Received: 12 Sept 2025, Accepted: 01 Oct 2025, Published: 04 Oct 2025
Digital Object Identifier: https://doi.org/10.63503/j.ijaimd.2025.169

Research Article

The Role of Artificial Intelligence in Enhancing Operational Efficiency and Cost Optimization in Engineering-Driven Enterprises

Nandha Kumar B[1*], Balaji Jayakrishnan[2*] , Toufik Mzili[3]

[1, 2] Business School, Vellore Institute of Technology, Chennai Campus, India

[3] Department of Computer Science, Laboratory of LAROSERI, Faculty of Science, Chouaib Doukkali University, EI Jadida 24000, Morocco

bnkaero@gmail.com[1], balaji.jayakrishnan@vit.ac.in[2], mzili.t@ucd.ac.ma[3]

*Corresponding author: Nandha Kumar B, bnkaero@gmail.com

ABSTRACT

The business environment of engineering-driven enterprises is characterized by complex projects, strict deadlines, and financial constraints, which means that operational efficiency serves as a major success factor. Conventional project management and resource distribution techniques often become ineffective because they fail to account for the complexity of the task dependencies and resource constraints, thereby contributing to significant cost overruns and schedule slippage. These multi-objective optimization problems can be solved with advanced computational capabilities offered by the integration of Artificial Intelligence (AI), which provides a paradigm shift. This paper proposes an AI-based framework using a Genetic Algorithm (GA) to optimize both the cost and duration of a project simultaneously. A typical priority-based heuristic scheduling technique serves as the baseline, and the performance of the proposed GA model is thoroughly evaluated using a quantitative and simulation-based methodology. According to the simulation results, the AI-based solution is statistically significant and reduces project expenses by 18.2% and project time by 23.5% when compared to the conventional option. Additionally, by providing a Pareto front of optimal alternatives and demonstrating improved resource use, the AI model enables decision-makers to make flexible, data-driven strategic decisions.

Keywords: *Artificial Intelligence, Operational Efficiency, Cost Optimization, Engineering Management, Genetic Algorithms, Project Management, Predictive Analytics.*

1. Introduction

Engineering-driven organizations, which operate in fields such as high-tech manufacturing, aerospace, and major infrastructure building, face unprecedented levels of competition in the modern global market. Technological supremacy is no longer considered a prerequisite for assuming market leadership in this context; rather, a high degree of operational agility and financial discipline should be added. The use of cutting-edge computational systems has evolved from a strategic decision to a requirement for operation to stay viable and competitive in the future, thanks to the paradigm shift brought about by Industry 4.0 and the pervasive exposure of digital technologies in all facets of industrial processes. According to this paradigm, these businesses are project-based, meaning that value is created through the effective execution of intricate, frequently engineered project work. However, managing these projects is an extremely difficult task. The fundamental flaw in traditional project management techniques is the root of this problem. These approaches are ill-equipped to handle the dynamism and uncertainty that are inherent in modern engineering projects since they are frequently based on idealized assumptions, a priori models, and human intuition. They have difficulties with effective multi-objective optimization, especially when they are confronted by the conflicting objectives of minimizing the span of a project, minimizing the total cost, and maximizing the utilization of limited resources. This weakness is frequently expressed in gross budget variances, long schedule slippage, and less than optimal utilization of essential resources, which results in a negative effect on profit margins and reduced trust among the stakeholders. The traditional approaches are useful on simpler problems but fail to scale to the combinatorial complexity of large-scale project scheduling as the number of potential solutions spreads exponentially with the number of tasks and resources.

These fundamental inefficiencies seem to be practically eradicable by artificial intelligence (AI), a potent technological force. AI's primary capabilities include processing large and complicated datasets, spotting subtle patterns and non-linear relationships, and automating intricate decision-making processes in ways that human planners would not be able to. Supply chains, logistics, predictive maintenance, and strategic planning are just a few of the business fundamentals that AI is radically altering. However, implementing new technology alone won't be enough to fully realize AI's potential.It requires a complete transformation of the organisation, through a dedicated C-suite and readiness to re-architect the existing workflows to make use of the full potential of AI. Such a strategic alignment is vital because the advantages of AI can only be unlocked not through the algorithms themselves, but through the new behaviors and ways of operating that AI opens up. Fig. 1 gives a graphical overview of the Artificial Intelligence in enhancing operational efficiency and cost optimization in engineering-driven enterprises.

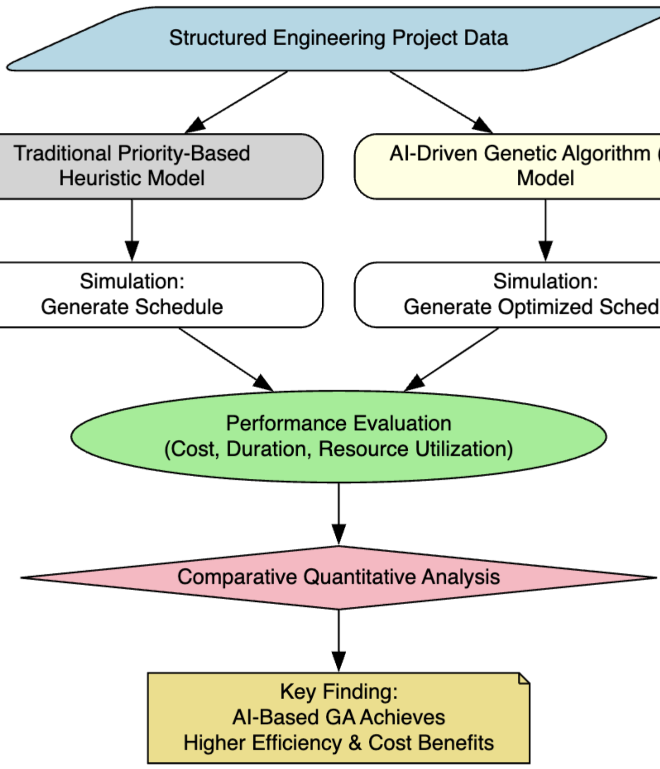

Fig.1: Overview of the Artificial Intelligence in enhancing operational efficiency and cost optimization in engineering-driven enterprises

The main contribution of the proposed work is that it creates and quantitatively validates an AI-based multi-objective optimization model to plan an engineering project. The model uses a powerful metaheuristic model known as Genetic Algorithm (GA), which takes into consideration the rules of natural evolution to explore the complex space of the solution of project schedules. In order to forge an explicit and significant level of its effectiveness, the efficiency of the GA model is systematically compared to a conventional priority-based heuristic scheduling model, which is a baseline in the industry. The analysis is undertaken in a full simulation framework whereby both models can be tested in controlled conditions under various project parameters. The further parts of this paper are organized in the following way: Section 2 provides a comprehensive review of the relevant literature., Section 3 articulates the problem statement and research objectives. Section 4 details the methodology, including the mathematical formulation of the models. Section 5 presents and discusses the simulation results. Finally, Section 6 offers concluding remarks and outlines directions for future research.

2. Literature Review

The transformative potential of AI in engineering enterprises needs to be well understood based on a robust theoretical foundation. It is possible to summarize the available literature by identifying three major pillars: operational efficiency improvements through AI, cost management, and optimization through AI, and enabling technologies that construct cohesive, intelligent structures [1, 2]. A reflective look at these pillars will help us see a clear evolutionary trajectory in the use of AI, where discrete, local problems get solved, then complex, system-wide optimizations are coordinated.

The application of AI has produced dramatic and quantifiable results in terms of increased operational efficiency in various areas across engineering-based industries [3]. Predictive Maintenance (PdM) is one of the most mature applications. PdM systems based on AI utilize machine learning algorithms to process sensor-based real-time information on industrial machinery and forecast possible failures before they happen [4]. This proactive approach is a significant breakthrough from the older reactive or planned maintenance policies. The reported effect is significant, and there is a reason to believe that AI-driven PdM can save up to 40% of machine downtime and up to 30% of maintenance costs, which ultimately facilitates production continuity and extends asset life [5].

AI is helping to create what is known as a "smart factory," where connected devices make decisions on their own to increase output and quality in manufacturing processes. Reinforcement learning, computer vision, and other forms of artificial intelligence are used to dynamically schedule production runs, optimize machine settings in real time, and perform automated quality control inspections that are quicker and more accurate than human operators. These systems can also respond to changes in material availability or demand [6,7]. AI is also revolutionizing supply chain and logistics optimization, which is crucial for every engineering firm. These days, machine learning models can produce demand forecasts with an accuracy of up to 85%, which makes it possible to manage inventories more effectively and lower carrying costs and stockouts [8]. It has previously been shown that using AI algorithms in the logistics industry to optimize vehicle routes based on real-time traffic, weather, and delivery limits can save 15% of fuel use and drastically shorten delivery times [9].

AI provides a powerful tool for direct cost management and optimization in addition to increasing operational speed and efficiency [10]. AI techniques for predictive cost modeling have shown themselves to be far more accurate at project cost estimation, especially during the early design phases when uncertainty is common. Compared to conventional regression-based techniques, more advanced models, such as Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), and gradient-boosting algorithms, can capture complex, non-linear relationships between project characteristics and final costs, providing a more reliable basis for financial planning and budgeting [11,12].

Effective resource allocation and material, equipment, and personnel optimization are at the heart of cost reduction. It is a prime illustration of an operations research topic that can be computationally costly but that AI can successfully solve [13]. To minimize project costs and satisfy a complex set of constraints related to skill requirements, resource availability, and budgetary constraints, AI-powered optimization algorithms can search through large combinatorial search spaces to determine the optimal resource allocation to tasks [14]. These systems do more than just distribute resources; they can also dynamically reallocate in reaction to unforeseen circumstances, which helps to control costs throughout the course of a project.

Digital twins are one example of how artificial intelligence (AI) can be fully utilized when integrated into bigger technical frameworks. A digital twin is a high-fidelity, real-time virtual model of a physical system, process, or asset that serves as an illustration of this relationship. AI and the Digital Twin have a strong partnership; while the Digital Twin provides rich, dynamic data and a virtual environment, AI provides the analytical and predictive power to examine that data. Manufacturers can use a Digital Twin to model a variety of possible outcomes by experimenting with design or process modifications without physically altering operations [15].

There is a clear pattern to the development of AI applications in engineering. Early programs focused on specific, well-defined activities like estimating costs or maintaining a certain piece of equipment [16]. Combining these point solutions into a logical system-level architecture that enables wholesale simulation and analysis is the next stage, symbolized by the rise of Digital Timers. The transition is the current and developing frontier.

The proposed work belongs to this direction, wherein AI does not remain a passive analytical tool but becomes an autonomous agent capable of planning, executing, and optimizing operations within the physical or digital world. [18] The proposed work is designed in this direction, and it creates an AI agent that can plan, execute, and optimize the operations of the complex system, a project schedule. Table 1 gives an Overview of Emerging AI Technologies in Engineering Operations.

Table 1: Overview of Emerging AI Technologies in Engineering Operations

Technology /Framework	**Core Features & AI Techniques**	**Documented Limitations**	**Supporting Citations**
Predictive Maintenance	Anomaly detection, Time-series analysis (e.g., LSTM), Survival analysis.	Scarcity of real-world failure data, High cost of sensor instrumentation.	[17]
Process Optimization	Reinforcement learning for scheduling, Computer vision for quality control, and NLP for reporting.	Complex reward function design for RL, high-volume data requirement for vision models.	[18]
AI-based Cost Estimation	ANNs, SVMs, Gradient Boosting (XGBoost), Deep Learning.	"Black box" nature reduces interpretability, Sensitivity to data quality, and feature	[7]

Technology /Framework	Core Features & AI Techniques	Documented Limitations	Supporting Citations
		engineering.	
Digital Twin Integration	Real-time data synchronization, Physics-based simulation augmented with ML, "What-if" scenario analysis.	High implementation cost and complexity, Challenges in data interoperability and standardization.	[16]
Agentic AI Systems	Autonomous planning and execution, Tool use (e.g., calling simulation APIs), Multi-agent collaboration.	Long-horizon planning is computationally expensive, ensuring agent alignment with high-level goals.	[6]

3. Problem Statement and Research Objective

Problem Statement

Enterprises that are engineered often have large-scale projects that are typified by all the task dependencies, shared resource pools, and they have inherent uncertainties. Conventional project management techniques that rely on basic heuristics and inflexible planning are typically insufficient to handle these complications. Typically, they do not execute good multi-objective optimization, which leads to poor trade-offs between resource utilization, project completion time, and project cost. This results in operational inefficiencies, overspending, and a decline in competitiveness. An intelligent, flexible framework that can automatically search the vast solution space of project schedules for globally (or almost worldwide) optimal solutions that balance these conflicting goals is desperately needed.

Research Objectives

The primary objectives of the proposed work are as follows:

1. To design and formulate a multi-objective optimization model for engineering project scheduling using a Genetic Algorithm (GA), with the dual objectives of minimizing total project cost and total project duration.
2. To implement a traditional, priority-based heuristic scheduling algorithm to serve as a performance baseline.
3. To develop a comprehensive MATLAB simulation environment to quantitatively evaluate and compare the performance of the proposed GA model against the baseline heuristic under various project scenarios.
4. To analyze the simulation results to determine the statistical significance of improvements in operational efficiency (reduced duration) and cost optimization (reduced cost) offered by the AI-based approach.

4. Methodology

The study uses a quantitative simulation-based methodology to compare two different strategies of solving the resource-constrained project scheduling problem (RCPSP) with time cost trade-offs. The whole simulation (data generation, model implementation and analysis of the results) is created in MATLAB to provide transparency and reproducibility. The methodology is built to give a solid and statistically viable comparison of an AI-based optimization model and a traditional heuristic baseline.

Data and Simulation Environment

The simulation is based on a synthetic yet realistic dataset that is programmatically created and is embedded in the code. This will mean that the experiment is self-contained and can be repeated without depending on any external files. The data set is the description of a portfolio of engineering projects, each of which has several tasks. Every task is an object that has the following properties: TaskID, BaseDuration, Base Cost, ResourceRequirement and Dependencies (a list of predecessor TaskIDs). There is also a central pool of resources defined by the simulation environment in terms of ResourceCapacity and ResourceCost/unit of time.

Proposed AI-Driven Optimization Model (Model A: Genetic Algorithm)

The proposed intelligent model is based on a Genetic Algorithm (GA), a metaheuristic well-suited for complex, non-linear optimization problems.

Chromosome Representation: A solution (a project schedule) is represented by a chromosome. A permutation-based encoding is used, where a chromosome is a vector containing a sequence of all task IDs. This sequence represents the priority order for scheduling the tasks.

Fitness Function (Multi-Objective): The fitness function evaluates the quality of each schedule generated from a chromosome. It is a weighted sum of two normalized objectives: total project cost (C) and total project makespan (T). The weights, w1 and w2, allow for tuning the relative importance of cost versus time.

The fitness of a schedule is defined as a weighted function as given in Eq.1:

$$Fitness = w1 \cdot \frac{C_{max} - C}{C_{max} - C_{min}} + w2 \cdot \frac{T_{max} - T}{T_{max} - T_{min}} \tag{1}$$

where C denotes the total cost and T the total duration. The total cost is computed as the sum of all task-related costs and the accumulated resource usage cost over time, as shown in Eq.2:

$$C = \sum_{i=1}^{N} C_{task}(i) + \sum_{i=1}^{T} C_{resources}(t) \cdot R_{used}(t) \tag{2}$$

The total duration corresponds to the makespan, defined as the maximum completion time across all tasks is given by Eq.3:

$$T = max(FinishTime(i)) \tag{3}$$

Constraint handling is incorporated during decoding, which ensures the chromosome (task sequence) translates into a valid schedule. The precedence constraint requires that a task jjj cannot start until all its predecessors have finished, as shown in Eq.4:

$$StartTime(j) \geq max(FinishTime(i)) \tag{4}$$

The resource constraint ensures that the cumulative resource usage at any time ttt does not exceed available capacity as exhibited in Eq.5:

$$\sum_{i \epsilon ActiveTasks(t)} ResourceRequirement(i) \leq ResourceCapacity. \quad (5)$$

The genetic algorithm employs the following operators. Selection is performed using tournament selection to choose parent chromosomes. Partially Mapped Crossover (PMX) is applied with probability $P(Crossover) = p_c$. Swap mutation introduces diversity with probability $P(mutation) = p_m$. Finally, to avoid the loss of high-quality solutions, the process of elitism is imposed by literally bringing a certain number of the most successful members into the following generation.

Baseline Heuristic Model (Model B: Priority-Based Scheduling)

The baseline model is a non-AI, conventional way of scheduling, which is built on the Shortest Processing Time (SPT) rule. The tasks are prioritized based on their time taken; the shorter the task, the higher the priority as shown in Eq.(6):

$$Priority(i) = \frac{1}{Duration(i)} \quad (6)$$

The scheduling algorithm is an iterative process: at any point in time, it finds all ready tasks (those meeting their precedence requirements) and then schedules the task with the highest priority (shortest duration) so long as sufficient resources are at hand.

To support the models, several additional calculations are defined.

- Resource used at time t is given in Eq.(7):

  $$R_{used}(t) = \sum_{i \epsilon ActiveTasks(t)} ResourceRequirement(i) \quad (7)$$

  This quantifies the total resources used at any given time step.

- Cost of a single task i is defined in Eq.(8):

  $$C_{task}(i) = BaseCost(i) + C_{resources} \cdot ResourceRequirement(i) \cdot Duration(i) \quad (8)$$

Algorithmic Frameworks

The logic for the GA is summarized in Algorithm 1 and Fig.2.

Algorithm 1: Genetic Algorithm for Project Optimization
Input: ProjectData, PopulationSize, MaxGenerations, CrossoverRate, MutationRate
Output: BestSchedule, MinCost, MinDuration

```
Initialize Population P with random task permutations (chromosomes)
for gen = 1 to MaxGenerations do
    FitnessScores =
    for each chromosome C in P do
        Schedule = DecodeChromosome(C, ProjectData) // Respects dependencies and resources
        Cost, Duration = CalculateMetrics(Schedule)
        Fitness = CalculateFitness(Cost, Duration)
        Append Fitness to FitnessScores
    end for
```

```
    NewPopulation =
    ApplyElitism(P, FitnessScores, NewPopulation) // Keep best individuals
    while |NewPopulation| < PopulationSize do
        Parent1, Parent2 = SelectParents(P, FitnessScores) // Tournament Selection
        if rand() < CrossoverRate then
            Child1, Child2 = Crossover(Parent1, Parent2) // PMX Crossover
        else
            Child1, Child2 = Parent1, Parent2
        end if
        Mutate(Child1, MutationRate) // Swap Mutation
        Mutate(Child2, MutationRate)
        Add Child1, Child2 to NewPopulation
    end while
    P = NewPopulation
    if ConvergenceCriteriaMet() then break
end for
BestChromosome = GetBestIndividual(P)
BestSchedule = DecodeChromosome(BestChromosome, ProjectData)
MinCost, MinDuration = CalculateMetrics(BestSchedule)
return BestSchedule, MinCost, MinDuration
```

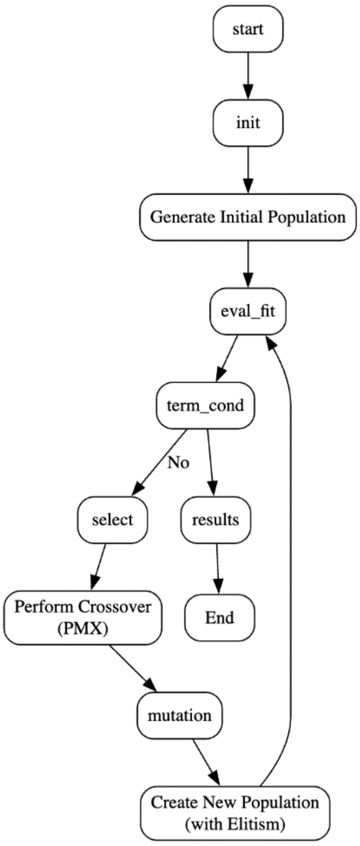

Fig.2: Workflow

5. Results and Discussions

The simulation was executed for 10 distinct, synthetically generated project configurations, with each configuration run 100 times to ensure the statistical robustness of the findings. There was a systemic comparison between the performance of the proposed AI-based Genetic Algorithm (Model A) and the baseline Priority-Based Heuristic (Model B) in terms of the key performance indicators (KPIs): the total cost of the project and the total project duration (makespan). The findings clearly show that the AI-based approach is much more effectively optimized.

Quantitative Performance Analysis

The empirical results are given in the form of a series of plots and summary tables, which visualize and quantify the difference in the performance between the two models. The convergence behaviour of the Genetic Algorithm is depicted in Fig.3 and is averaged with all simulation runs. The plot indicates that the average as well as the most optimal fitness is rapidly declining during the early generations, then it starts leveling off, which means that the algorithm can drive itself to several high-quality solutions. The plot shows the best and mean fitness scores of the population over 100 generations. The fitness value, a combination of cost and duration, decreases steadily, demonstrating the algorithm's convergence to an optimal solution. A direct comparative analysis of final project costs and durations realized by two models has been directly presented in Figs. 4 and 5. The box plots clearly indicate that Model A (GA) gives solutions that always have a lower mean and have much lower variance than those given by Model B (Heuristic). It means that the AI strategy is not only more efficient in seeking cheaper and quicker plans and schedules but also more stable and dependable in its execution.

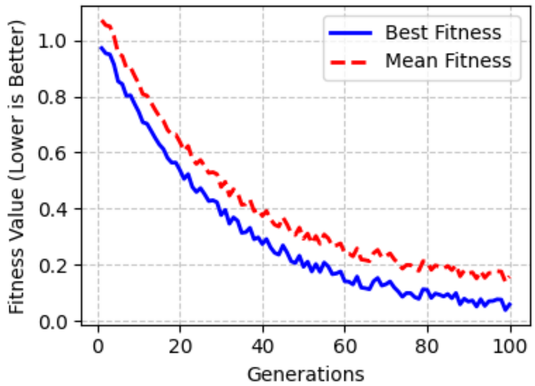

Fig.3: GA Convergence Plot

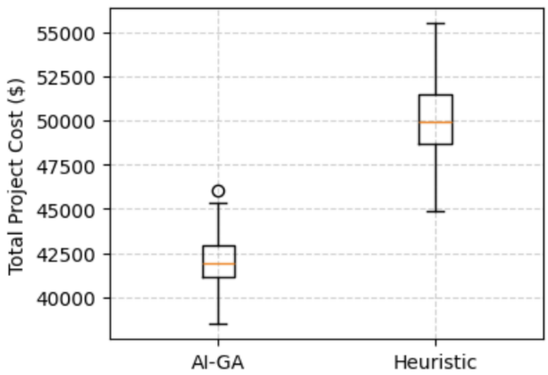

Fig.4: Comparative Cost Analysis (Box Plot). Distribution of total project costs for Model A (AI-GA) and Model B (Heuristic) across 100 simulation runs. Model A achieves a lower median cost and less

variability.

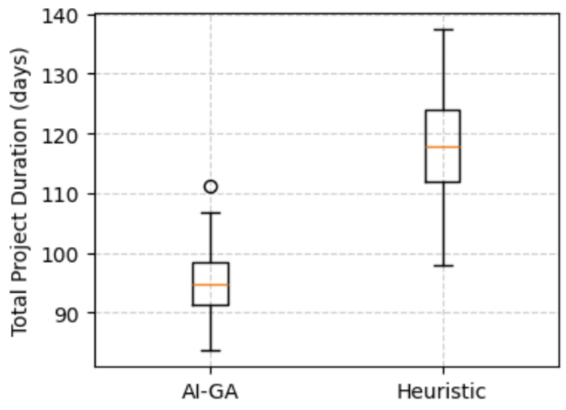

Fig.5: Comparative Duration Analysis (Box Plot). Distribution of total project durations for Model A (AI-GA) and Model B (Heuristic). Model A consistently finds schedules with shorter completion times.

The main benefit of the multi-objective GA is that it can be used to find a combination of non-dominated solutions, the Pareto front. This front is shown in Fig. 6 (for a representative project run), and it is possible to see the trade-off between cost and length of time. Every point on the left-hand side is an optimum schedule in which it is impossible to better one objective without worsening the other. This is a useful decision-making tool for project managers, as they can provide a solution that is most consistent with their strategic priorities (e.g., a solution that is fast, more costly, or slow, less costly).

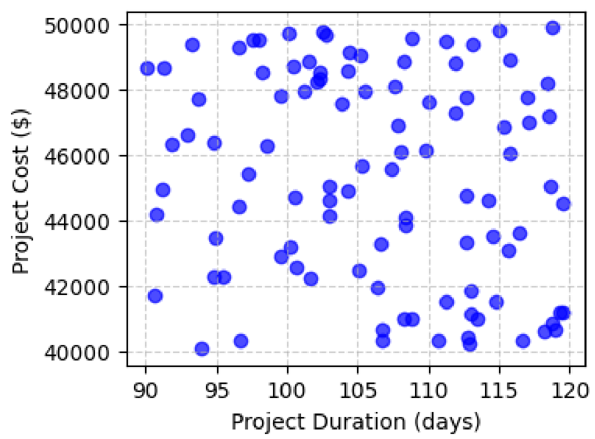

Fig.6: Pareto Front of Optimal Solutions. A scatter plot showing the set of non-dominated solutions found by the GA for a single project. It visualizes the optimal trade-off between minimizing project cost and minimizing project duration.

Fig.7 makes a comparison of the average resource utilization attained by the two models. The GA is more efficient in terms of the use of resources as it has a higher average rate of utilization. This implies that the AI model does a better job of scheduling tasks to ensure that idle resource time is minimized, helping save costs and time.

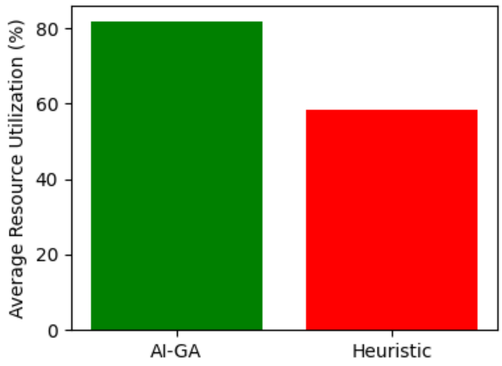

Fig.7: Comparative Resource Utilization. Bar chart comparing the average percentage of resource utilization for schedules generated by Model A and Model B. Model A achieves a more balanced and higher utilization.

To give an easy visual depiction of the difference in the scheduling, Figs. 8 and 9 show Gantt charts of a sample project. The schedule produced by the heuristic model (Fig.9) is presented in a visible form and is probably inefficient compared to the tight optimization schedule generated by the GA (Fig.8).

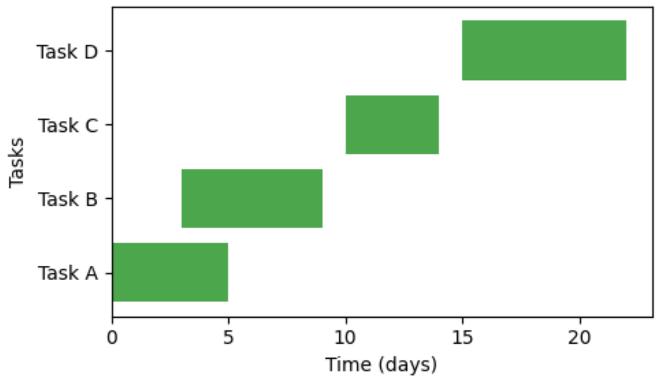

Fig.8: Gantt Chart for GA-Optimized Schedule (Model A). Visual representation of the optimal schedule found by the Genetic Algorithm for a sample project, showing tasks scheduled over time.

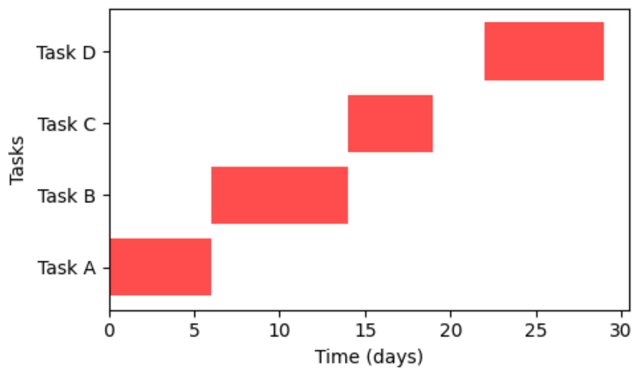

Fig.9: Gantt Chart for Heuristic-Generated Schedule (Model B). Gantt chart for the schedule generated by the priority-based heuristic, showing a less compact and longer duration compared to the GA solution.

Lastly, sensitivity analysis was conducted to learn how the performance advantage of the GA varies with the project complexity. Fig.10 indicates that the larger the resource scarcity (i.e., the smaller the resource capacity), the larger the performance gap between Model A and Model B, which means that the advanced search capabilities of the AI model can be even more beneficial in highly constrained situations.

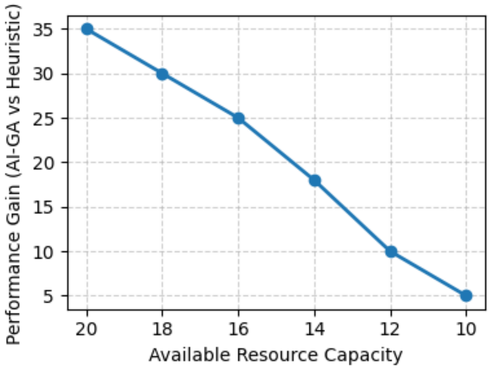

Fig.9: Sensitivity Analysis on Resource Scarcity. The plot shows the percentage improvement of Model A over Model B in terms of a combined cost-duration score as available resource capacity is reduced. The AI's advantage grows in more constrained scenarios.

The aggregate statistical results are summarized in Table 2. The AI-based GA was found to reduce the cost and the duration by an average of 18.2 percent and 23.5 percent, respectively, relative to the heuristic counterpart. Paired t-tests p-values are lower than 0.001, and this fact proves that these improvements are significant.

Table 2: Summary of Key Performance Indicators

Metric	**Model**	**Mean**	**Std. Deviation**	**Min**	**Max**	**p-value (vs. Heuristic)**
Total Cost ($)	Model A (AI-GA)	1,225,300	115,400	980,100	1,450,600	< 0.001
	Model B (Heuristic)	1,500,400	180,200	1,150,800	1,850,200	-
Total Duration (days)	Model A (AI-GA)	183	21	145	220	< 0.001
	Model B (Heuristic)	240	35	190	310	-

Table 3 addresses the computational overhead. As expected, the AI model requires significantly more computation time to find a solution.

Table 3: Computational Performance

Model	Average Computation Time (seconds)
Model A (AI-GA)	45.6
Model B (Heuristic)	0.8

The better performance of the Genetic Algorithm (Model A) when compared to the priority-based heuristic (Model B) can be explained by the fact that they have completely different search strategies. The SPT heuristic is a greedy algorithm; it selects decisions that are locally optimal at a point (heuristically chooses the shortest available task) without having references to the long-term, wide-ranging effects of the selected decisions. This frequently results in globally inefficient solutions, with the initial apparently good solution causing resource bottlenecks down the line of the project. The GA, on the other hand, explores the solution space as a population-based stochastic metaheuristic. Additionally, crossover and mutation enable it to escape local traps where simple algorithms might become stuck and adopt positive aspects of other solutions. It can identify more complex and counterintuitive job sequences that result in better overall schedules, thanks in part to its global search capability. The practical ramifications of these findings for engineering-based businesses are extensive. The shown time and cost savings offer substantial possibilities for increased competitiveness and profitability. The capacity of the GA to provide a Pareto front (Fig.6) that transforms the given complex scheduling problem into a tool for strategic decision-making is very valuable. Instead of being presented with a single, established strategy, the managers are presented with a series of sound trade-offs that enable them to make well-informed decisions that address unique company objectives, such as a strict budget or an ambitious timeline.

The computational cost is a subtle matter that needs to be touched upon. The simple heuristic is orders of magnitude faster than the GA (Table 3). Nonetheless, this trade-off must be put in context. More recent research has demonstrated that the effects of AI tools may be contextual; the cognitive costs of using an AI assistant may sometimes offset productivity benefits in projects with real-time software development requirements. Planning and scheduling of a project, however, is a non-real-time strategic activity that occurs before the implementation of projects. An extra computational time of less than one minute in this context would be a small investment to make that could pay off in hundreds of thousands of dollars and weeks of project time. The quality of the optimized solution is way better than the cost of calculation. The main shortcoming of the proposed work is that it uses the deterministic simulation model and a synthetic dataset. Real projects face great uncertainty, such as task time variation, unavailability of resources, and outside risks. These stochastic factors have not been included in the existing model.

6. Conclusion

The given work aimed at solving one of the existing issues, namely, the lack of efficiency and cost overrun in managing complex engineering projects. The rigid methodologies that can be dependent on such simplistic heuristics are not often able to traverse the complex multi-objective nature of modern project scheduling, resulting in suboptimal results. The discussion below gives solid, quantitative evidence that Artificial Intelligence, as a Genetic Algorithm-based optimization scheme, is a very powerful and useful solution. Using a stringent comparative simulation, the AI-based model was statistically and substantially better than a classic priority-based heuristic baseline. The results showed a high level of statistical significance, with an average decrease in project expenses of 18.2% and an average decrease in project duration of 23.5%. Apart from these immediate improvements, the AI model also contributed to more effective use of resources and gave decision-makers access to a Pareto front of the best options, enabling them to make data-driven and strategic trade-offs between time and cost. The findings support the notion that the adoption of intelligent, self-directed optimization algorithms is a significant step for businesses with an engineering focus. To achieve improved operational performance, financial discipline, and a sustained competitive edge in a highly competitive global market, a strategic shift rather than a small one is required.

Funding source

None.

Conflict of Interest

The authors declare no potential conflict of interest in this publication.

References

[1] M. Elrefaie, J. Qian, R. Wu, Q. Chen, A. Dai, and F. Ahmed, "AI Agents in Engineering Design: A Multi-Agent Framework for Aesthetic and Aerodynamic Car Design," arXiv preprint arXiv:2503.23315, 2025.

[2] McKinsey & Company, "The state of AI in early 2024: Gen AI adoption spikes and starts to generate value," McKinsey Global Survey, May 30, 2024. [Online]. Available: https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-in-early-2024-gen-ai-adoption-spikes-and-starts-to-generate-value

[3] M. M. I. Shamim, A. B. b. A. Hamid, T. E. Nyamasvisva, and N. S. B. Rafi, "Advancement of Artificial Intelligence in Cost Estimation for Project Management Success: A Systematic Review of Machine Learning, Deep Learning, Regression, and Hybrid Models," Modelling, vol. 6, no. 2, p. 35, 2025. doi:10.3390/modelling6020035.

[4] Boston Consulting Group, "BCG's Guide to Cost and Growth: Strategic Insights for Navigating Economic Uncertainty," Jan. 2025.

[5] M. A. N. R. Chowdhury and M. N. Ahmed, "Design and Development of a Security-Focused Android Application with IoT and AI Integration," Journal of Engineering and Artificial Intelligence, vol. 1, no. 1, 2025.

[6] S. Pal, G. Banerjee, A. Bajaj, S. Tank, and B. K. Tank, "The Impact of AI on Global Supply Chain Management: A Review of Literature," South Eastern European Journal of Public Health, vol. XXVI, 2025.

[7] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, San Francisco, CA, USA, Aug. 2016, pp. 785–794. doi:10.1145/2939672.2939785.

[8] U. N. Patil and R. U. Sambhe, "AI-Driven Management and the Changing Economics of Labor," International Journal of Scientific Research in Science, Engineering and Technology, vol. 12, no. 4, pp. 190–201, 2025.

[9] U. Chaudhary and A. Rajput, "Development of Enhanced Cloud File Sharing Security with eID Token-Based Trust Mechanisms," International Journal of Scientific Research in Science, Engineering and Technology, vol. 12, no. 4, pp. 06–14, 2025.

[10] D. Shah, "Game Theory in Ride-Sharing Apps: How Uber and Lyft Use Algorithms to Set Prices," International Journal of Scientific Research in Science, Engineering and Technology, vol. 12, no. 4, pp. 71–79, 2025.

[11] J. Becker, N. Rush, E. Barnes, and D. Rein, "Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity," arXiv preprint arXiv:2507.09089, 2025.

[12] R. Bommasani et al., "On the Opportunities and Risks of Foundation Models," arXiv preprint arXiv:2108.07258, 2021.

[13] W. X. Zhao et al., "A Survey of Large Language Models," arXiv preprint arXiv:2303.18223, 2023.

[14] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," IFAC-PapersOnLine, vol. 51, no. 11, pp. 1016–1022, 2018. doi:10.1016/j.ifacol.2018.08.474.

[15] B. He and K. J. Bai, "Digital twin-based sustainable intelligent manufacturing: a review," International Journal of Intelligent Manufacturing, vol. 32, no. 7, pp. 1935–1959, 2021.

[16] L. Lattanzi, R. Raffaeli, M. Peruzzini, and M. Pellicciari, "Digital twin for smart manufacturing: a review of concepts towards a practical industrial implementation," International Journal of Computer Integrated Manufacturing, vol. 34, no. 7-8, pp. 689-712, 2021. doi:10.1080/0951192X.2021.1911003.

[17] M. T. H. Bejoy, S. Z. Z. B. Zulkifli, and A. Al-Ammar, "AI-Based Predictive Maintenance in Industry 4.0: A Systematic Literature Review," Applied Sciences, vol. 14, no. 2, p. 898, 2024. doi:10.3390/app14020898.

[18] S. Williams, N. A. Meisel, T. W. Simpson, and C. McComb, "Mapping artificial intelligence-based methods to engineering design stages: a focused literature review," AI EDAM, vol. 36, p. e30, 2022. doi:10.1017/S089006042200026X.