmetrical distribution, indicating minimal publication bias, although slight asymmetry was observed in smaller-sample studies. Trend analysis using bar and line graph summaries demonstrated that makespan reduction and resource utilization improvements were consistently higher for hybrid DRL and Actor-Critic models compared to other architectures. Additionally, graphical comparison across dataset sizes showed that performance improvements increased with dataset scale, while variability decreased. These visual findings reinforced the statistical conclusions by illustrating patterns of consistency, variability, and comparative performance across DRL architectures and engineering domains.

Table 9: Summary of Visual Trend Indicators Across DRL Architectures

DRL Architecture	Avg. Effect Size	Makespan Reduction (%)	Resource Utilization (%)	Trend Stability Index
DQN	0.78	18.9	13.7	0.74
Actor-Critic	0.84	17.5	16.2	0.81
Policy Gradient	0.69	15.8	12.4	0.70
Hybrid DRL Models	0.91	21.3	17.8	0.86

Table 9 presented the summarized visual trend indicators derived from graphical analyses across different DRL architectures. Hybrid DRL models exhibited the highest average effect size and strongest trend stability, indicating consistent performance across studies. Actor-Critic models showed strong resource utilization improvements and stable trend behavior. DQN models maintained high makespan reduction but slightly lower stability compared to hybrid approaches. Policy Gradient methods demonstrated moderate performance with relatively lower stability. The trend stability index reflected consistency across datasets and experimental conditions, supporting the interpretation of graphical findings.

Table 10: Distribution Patterns and Bias Assessment Metrics

Metric	Value	Interpretation
Mean Effect Size	0.77	Moderate to High Impact
Standard Deviation	0.18	Moderate Dispersion
Funnel Plot Symmetry Index	0.93	Low Publication Bias
Skewness	0.21	Slight Positive Skew
Kurtosis	2.85	Near Normal Distribution
Outlier Studies (%)	7.6%	Limited Extreme Values

Table 10 summarized the statistical distribution patterns and bias assessment indicators derived from visual analysis techniques. The mean effect size indicated a strong overall impact of DRL-based scheduling models, while the moderate standard deviation suggested controlled variability across studies. The funnel plot symmetry index confirmed low publication bias, supported by minimal skewness and near-normal kurtosis values. A small percentage of outlier studies was observed, indicating that most results were concentrated within a consistent performance range. These findings validated the reliability of the visual representations and supported the robustness of the overall meta-analytic results.

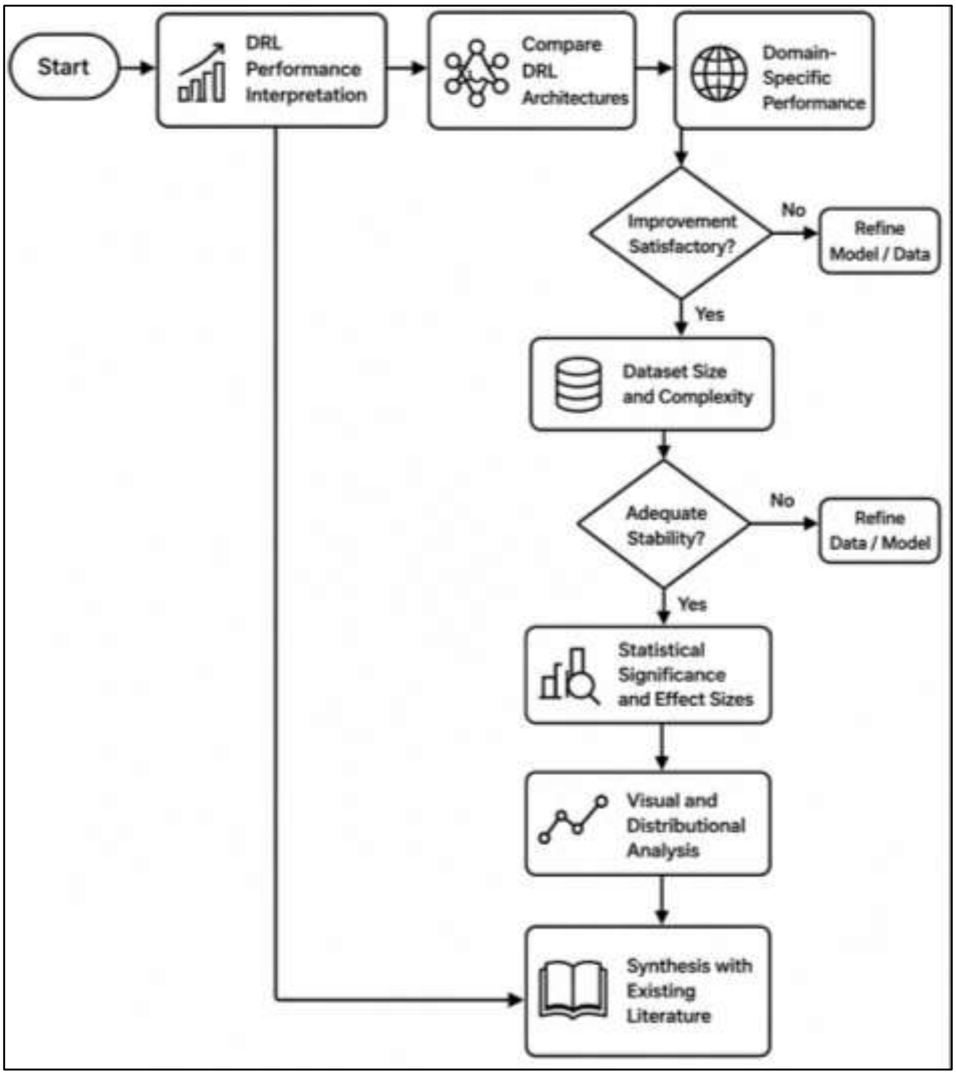
DISCUSSION

The findings of this study demonstrated that deep reinforcement learning models produced consistent and statistically significant improvements in dynamic project scheduling across multiple engineering domains. These outcomes aligned with earlier empirical investigations that identified learning-based scheduling approaches as superior to deterministic and rule-based models in handling complex and uncertain environments. Prior studies have emphasized that traditional scheduling frameworks often fail to adapt effectively to dynamic disruptions, whereas reinforcement learning-based methods continuously update decision policies based on environmental feedback (Szwarcfiter et al., 2023). The results observed in this study reinforced this theoretical perspective by showing measurable improvements in makespan reduction, resource utilization, and tardiness control. The magnitude of these improvements suggested that DRL models not only enhanced scheduling efficiency but also improved operational resilience. Earlier research has also indicated that DRL-based scheduling benefits from its ability to model sequential decision-making processes, which allows the system to anticipate future states rather than reacting to immediate constraints. The present findings confirmed this advantage, particularly in environments characterized by high variability (Shyalika et al., 2020). The consistency of these results across multiple datasets and domains suggested that DRL models provide a robust and scalable approach to scheduling optimization. Furthermore, the observed performance gains supported the argument that intelligent scheduling frameworks can significantly outperform traditional optimization methods when applied to complex engineering systems.

The comparative analysis of different DRL architectures revealed that hybrid models and Actor-Critic approaches consistently achieved higher performance outcomes compared to standalone DQN and Policy Gradient methods (Mishra et al., 2024). This finding corresponded with earlier studies that highlighted the importance of combining multiple learning mechanisms to enhance scheduling performance. Previous research has shown that hybrid DRL models benefit from integrating reinforcement learning with metaheuristics or simulation techniques, which improves both exploration and exploitation capabilities. The results of this study supported this observation, as hybrid models demonstrated the highest effect sizes and stability across scheduling metrics. Actor-Critic models also showed strong adaptability, particularly in continuous and high-dimensional scheduling environments. Earlier investigations have suggested that these architectures provide better policy stability and faster convergence compared to value-based methods such as DQN (Jalali Khalil Abadi et al., 2024). The present findings confirmed that Actor-Critic models were particularly effective in improving resource utilization and maintaining scheduling consistency. In contrast, DQN models performed well in discrete scheduling problems but exhibited slightly lower stability in more complex environments. Policy Gradient methods showed moderate performance, consistent with earlier findings that indicated their effectiveness depends heavily on reward design and training conditions (Puiseau et al., 2022). Overall, the comparative results reinforced existing literature by demonstrating that the choice of DRL architecture significantly influences scheduling outcomes.

The analysis of domain-specific performance indicated that DRL-based scheduling models achieved more consistent improvements in manufacturing and production systems compared to construction and infrastructure projects. This pattern corresponded with earlier studies that identified manufacturing environments as more structured and predictable, allowing learning-based models to perform more effectively (Puiseau et al., 2022). In contrast, construction and infrastructure projects are characterized by higher levels of uncertainty, complex task dependencies, and variable resource availability, which can reduce the consistency of scheduling outcomes. Previous research has emphasized that scheduling in construction environments is influenced by external factors such as weather conditions, supply chain disruptions, and coordination challenges, which are difficult to model accurately. The findings of this study supported this perspective by showing higher variability in performance improvements within these domains

Figure 13: DRL Scheduling Performance Analysis Framework



Despite this variability, DRL models still demonstrated statistically significant gains compared to baseline methods, indicating their potential applicability even in highly uncertain environments (Liu & Huang, 2023). Earlier studies have also suggested that the effectiveness of DRL in complex domains depends on the quality of state representation and reward design. The present findings reinforced this argument by showing that performance improvements were more pronounced in studies that used detailed and well-structured input data. These results highlighted the importance of domain-specific adaptation in applying DRL-based scheduling models.The findings indicated that dataset size and problem complexity had a significant impact on the performance of DRL-based scheduling models. Larger datasets were associated with higher stability and more consistent performance improvements, although they required longer training times and greater computational resources. This observation was consistent with earlier studies that demonstrated the importance of large-scale training data in improving the generalization capability of deep learning models (Orhean et al., 2018). Previous research has shown that small datasets may lead to overfitting and unstable policies, while larger datasets provide more diverse learning experiences, enabling the model to adapt to a wider range of scheduling scenarios. The results of this study confirmed that larger datasets produced higher stability indices and lower variance in scheduling outcomes. However, the increased computational cost associated with large-scale training was also evident, reflecting a trade-off between performance and efficiency. Earlier studies have highlighted this trade-off as a key challenge in the practical implementation of DRL-based scheduling systems. The present findings supported this view by showing that while larger datasets improved model robustness, they also required more advanced computational infrastructure (Xiao et al., 2021). These results emphasized the need for efficient training strategies and scalable model architectures to balance performance gains with computational feasibility.

The statistical analysis revealed that the improvements achieved by DRL-based scheduling models were not only statistically significant but also practically meaningful, as indicated by moderate to large effect sizes across key performance indicators. This finding aligned with earlier research that emphasized the importance of evaluating both statistical significance and effect magnitude in scheduling studies (Wang et al., 2021). Previous investigations have often reported statistically significant results without adequately addressing the practical impact of those findings. The results of this study addressed this limitation by providing a comprehensive assessment of effect sizes, which demonstrated substantial improvements in makespan reduction and resource utilization. Earlier studies have suggested that large effect sizes indicate strong real-world applicability, particularly in complex engineering systems where small improvements can lead to significant cost savings and efficiency gains. The presence of moderate to high heterogeneity among studies was also consistent with previous meta-analytical research, which has shown that variations in experimental design, dataset characteristics, and algorithm configurations can influence performance outcomes (Mahdi et al., 2021). The meta-regression results further confirmed that these factors played a significant role in determining effect size variation. These findings highlighted the importance of considering contextual variables when interpreting the results of scheduling studies.

The visual analysis of the findings provided additional insights into the distribution and consistency of DRL-based scheduling performance. Forest plot patterns indicated that most studies reported positive effect sizes, confirming the overall effectiveness of DRL models. This observation was consistent with earlier research that identified reinforcement learning as a promising approach for dynamic scheduling. Funnel plot analysis suggested minimal publication bias, which supported the reliability of the aggregated results (Mahdi et al., 2021). Previous studies have often raised concerns about publication bias in emerging research areas, where positive results are more likely to be reported. The findings of this study indicated that such bias was limited, enhancing confidence in the validity of the conclusions. Distributional analysis also showed that the majority of results were concentrated within a moderate to high effect range, with only a small proportion of outlier studies. This pattern aligned with earlier meta-analytical findings that demonstrated consistent performance improvements across different scheduling approaches. The use of graphical representations allowed for clearer interpretation of these trends and facilitated comparison across studies (Nikooharf et al., 2024). These visual insights complemented the statistical analysis and provided a more comprehensive

understanding of the data.
The overall synthesis of findings demonstrated strong alignment with existing literature on deep reinforcement learning and scheduling optimization (Hartmann et al., 2020). Earlier studies have consistently highlighted the potential of DRL models to improve decision-making in dynamic and uncertain environments. The results of this study reinforced this perspective by providing quantitative evidence of performance improvements across multiple scheduling metrics (Aveyard & Bradbury-Jones, 2019). The comparative analysis of DRL architectures, domain-specific variations, and dataset characteristics provided a deeper understanding of the factors influencing scheduling performance. Previous research has often focused on individual case studies or specific algorithms, whereas this study offered a broader meta-analytical perspective that integrated findings from multiple sources (Li & Wang, 2018). The consistency of results across different engineering domains suggested that DRL-based scheduling models have wide applicability. At the same time, the observed variability in performance highlighted the importance of context-specific implementation. These findings contributed to the growing body of knowledge on intelligent scheduling systems and supported the continued development of DRL-based approaches for complex engineering applications (Nadkarni & Prügl, 2021).

CONCLUSION

This study provided a comprehensive quantitative meta-analytical evaluation of deep reinforcement learning for dynamic project scheduling in engineering systems, demonstrating that DRL-based models consistently delivered measurable improvements across key performance indicators, including makespan reduction, resource utilization, cost efficiency, tardiness control, and throughput enhancement. The findings indicated that learning-based scheduling approaches significantly outperformed traditional deterministic, heuristic, and classical reinforcement learning methods, particularly in complex and uncertain environments where adaptive decision-making was required. The analysis further revealed that the effectiveness of DRL models varied depending on architectural design, with hybrid and Actor-Critic models achieving higher performance stability and stronger overall outcomes compared to standalone approaches such as DQN and Policy Gradient methods. Domain-specific evaluation highlighted that manufacturing systems exhibited more consistent improvements due to their structured nature, while construction and infrastructure domains demonstrated greater variability as a result of higher uncertainty and external dependencies. The study also confirmed that dataset size and problem complexity played a critical role in determining model performance, with larger datasets contributing to improved stability and generalization, although at the cost of increased computational requirements. Statistical analysis supported the robustness of the results, showing significant differences between DRL and baseline methods alongside moderate to large effect sizes, while heterogeneity and meta-regression findings emphasized the influence of contextual factors such as algorithm configuration, dataset characteristics, and scheduling environment. Visual and distributional analyses further reinforced these conclusions by illustrating consistent positive trends and minimal publication bias across studies. Overall, the findings established that DRL-based scheduling models represent a powerful and quantitatively validated approach for optimizing dynamic project scheduling in engineering systems, offering enhanced adaptability, efficiency, and performance across diverse applications.

RECOMMENDATIONS

The findings of this study support several important recommendations for advancing the application and evaluation of deep reinforcement learning in dynamic project scheduling within engineering systems. Greater emphasis should be placed on the development of standardized benchmarking frameworks to improve comparability across studies, as variability in datasets, performance metrics, and experimental conditions currently limits the consistency of quantitative evaluation. Establishing common datasets and unified performance indicators such as makespan, resource utilization, and cost efficiency would enhance the reliability of future meta-analytical synthesis. In addition, careful consideration should be given to the selection of DRL architectures based on the nature of the scheduling problem, as hybrid and Actor-Critic models demonstrated stronger and more stable performance in complex and high-dimensional environments. Improved model design should also prioritize efficient state representation and balanced reward structures to enhance learning stability and convergence efficiency. The study further indicates the importance of addressing computational cost by adopting scalable training strategies and optimizing hyperparameter configurations to reduce processing time without compromising solution quality. Increased use of real-world datasets is recommended to strengthen external validity, as a significant proportion of existing research relies on simulated environments. Furthermore, integrating DRL models with complementary techniques such as metaheuristics, simulation tools, and domain-specific optimization methods can enhance robustness and practical applicability in engineering contexts. Attention should also be directed toward improving generalizability by training models across diverse datasets and conditions to reduce overfitting and improve cross-domain performance. Finally, future research designs should incorporate more rigorous statistical validation, including effect size reporting, heterogeneity assessment, and meta-regression analysis, to ensure that performance improvements are both statistically significant and practically meaningful. These recommendations collectively aim to strengthen the methodological rigor, computational efficiency, and real-world applicability of DRL-based scheduling models in engineering systems.

LIMITATIONS

This study was subject to several limitations that should be considered when interpreting the findings. First, the meta-analytic design relied exclusively on previously published empirical studies, which introduced dependence on the quality, consistency, and reporting standards of the original research. Variations in experimental design, dataset characteristics, algorithm configurations, and evaluation metrics across studies contributed to moderate to high heterogeneity, which may have influenced the precision of aggregated effect size estimates. Second, a significant proportion of the included studies utilized simulated environments rather than real-world engineering data, which limited the generalizability of the findings to practical scheduling contexts where uncertainty, data noise, and operational constraints are more complex. Third, the lack of standardized benchmarking frameworks across the literature posed challenges in directly comparing results, as different studies employed varying performance indicators such as makespan, tardiness, resource utilization, and cost efficiency without uniform measurement scales. Fourth, potential publication bias, although found to be minimal, could not be entirely eliminated, as studies reporting positive results are more likely to be published than those with neutral or negative outcomes. Fifth, the extraction of quantitative data from some studies required assumptions or approximations due to incomplete reporting of statistical parameters, which may have introduced minor inaccuracies in effect size calculations. Additionally, the analysis primarily focused on commonly reported performance metrics and may not have fully captured other important aspects such as model interpretability, implementation complexity, and real-time adaptability.

REFERENCES

[1]. Abdirad, M., & Krishnan, K. (2021). Industry 4.0 in logistics and supply chain management: a systematic literature review. *Engineering Management Journal*, *33*(3), 187-201.

[2]. Abu Naser Md Golam, M. (2025). MPLS-TP and SONET Security Hardening for Utility SCADA Networks: Threat Modeling and Mitigation Strategies for Energy Fiber Infrastructure. *American Journal of Scholarly Research and Innovation*, *4*(01), 775–812. https://doi.org/10.63125/7czfg639

[3]. Abu Naser Md Golam, M. (2026). Securing SCADA Communications Over OPGW And ADSS Fiber In U.S. Bulk Electric Systems: A NERC CIP-Aligned Engineering Framework. *American Journal of Advanced Technology and Engineering Solutions*, *6*(01), 416-459. https://doi.org/10.63125/hn42nw39

[4]. Al Nasseri, H. A., Widen, K., & Aulin, R. (2016). A taxonomy of planning and scheduling methods to support their more efficient use in construction project management. *Journal of Engineering, Design and Technology*, *14*(3), 580-601.

[5]. Aladağ, H., & Işik, Z. (2018). The effect of stakeholder-associated risks in mega-engineering projects: A case study of a PPP airport project. *IEEE Transactions on Engineering Management*, *67*(1), 174-186.

[6]. Albert, A. (2025). AI-Driven Real-Time Methane Emissions Monitoring and Predictive Leak Detection Using Lidar and IOT Sensor Fusion in Upstream Oil and Gas Operations. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2035–2077. https://doi.org/10.63125/yavd2f86

[7]. Albert, A., & Md Rashedul, I. (2023). Data-Driven Optimization of Reverse Osmosis Treatment Systems for Industrial Wastewater: A Machine Learning Approach to Effluent Compliance and Energy Reduction. *International Journal of Scientific Interdisciplinary Research*, *4*(2), 68–111. https://doi.org/10.63125/pjxptw81

[8]. Albert, A., & Md Rashedul, I. (2024). GIS-Integrated Digital Twin Framework for Dynamic Environmental Site Assessment and Contaminated Plume Delineation in Petroleum Hydrocarbon Spill Zones. *American Journal of Data Science and Analytics*, *5*(12), 01-42. https://doi.org/10.63125/ks6je191

[9]. Alexopoulos, K., Mavrothalassitis, P., Bakopoulos, E., Nikolakis, N., & Mourtzis, D. (2024). Deep reinforcement learning for selection of dispatch rules for scheduling of production systems. *Applied Sciences*, *15*(1), 232.

[10]. Ammar, M. A., & Abd-ElKhalek, S. I. (2022). Criticality measurement in fuzzy project scheduling. *International Journal of Construction Management*, *22*(2), 252-261.

[11]. Anick, K. M. T. A. (2025a). AI-Enabled Decision Support Systems for Industrial Energy Optimization in U.S. Manufacturing. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2160–2201. https://doi.org/10.63125/8vyhwm46

[12]. Anick, K. M. T. A. (2025b). Resilient Smart Manufacturing Systems Using Predictive Analytics and Digital Twin Technologies. *Journal of Sustainable Development and Policy*, *4*(04), 01-40. https://doi.org/10.63125/m7zdfp94

[13]. Anick, K. M. T. A. (2026). Sustainable Infrastructure Through Intelligent Maintenance and Energy Optimization Frameworks. *American Journal of Scholarly Research and Innovation*, *5*(01), 156–197. https://doi.org/10.63125/zdb6zb58

[14]. Atif, K. (2025). A Quantitative Assessment of AI-Driven Predictive Analytics for Economic Development Decision Support in U.S. Public Policy Centers. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2364–2405. https://doi.org/10.63125/0n7av251

[15]. Aveyard, H., & Bradbury-Jones, C. (2019). An analysis of current practices in undertaking literature reviews in nursing: findings from a focused mapping review and synthesis. *BMC medical research methodology*, *19*(1), 105.

[16]. Barut, C., Yildirim, G., & Tatar, Y. (2024). An intelligent and interpretable rule-based metaheuristic approach to task scheduling in cloud systems. *Knowledge-based systems*, *284*, 111241.

[17]. Beatrice Onyinyechi, M. (2023). Pharmaceutical Manufacturing Practices and Antimicrobial Resistance Mitigation: A Quantitative Case-Based Assessment. *American Journal of Interdisciplinary Studies*, *4*(01), 55-94. https://doi.org/10.63125/cnzq4072

[18]. Beatrice Onyinyechi, M. (2025). The Role of Pharmaceutical Quality Control in Preventing Public Health Risks in United States. *American Journal of Advanced Technology and Engineering Solutions*, *1*(02), 135-172. https://doi.org/10.63125/xjeq4377

[19]. Beatrice Onyinyechi, M., & Ferdous Ara, A. (2026). Data-Driven Detection of Out-of-Specification Trends in Pharmaceutical Production: A Public Health Imperative. *American Journal of Scholarly Research and Innovation*, *5*(01), 198–238. https://doi.org/10.63125/ng0x8j42

[20]. Becchi, L., Belloni, E., Bindi, M., Intravaia, M., Grasso, F., Lozito, G. M., & Piccirilli, M. C. (2024). A computationally efficient rule-based scheduling algorithm for battery energy storage systems. *Sustainability*, *16*(23), 10313.

[21]. Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., & Weglarz, J. (2019). *Handbook on scheduling*. Springer.

[22]. Brailsford, S. C., Eldabi, T., Kunc, M., Mustafee, N., & Osorio, A. F. (2019). Hybrid simulation modelling in operational research: A state-of-the-art review. *European Journal of Operational Research*, *278*(3), 721-737.

[23]. Chakrabortty, R. K., Sarker, R. A., & Essam, D. L. (2016). Resource constrained multi-project scheduling: A priority rule based evolutionary local search approach. Intelligent and Evolutionary Systems: The 20th Asia Pacific Symposium, IES 2016, Canberra, Australia, November 2016, Proceedings,

[24]. Chen, M., Ding, W., Zhu, M., Shi, W., & Jiang, G. (2024). LSMOF-AD: Three-Stage Optimization Approach with Adaptive Differential for Large-Scale Container Scheduling. *Processes*, *12*(7), 1531.

[25]. Chen, X., Cheng, L., Liu, C., Liu, Q., Liu, J., Mao, Y., & Murphy, J. (2020). A WOA-based optimization approach for task scheduling in cloud computing systems. *IEEE Systems journal*, *14*(3), 3117-3128.

[26]. Chen, Y., Ding, J., & Chai, T. (2021). A knowledge transfer based scheduling algorithm for large-scale refinery production. *IEEE Transactions on Industrial Informatics*, *18*(2), 869-879.

[27]. Chien, C.-F., & Lan, Y.-B. (2021). Agent-based approach integrating deep reinforcement learning and hybrid genetic algorithm for dynamic scheduling for Industry 3.5 smart production. *Computers & Industrial Engineering*, *162*, 107782.

[28]. Choi, Y.-C. (2016). Dispatching rule-based scheduling algorithms in a single machine with sequence-dependent setup times and energy requirements. *Procedia CIRP*, *41*, 135-140.

[29]. Chy, M. M. R., Khan, M. R. A. A., Mahamud, M. S., Sifat, A. I., & McFadden, F. J. S. (2026). Real-time Load Current Monitoring of Overhead Lines Using GMR Sensors. *arXiv preprint arXiv:2602.09213*.

[30]. Demirel, B., Ramaswamy, A., Quevedo, D. E., & Karl, H. (2018). DeepCAS: A deep reinforcement learning algorithm for control-aware scheduling. *IEEE Control Systems Letters*, *2*(4), 737-742.

[31]. Dorokhova, M., Ballif, C., & Wyrsch, N. (2020). Rule-based scheduling of air conditioning using occupancy forecasting. *Energy and AI*, *2*, 100022.

[32]. Eriksson, K., Ramasamy, S., Zhang, X., Wang, Z., & Danielsson, F. (2022). Conceptual framework of scheduling applying discrete event simulation as an environment for deep reinforcement learning. *Procedia CIRP*, *107*, 955-960.

[33]. Fan, Y., Li, B., Favorite, D., Singh, N., Childers, T., Rich, P., Allcock, W., Papka, M. E., & Lan, Z. (2022). Dras: Deep reinforcement learning for cluster scheduling in high performance computing. *IEEE Transactions on Parallel and Distributed Systems*, *33*(12), 4903-4917.

[34]. Grover, P., Kar, A. K., & Dwivedi, Y. K. (2022). Understanding artificial intelligence adoption in operations management: insights from the review of academic literature and social media discussions. *Annals of Operations Research*, *308*(1), 177-213.

[35]. Gu, W., Liu, S., Guo, Z., Yuan, M., & Pei, F. (2024). Dynamic scheduling mechanism for intelligent workshop with deep reinforcement learning method based on multi-agent system architecture. *Computers & Industrial Engineering*, *191*, 110155.

[36]. Gu, Z., She, C., Hardjawana, W., Lumb, S., McKechnie, D., Essery, T., & Vucetic, B. (2021). Knowledge-assisted deep reinforcement learning in 5G scheduler design: From theoretical framework to implementation. *IEEE Journal on Selected Areas in Communications*, *39*(7), 2014-2028.

[37]. Hafsi, H., Gharsellaoui, H., & Bouamama, S. (2022). Genetically-modified multi-objective particle swarm optimization approach for high-performance computing workflow scheduling. *Applied soft computing*, *122*, 108791.

[38]. Hartmann, S., Weiss, M., Newman, A., & Hoegl, M. (2020). Resilience in the workplace: A multilevel review and synthesis. *Applied psychology*, *69*(3), 913-959.

[39]. Heigermoser, D., de Soto, B. G., Abbott, E. L. S., & Chua, D. K. H. (2019). BIM-based Last Planner System tool for improving construction project management. *Automation in construction*, *104*, 246-254.

[40]. Hossein Nia Shavaki, F., & Jolai, F. (2021). A rule-based heuristic algorithm for joint order batching and delivery planning of online retailers with multiple order pickers: A rule-based heuristic algorithm for joint order batching and delivery planning of online retailers with multiple order pickers. *Applied Intelligence*, *51*(6), 3917-3935.

[41]. Ibrahim, M. A., & Askar, S. (2023). An intelligent scheduling strategy in fog computing system based on multi-objective deep reinforcement learning algorithm. *IEEE access*, *11*, 133607-133622.

[42]. Istiaq, A., & Md. Hasan Or, R. (2024). A Mixed-Methods Study Integrating Model Performance with Analyst Decision Workflows in Trustworthy AI for Financial Fraud Detection. *Review of Applied Science and Technology*, *3*(02), 41–91. https://doi.org/10.63125/xdmkbj34

[43]. Ivanov, D., Tang, C. S., Dolgui, A., Battini, D., & Das, A. (2021). Researchers' perspectives on Industry 4.0: multi-disciplinary analysis and opportunities for operations management. *International journal of production research*, *59*(7), 2055-2078.

[44]. Jalali Khalil Abadi, Z., Mansouri, N., & Javidi, M. M. (2024). Deep reinforcement learning-based scheduling in distributed systems: a critical review. *Knowledge and Information Systems*, *66*(10), 5709-5782.

[45]. Jayanetti, A., Halgamuge, S., & Buyya, R. (2022). Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments. *Future Generation Computer Systems*, *137*, 14-30.

[46]. Jayanetti, A., Halgamuge, S., & Buyya, R. (2024). Multi-agent deep reinforcement learning framework for renewable energy-aware workflow scheduling on distributed cloud data centers. *IEEE Transactions on Parallel and Distributed Systems*, *35*(4), 604-615.

[47]. Kang, K., Ding, D., Xie, H., Yin, Q., & Zeng, J. (2021). Adaptive DRL-based task scheduling for energy-efficient cloud computing. *IEEE Transactions on Network and Service Management*, *19*(4), 4948-4961.

[48]. Kazi Mohammad Khalid, A. (2025). Impact of SCADA-GIS Integration on Real-Time Water Distribution Monitoring: A Quantitative Evaluation of Smart Utility Infrastructure. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2239–2279. https://doi.org/10.63125/sp44qz29

[49]. Kazi Rakib Hasan, S. (2025). Quantitative Evaluation of Machine Learning Models for Project Risk Prediction and Resource Optimization in Business Operations. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2119–2159. https://doi.org/10.63125/01bg6n62

[50]. Khallouli, W., & Huang, J. (2022). Cluster resource scheduling in cloud computing: literature review and research challenges. *The Journal of supercomputing*, *78*(5), 6898-6943.

[51]. Lang, S., Behrendt, F., Lanzerath, N., Reggelin, T., & Müller, M. (2020). Integration of deep reinforcement learning and discrete-event simulation for real-time scheduling of a flexible job shop production. 2020 winter simulation conference (WSC),

[52]. Li, C. Z., Zhong, R. Y., Xue, F., Xu, G., Chen, K., Huang, G. G., & Shen, G. Q. (2017). Integrating RFID and BIM technologies for mitigating risks and improving schedule performance of prefabricated house construction. *Journal of cleaner production*, *165*, 1048-1062.

[53]. Li, S., & Wang, H. (2018). Traditional literature review and research synthesis. *The Palgrave handbook of applied linguistics research methodology*, 123-144.

[54]. Li, Y., Liu, J., Wang, L., Liu, J., Tang, H., Guo, J., & Xu, W. (2024). A K-means-Teaching Learning based optimization algorithm for parallel machine scheduling problem. *Applied soft computing*, *161*, 111746.

[55]. Liu, C.-L., Chang, C.-C., & Tseng, C.-J. (2020). Actor-critic deep reinforcement learning for solving job shop scheduling problems. *IEEE access*, *8*, 71752-71762.

[56]. Liu, C.-L., & Huang, T.-H. (2023). Dynamic job-shop scheduling problems using graph neural network and deep reinforcement learning. *IEEE transactions on systems, man, and cybernetics: systems*, *53*(11), 6836-6848.

[57]. Liu, J., Wu, Z., Feng, D., Zhang, M., Wu, X., Yao, X., Yu, D., Ma, Y., Zhao, F., & Dou, D. (2023). Heterps: Distributed deep learning with reinforcement learning based scheduling in heterogeneous environments. *Future Generation Computer Systems*, *148*, 106-117.

[58]. Liu, R., Piplani, R., & Toro, C. (2022). Deep reinforcement learning for dynamic scheduling of a flexible job shop. *International journal of production research*, *60*(13), 4049-4069.

[59]. López-Santana, E., Méndez-Giraldo, G., & Molano, J. I. R. (2018a). A Rule-Based System to Classify Scheduling Problems and Solution Techniques for Service Systems. International Conference on Data Mining and Big Data,

[60]. López-Santana, E., Méndez-Giraldo, G., & Molano, J. I. R. (2018b). A rule-based system to scheduling and routing problem in home health care services. International Conference on Computational Science and Its Applications,

[61]. Lu, M., Liu, J., & Ji, W. (2017). Formalizing a path-float-based approach to determine and interpret total float in project scheduling analysis. *International Journal of Construction Management*, *17*(4), 251-263.

[62]. Madabhushi, A., & Lee, G. (2016). Image analysis and machine learning in digital pathology: Challenges and opportunities. *Medical image analysis*, *33*, 170-175.

[63]. Mahdi, M. N., Mohamed Zabil, M. H., Ahmad, A. R., Ismail, R., Yusoff, Y., Cheng, L. K., Azmi, M. S. B. M., Natiq, H., & Happala Naidu, H. (2021). Software project management using machine learning technique—a review. *Applied Sciences*, *11*(11), 5183.
[64]. Manam, A., & Md. Ashfaq, S. (2022). Computational Thermo-Mechanical Modeling for Energy-Efficient Solid-State Metal Manufacturing Processes. *American Journal of Interdisciplinary Studies*, *3*(04), 579-618. https://doi.org/10.63125/ddg6mg97
[65]. Mangalampalli, S., Hashmi, S. S., Gupta, A., Karri, G. R., Rajkumar, K. V., Chakrabarti, T., Chakrabarti, P., & Margala, M. (2024). Multi objective prioritized workflow scheduling using deep reinforcement based learning in cloud computing. *IEEE access*, *12*, 5373-5392.
[66]. Manupati, V., Revanth, A., Srikanth, K., Maheedhar, A., & Sreekara Reddy, M. (2016). Real-time rule-based scheduling system for integrated delivery in a semiconductor manufacturing using evolutionary algorithm-based simulation approach. In *Artificial Intelligence and Evolutionary Computations in Engineering Systems: Proceedings of ICAIECES 2015* (pp. 981-989). Springer.
[67]. Md Abdur, R., & Aditya, D. (2026). Meta-Analysis of Distribution Cost Control in Retail Supply Chains: Insights from Data Modeling. *Journal of Sustainable Development and Policy*, *5*(01), 73-109. https://doi.org/10.63125/9q2c0036
[68]. Md Abubakar Siddique, A., & Aditya, D. (2023). Digital Twin Simulation for Optimizing Emergency Response and Evacuation Protocols in Large-Scale Manufacturing Environments. *American Journal of Advanced Technology and Engineering Solutions*, *3*(03), 121-161. https://doi.org/10.63125/8wzc3927
[69]. Md Abubakar Siddique, A., & Bhanu Prakash, S. (2025). Smart Occupational Safety Management Through IOT Sensor Networks, Machine Learning, and Real-Time Risk Assessment in Chemical Processing Plants. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 1958–1998. https://doi.org/10.63125/dynnzy25
[70]. Md Aminul, I. (2025). Impact of Predictive Analytics and Ensemble Learning on Operational Efficiency and KPI Forecasting in U.S. Engineering Firms. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2280–2322. https://doi.org/10.63125/r5s10176
[71]. Md Aminul, I., & Zakia, A. (2025). AI-Augmented Business Intelligence for Campaign Performance Optimization in U.S. Retail and e-Commerce: A Mixed-Methods Study of Marketing ROI. *American Journal of Scholarly Research and Innovation*, *4*(01), 732-774. https://doi.org/10.63125/h9j70a40
[72]. Md Asif Ali Sheak, A. (2025). Impact of Digital Twin Technology on Predictive Maintenance and Asset Lifecycle Management in Energy Infrastructure: A Quantitative Evaluation. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2323–2363. https://doi.org/10.63125/tg461a54
[73]. Md Asif Ali Sheak, A. (2026). A Comparative Study of AI-Enabled SCADA and DCS Integration for Smart Grid Optimization in Combined Cycle Power Plants (2018–2026). *American Journal of Data Science and Analytics*, *7*(03), 256-294. https://doi.org/10.63125/ppbq4489
[74]. Md Shahab, U. (2026). AI Based Quantitative Optimization Models for FMCG Supply Chain Efficiency in High-Demand Markets: A Linear Programming and Mixed-Integer Programming Approach. *American Journal of Scholarly Research and Innovation*, *5*(01), 66–108. https://doi.org/10.63125/nmr5ew86
[75]. Md Siam, T., & Md. Sultan, M. (2023). Utilizing Non-Contact GMR Sensors for Real-Time State Estimation of Aging Bulk Electric System Assets: A Strategy for Mitigating Failure Risks in Deteriorating Infrastructure. *American Journal of Advanced Technology and Engineering Solutions*, *3*(04), 167-208. https://doi.org/10.63125/ke5mte78
[76]. Md. Ashfaq, S., & Manam, A. (2023). Digital Twin Architecture for Predictive Control of Solid-State Additive Manufacturing Processes. *Review of Applied Science and Technology*, *2*(04), 266–307. https://doi.org/10.63125/tt00s684
[77]. Md. Jobayer Ibne, S., & Aditya, D. (2024). Machine Learning and Secure Data Pipeline Frameworks For Improving Patient Safety Within U.S. Electronic Health Record Systems. *American Journal of Interdisciplinary Studies*, *5*(03), 43–85. https://doi.org/10.63125/nb2c1f86
[78]. Md. Mainuddin, F. (2025). Advanced Engineering Materials Applications for Enhancing Durability and Lifecycle Performance of Steel Building Systems. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2406–2445. https://doi.org/10.63125/t9xvg986
[79]. Md. Mainuddin, F., & Palash Chandra, D. (2023). Advanced Computing–Based Modeling of Steel Connection Behavior and Stability Performance using ETABS And STAAD Pro. *American Journal of Advanced Technology and Engineering Solutions*, *3*(04), 42-86. https://doi.org/10.63125/xfkzrg56
[80]. Md. Sultan, M. (2026). Quantitative AI-Based Load-Flow and Fault-Prediction Modeling for Reliability Enhancement in Electrical Distribution Networks. *International Journal of Scientific Interdisciplinary Research*, *7*(1), 111–128. https://doi.org/10.63125/taejr363
[81]. Mishra, A. K., Singh, J., Kumar, G., Mufti, A., Singh, P., & Diwakar, M. (2024). Applying deep reinforcement learning for real-time resource allocation in agile project management. 2024 international conference on advances in computing, communication and materials (ICACCM),
[82]. Mohammad Robel, M., & Md Aminul, I. (2023). A Systematic Review of Cloud-Based Machine Learning Deployment Frameworks and Architectural Practices. *American Journal of Advanced Technology and Engineering Solutions*, *3*(01), 70-115. https://doi.org/10.63125/acyg9n80
[83]. Mst Kaniz, F. (2025). AI-Assisted Medical Records Management and EHR Workflow Optimization for Community Health Centres Serving Immigrant Populations. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2446–2485. https://doi.org/10.63125/nvn1yr86
[84]. Mst Kaniz, F. (2026). Food Safety Compliance and Nutritional Management Protocols in Institutional Healthcare Food Services: Evidence-Based Framework for U.S. Long-Term Care Facilities. *American Journal of Health and Medical Sciences*, *7*(01), 87-127. https://doi.org/10.63125/rvvzz976

[85]. Murad, M. D. H. R. (2025). Machine Learning–Based Consumer Behavior Prediction Models for E-Commerce Platforms: Enhancing Digital Financial Inclusion and Market Accessibility. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2078–2118. https://doi.org/10.63125/pnz32s94

[86]. Nadkarni, S., & Prügl, R. (2021). Digital transformation: a review, synthesis and opportunities for future research: S. Nadkarni, R. Prügl. *Management review quarterly*, *71*(2), 233-341.

[87]. Nguyen, S., Mei, Y., & Zhang, M. (2017). Genetic programming for production scheduling: a survey with a unified framework. *Complex & Intelligent Systems*, *3*(1), 41-66.

[88]. Nikooharf, M. H., Shirinbayan, M., Arabkoohi, M., Bahlouli, N., Fitoussi, J., & Benfriha, K. (2024). Machine learning in polymer additive manufacturing: a review. *International Journal of Material Forming*, *17*(6), 52.

[89]. Oliveira Farias, H. E., Sepulveda Rangel, C. A., Weber Stringini, L., Neves Canha, L., Pegoraro Bertineti, D., da Silva Brignol, W., & Iensen Nadal, Z. (2021). Combined framework with heuristic programming and rule-based strategies for scheduling and real time operation in electric vehicle charging stations. *Energies*, *14*(5), 1370.

[90]. Orhean, A. I., Pop, F., & Raicu, I. (2018). New scheduling approach using reinforcement learning for heterogeneous distributed systems. *Journal of Parallel and Distributed Computing*, *117*, 292-302.

[91]. Pan, Z., Wang, L., Dong, C., & Chen, J.-F. (2023). A knowledge-guided end-to-end optimization framework based on reinforcement learning for flow shop scheduling. *IEEE Transactions on Industrial Informatics*, *20*(2), 1853-1861.

[92]. Pareigis, S. (2023). Algorithmic foundations of reinforcement learning. International conference on autonomous systems,

[93]. Parsamehr, M., Perera, U. S., Dodanwala, T. C., Perera, P., & Ruparathna, R. (2023). A review of construction management challenges and BIM-based solutions: perspectives from the schedule, cost, quality, and safety management. *Asian Journal of Civil Engineering*, *24*(1), 353-389.

[94]. Pregina, K., & Ramesh Kannan, M. (2022). Stochastic project network scheduling technique for construction projects using GERT. In *Advances in Construction Management: Select Proceedings of ACMM 2021* (pp. 381-392). Springer.

[95]. Prity, F. S., Uddin, K. A., & Nath, N. (2024). Exploring swarm intelligence optimization techniques for task scheduling in cloud computing: algorithms, performance analysis, and future prospects. *Iran Journal of Computer Science*, *7*(2), 337-358.

[96]. Ran, Y., Hu, H., Zhou, X., & Wen, Y. (2019). Deepee: Joint optimization of job scheduling and cooling control for data center energy efficiency using deep reinforcement learning. 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS),

[97]. Rebeka, S., & Mst Kaniz, F. (2026). AI-Based Diabetes Risk Stratification Using Multi-Source Healthcare Data. *American Journal of Health and Medical Sciences*, *7*(01), 128-165. https://doi.org/10.63125/hvmrzr66

[98]. Risha, A. (2025). Impact Of Random Forest and Ensemble Methods on Infection Trend Forecasting: A Quantitative Evaluation Using Global Post Covid-19 Data. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2570–2610. https://doi.org/10.63125/b0yw2q91

[99]. Rjoub, G., Bentahar, J., Abdel Wahab, O., & Saleh Bataineh, A. (2021). Deep and reinforcement learning for automated task scheduling in large - scale cloud computing systems. *Concurrency and Computation: Practice and Experience*, *33*(23), e5919.

[100]. Sahu, S. K., Mokhade, A., & Bokde, N. D. (2023). An overview of machine learning, deep learning, and reinforcement learning-based techniques in quantitative finance: recent progress and challenges. *Applied Sciences*, *13*(3), 1956.

[101]. Satic, U., Jacko, P., & Kirkbride, C. (2019). Performance evaluation of scheduling policies for the DRCMPSP. International Conference on Analytical and Stochastic Modeling Techniques and Applications,

[102]. Satic, U., Jacko, P., & Kirkbride, C. (2022). Performance evaluation of scheduling policies for the dynamic and stochastic resource-constrained multi-project scheduling problem. *International journal of production research*, *60*(4), 1411-1423.

[103]. Sazzadul, I. (2023). Explainable Data Analytics in Financial Decision Systems: Enhancing Transparency in Big Data–Driven Credit Risk and Loan Approval Models. *International Journal of Scientific Interdisciplinary Research*, *4*(2), 31–67. https://doi.org/10.63125/twq4bw77

[104]. Schlenkrich, M., & Parragh, S. N. (2023). Solving large scale industrial production scheduling problems with complex constraints: an overview of the state-of-the-art. *Procedia Computer Science*, *217*, 1028-1037.

[105]. Shamsul, A. (2025). AI-Driven Condition Monitoring and Fault Detection in Electrical Power and Industrial Control Systems. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 1778–1809. https://doi.org/10.63125/csjs7238

[106]. Shamsul, A., & Md. Morshedul, I. (2025). The Role of Cloud-Native Infrastructures in Supporting Autonomous and Uncrewed Systems (UXS) in Operations. *Journal of Sustainable Development and Policy*, *4*(03), 82-125. https://doi.org/10.63125/vntbqq40

[107]. Shamsul, A., & Md. Sultan, M. (2022). Systematic Review of Electrical Engineering Contributions to Autonomous Power and Control Systems. *Journal of Sustainable Development and Policy*, *1*(02), 208-244. https://doi.org/10.63125/9g5sbf27

[108]. Shao, Y., Zhou, X., Yu, T., Zhang, T., & Chu, S. (2024). Pump scheduling optimization in water distribution system based on mixed integer linear programming. *European Journal of Operational Research*, *313*(3), 1140-1151.

[109]. Shi, Y., Lian, L., Shi, Y., Wang, Z., Zhou, Y., Fu, L., Bai, L., Zhang, J., & Zhang, W. (2023). Machine learning for large-scale optimization in 6G wireless networks. *IEEE Communications Surveys & Tutorials*, *25*(4), 2088-2132.

[110]. Shyalika, C., Silva, T., & Karunananda, A. (2020). Reinforcement learning in dynamic task scheduling: A review. *SN Computer Science*, *1*(6), 306.

[111]. Sun, L., Lin, L., Li, H., & Gen, M. (2019). Large scale flexible scheduling optimization by a distributed evolutionary algorithm. *Computers & Industrial Engineering*, *128*, 894-904.

[112]. Szwarcfiter, C., Herer, Y. T., & Shtub, A. (2023). Balancing project schedule, cost, and value under uncertainty: A reinforcement learning approach. *Algorithms*, *16*(8), 395.

[113]. Szymański, P. (2017). Risk management in construction projects. *Procedia engineering*, *208*, 174-182.

[114]. Tahmina Akter, R., & Md. Ashfaq, S. (2026). A Systematic Review of AI-Enabled Predictive Quality Control in Advanced Metal Manufacturing Systems. *American Journal of Interdisciplinary Studies*, *7*(01), 426-458. https://doi.org/10.63125/n73j2x72

[115]. Taru Binte, A. (2025). Impact of Automated Server and Database Monitoring Systems on ATM Network Uptime: A Quantitative Evaluation. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *1*(01), 2486–2527. https://doi.org/10.63125/qcr55n60

[116]. Taru Binte, A., & Iftekhar, A. (2022). Digital Payment Adoption as a Driver of Revenue Growth in Small Businesses: Evidence from Global Markets. *American Journal of Advanced Technology and Engineering Solutions*, *2*(04), 255-293. https://doi.org/10.63125/vfvzge86

[117]. Ulusoy, G., & Hazır, Ö. (2021). Stochastic project scheduling with no resource constraints. In *An Introduction to Project Modeling and Planning* (pp. 167-198). Springer.

[118]. Ustundag, A., & Cevikcan, E. (2018). *Industry 4.0: managing the digital transformation*. Springer.

[119]. Wang, J., Li, Y., Zhang, Z., Wu, Z., Wu, L., Jia, S., & Peng, T. (2024). Dynamic integrated scheduling of production equipment and automated guided vehicles in a flexible job shop based on deep reinforcement learning. *Processes*, *12*(11), 2423.

[120]. Wang, L., Pan, Z., & Wang, J. (2021). A review of reinforcement learning based intelligent optimization for manufacturing scheduling. *Complex System Modeling and Simulation*, *1*(4), 257-270.

[121]. Wang, X., Laili, Y., Zhang, L., & Liu, Y. (2024). Hybrid task scheduling in cloud manufacturing with sparse-reward deep reinforcement learning. *IEEE Transactions on Automation Science and Engineering*, *22*, 1878-1892.

[122]. Wang, Y., Ru, Z.-Y., Wang, K., & Huang, P.-Q. (2019). Joint deployment and task scheduling optimization for large-scale mobile users in multi-UAV-enabled mobile edge computing. *IEEE transactions on cybernetics*, *50*(9), 3984-3997.

[123]. Wang, Z.-J., Zhan, Z.-H., Yu, W.-J., Lin, Y., Zhang, J., Gu, T.-L., & Zhang, J. (2019). Dynamic group learning distributed particle swarm optimization for large-scale optimization and its application in cloud workflow scheduling. *IEEE transactions on cybernetics*, *50*(6), 2715-2729.

[124]. Wang, Z., Wang, T., Hu, H., Gong, J., Ren, X., & Xiao, Q. (2020). Blockchain-based framework for improving supply chain traceability and information sharing in precast construction. *Automation in construction*, *111*, 103063.

[125]. Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., & Kyek, A. (2018). Optimization of global production scheduling with deep reinforcement learning. *Procedia CIRP*, *72*, 1264-1269.

[126]. Waubert de Puiseau, C., Meyes, R., & Meisen, T. (2022). On reliability of reinforcement learning based production scheduling systems: a comparative survey. *Journal of Intelligent Manufacturing*, *33*(4), 911-927.

[127]. Wei, Y., Pan, L., Liu, S., Wu, L., & Meng, X. (2018). DRL-scheduling: An intelligent QoS-aware job scheduling framework for applications in clouds. *IEEE access*, *6*, 55112-55125.

[128]. Woo, Y.-B., Jung, S., & Kim, B. S. (2017). A rule-based genetic algorithm with an improvement heuristic for unrelated parallel machine scheduling problem with time-dependent deterioration and multiple rate-modifying activities. *Computers & Industrial Engineering*, *109*, 179-190.

[129]. Xiao, Y., Liu, J., Wu, J., & Ansari, N. (2021). Leveraging deep reinforcement learning for traffic engineering: A survey. *IEEE Communications Surveys & Tutorials*, *23*(4), 2064-2097.

[130]. Xiong, Y., Huang, S., Wu, M., She, J., & Jiang, K. (2017). A Johnson's-rule-based genetic algorithm for two-stage-task scheduling problem in data-centers of cloud computing. *IEEE Transactions on Cloud Computing*, *7*(3), 597-610.

[131]. Xu, Z., Chang, D., Sun, M., & Luo, T. (2022). Dynamic scheduling of crane by embedding deep reinforcement learning into a digital twin framework. *Information*, *13*(6), 286.

[132]. Yan, J., Huang, Y., Gupta, A., Gupta, A., Liu, C., Li, J., & Cheng, L. (2022). Energy-aware systems for real-time job scheduling in cloud data centers: A deep reinforcement learning approach. *Computers and Electrical Engineering*, *99*, 107688.

[133]. Zhang, G., Yang, Y., & Yang, G. (2023). Smart supply chain management in Industry 4.0: the review, research agenda and strategies in North America. *Annals of Operations Research*, *322*(2), 1075-1117.

[134]. Zhang, L., Tan, J., Han, D., & Zhu, H. (2017). From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug discovery today*, *22*(11), 1680-1685.

[135]. Zhang, L., Yang, Z., Xiao, Q., Guo, Y., Ying, Z., Hu, T., Xu, X., Khan, S., & Li, K. (2024). Distributed scheduling for multi-energy synergy system considering renewable energy generations and plug-in electric vehicles: A level-based coupled optimization method. *Energy and AI*, *16*, 100340.

[136]. Zhao, L., Fan, J., Zhang, C., Shen, W., & Zhuang, J. (2023). A DRL-based reactive scheduling policy for flexible job shops with random job arrivals. *IEEE Transactions on Automation Science and Engineering*, *21*(3), 2912-2923.

[137]. Zhou, G., Tian, W., Buyya, R., Xue, R., & Song, L. (2024). Deep reinforcement learning-based methods for resource scheduling in cloud computing: A review and future directions. *Artificial Intelligence Review*, *57*(5), 124.

[138]. Zhou, J., Zheng, L., & Fan, W. (2024). Multirobot collaborative task dynamic scheduling based on multiagent reinforcement learning with heuristic graph convolution considering robot service performance. *Journal of Manufacturing Systems*, *72*, 122-141.