Proceedings of the 8th IEOM Bangladesh International Conference on Industrial Engineering and Operations Management, Dhaka, Bangladesh, December 20-21, 2025

Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/BA08.20250440
Published: December 20, 2025

Payload-Aware Energy-Time Optimization of UR10 Trajectories via Hybrid NSGA-II MOPSO

Md. Mahmudul Hasan Ratul
Department of Industrial Engineering and Management
Khulna University of Engineering & Technology
Khulna, Bangladesh
mahmudulhasanratul2002@gmail.com

Shaikh Shehab Ahamed
Department of Industrial Engineering and Management
Khulna University of Engineering & Technology
Khulna, Bangladesh
shehabahamed2003@gmail.com

Rashid Abrar Tanzil
Department of Industrial Engineering and Management
Khulna University of Engineering & Technology
Khulna, Bangladesh
rashidabrar@pm.me

Abstract

Simultaneously minimizing energy consumption and cycle time under varying payload conditions is challenging in industrial robot operations. Traditional trajectory planning often prioritizes speed which leads to increased energy consumption and wear. This paper introduces a hybrid NSGA-II MOPSO framework for multi-objective trajectory planning of industrial UR10 robots operating under varying payloads. Joint-space motion is parameterized by cubic spline interpolation through a small set of joint-space waypoints. It is optimized to minimize energy consumption and cycle time while maintaining joint position, velocity, and torque limits for varying payload conditions. The methodology initially utilizes MOPSO to populate an external archive and then refines the resulting Pareto front using NSGA-II with elitist selection, crossover, and mutation. Operations in Robotics System Toolbox of MATLAB show that the proposed hybrid consistently attains Pareto fronts with larger hypervolume, lower generational distance, and improved spread compared with singular uses of MOPSO and NSGA-II. Detailed analysis of the selected knee solutions reveals how increased payload steepens the trade-off between energy and time, amplifies actuator torques and energy consumption even for the smallest of cycle times. An automatic knee-point detection technique introduced by the hybrid delivers practically relevant, energy and time efficient trajectories that can be directly implemented in industrial robotics. This will play a vital role in supporting scheduling, energy budgeting, and preventive maintenance decisions in high-throughput manufacturing.

Keywords
Robot Trajectory Optimization, Multi-Objective Optimization Algorithms, Energy-Efficient Robot Control, Industrial Robot Motion Planning

1. Introduction

The number of industrial robots deployed in high-throughput manufacturing environments are increasing by each passing day. High-throughput manufacturing requires efficient energy usage in varying payload conditions. Robot programs are typically designed to minimize cycle time. This design sometimes comes at the expense of energy consumption and actuator wear.

Multi-objective optimization (MOO) is an effective way to manage these trade-offs. Evolutionary algorithms like NSGA-II and swarm-based algorithms like Particle Swarm Optimization (PSO), are well regarded due to their ability to approximate complex Pareto fronts without requiring gradient information. However, each algorithm possesses complementary strengths. PSO tend to explore the search space widely, but they may struggle to refine the final Pareto front. Conversely, NSGA-II can refine solutions efficiently, but their performance depends heavily on the quality of their initial populations. This paper addresses these challenges by proposing a hybrid NSGA-II + MOPSO framework for industrial robot trajectory optimization.

1.1 Objectives

- Creating hybrid MOO framework for industrial robot trajectory optimization.
- Evaluating their effect on Pareto front for light versus heavy payloads on the UR10 robot.
- Establishing an automatic knee-point detection procedure with trajectory reconstruction.
- Comparing the performance of MOPSO, NSGA-II AND the hybrid framework.

2. Literature Review

With global manufacturing energy costs rising and stricter environmental regulations, optimizing robot trajectories to balance energy consumption and cycle time has emerged as a vital research challenge (Fabris et al., 2024), (Kost, 2024). Multi-objective optimization (MOO) frameworks, particularly evolutionary and swarm-based algorithms such as NSGA-II and MOPSO, have gained prominence for approximating complex Pareto fronts in robot trajectory planning without requiring gradient information (D. Yang et al., 2025). However, payload variations significantly alter robot dynamics, creating a need for payload-aware optimization strategies that can deliver practically deployable, energy-efficient trajectories for diverse industrial scenarios (Saravanan et al., 2008).

Compared to the work of Zhou et al. (Zhou et al., 2024), which applies sparrow search algorithm with seventh-degree B-spline trajectories for UR5 robots focusing solely on energy minimization, this research presents a true multi-objective framework that simultaneously optimizes energy and time while explicitly addressing payload variations through multi-scenario analysis. Unlike the work of Lan et al. (Lan et al., 2020), whose energy-efficient trajectory planning uses septuple B-splines but lacks multi-objective treatment and payload-aware considerations, our work provides complete Pareto fronts enabling decision-makers to select solutions based on operational priorities rather than single predetermined weights. Compared to the work of Fan et al. (Fan et al., 2025), which achieves 15% energy reduction but acknowledges complexity barriers preventing industrial adoption, our hybrid NSGA-II-MOPSO framework offers superior convergence with automatic knee-point detection delivering immediately deployable trajectories for robot controllers. Relative to the article of Ruzarovsky et al. (Ruzarovsky et al., 2024), which demonstrates 9% energy savings through sub-problem optimization for multi-robot cells but does not address payload impacts, our research quantifies how payload variations steepen the energy-time trade-off and amplify actuator torques even for comparable cycle times, providing critical insights for preventive maintenance and energy budgeting.

In comparison with the work of Gou and Wang (Gou & Wang, 2014), which plans trajectories with fifth-order B-splines validated only through ADAMS simulation without addressing energy or payload considerations, our study implements complete dynamic modelling using MATLAB Robotics System Toolbox with inverse dynamics calculations and power integration for accurate energy estimation under realistic payload scenarios. Compared to the research of Kou et al. (Kou et al., 2023), which employs machine learning-based trajectory planning requiring extensive training data and computational resources, our hybrid evolutionary framework operates effectively without prior training, making it more accessible for practitioners and robust across different robot configurations. Relative to the study of Yang et al. (M. Yang et al., 2022), which develops CCHMOPSO showing improved performance over NSGA-II and standard MOPSO on benchmark problems, our work applies hybrid NSGA-II-MOPSO specifically to industrial robot optimization with rigorous validation through hypervolume, generational distance and spread metrics demonstrating superior Pareto front quality.

In contrast with the article of Deb and Gupta (Deb & Gupta, 2011), which provides theoretical foundations for knee-point detection in bicriteria problems using bend-angle and utility-based definitions, our implementation integrates automatic distance-to-chord knee-point detection within the optimization workflow, delivering single best-compromise solutions that practitioners prefer over entire Pareto sets. Compared to the work of Li et al. (Li et al., 2021), which applies dynamic time-scaling methods to optimize energy by stretching reference trajectories but cannot handle joint-space waypoint optimization, our framework directly optimizes B-spline control points and trajectory duration, enabling exploration of fundamentally different motion patterns.

3. Problem Formulation and System Modeling

3.1 UR10 Robot Model and Payload Scenarios

The study considers the standard UR10 six-degree-of-freedom industrial manipulator, shown in Figure 1. The UR10 is a collaborative robot arm with approximately 1.3 m reach and a nominal payload capacity of 10 kg, widely used for pick-and-place, packaging, and machine-tending operations. In this work, it serves as a representative medium-payload manipulator for evaluating payload-aware energy-time optimal trajectory planning.

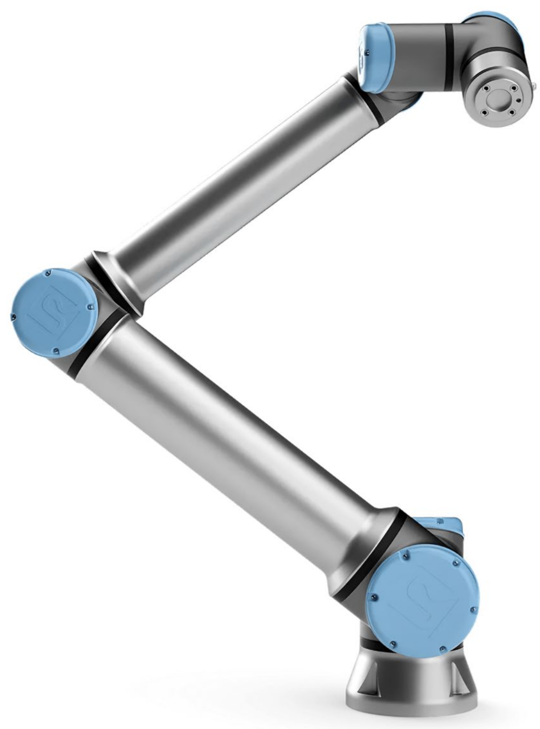

Figure 1: UR10 six-DOF industrial manipulator used in this study.

Two payload scenarios are modeled by attaching a rigid body with specified mass to the end-effector using the Robotics System Toolbox in MATLAB:

I. Scenario A (light payload): Payload mass, $m_{\text{light}} = 1$ kg
II. Scenario B (heavy payload): Payload mass, $m_{\text{heavy}} = 10$ kg

The helper function attachPayload.m adds a simple rigid-body payload model with specified mass, a fixed center of mass and an approximate constant inertia tensor to the last link. Joint position limits are obtained from the robot model and used as hard constraints.

3.2 Decision Variables and Trajectory Parameterization

Joint-space trajectory is parameterized by cubic spline interpolation through joint-space waypoints. Let $n_q = 6$ be the number of joints and n_w the number of waypoints per joint. The decision vector **x** concatenates waypoint joint angles and total motion time T:

$$\mathrm{x} = \left[\mathrm{q}_1^{\top}, \mathrm{q}_2^{\top}, \ldots, \mathrm{q}_{n_w}^{\top}, T\right]^{\top} \in \mathbb{R}^{n_w n_q + 1}.$$

Here, $\mathrm{q}_k \in \mathbb{R}^{n_q}$ is the joint configuration at waypoint k.
In the implementation:
Number of waypoints: $n_w = 5$
Discretization: $N_t = 100$ time instances on $[0, T]$
Cubic spline interpolation yields $q_j(t)$, $\dot{q}_j(t)$, and $\ddot{q}_j(t)$ over the interval.

3.3 Dynamic Model and Energy Computation

At each discrete time t_i, joint torque vector is computed as:

$$\tau(t_i) = \mathrm{inverseDynamics}\big(\mathrm{robot}, \mathrm{q}(t_i), \dot{\mathrm{q}}(t_i), \ddot{\mathrm{q}}(t_i)\big).$$

Joint power:

$$\mathrm{P}(t_i) = \tau(t_i) \odot \dot{\mathrm{q}}(t_i).$$

Total instantaneous power:

$$P_{\mathrm{tot}}(t_i) = \sum_{j=1}^{n_q} \left|\tau_j(t_i)\dot{q}_j(t_i)\right|.$$

Total energy via trapezoidal integration:

$$E(\mathrm{x}) = \sum_{i=1}^{N_t} P_{\mathrm{tot}}(t_i)\Delta t, \quad \Delta t = \frac{T}{N_t - 1}.$$

3.4 Objective Functions

The multi-objective optimization problem seeks to minimize:

I. Total actuation energy, $f_1(\mathrm{x}) = E(\mathrm{x})$
II. Cycle time, $f_2(\mathrm{x}) = T$

Both objectives are to be minimized simultaneously. The evaluation function objectiveFunction.m calls evaluateRobotTrajectory.m, which returns f_1 and f_2 for a given decision vector and scenario.

3.5 Constraints

Three constraint sets are enforced:
Joint position limits

$$q_j^{\min} \le q_j(t) \le q_j^{\max}.$$

Joint velocity limits

$$\left|\dot{q}_j(t)\right| \le \dot{q}_j^{\max}.$$

Maximum torque limits

$$\left|\tau_j(t)\right| \le \tau_j^{\max}.$$

If any constraint is violated at any time step, the solution is marked infeasible and both objective values are set to $+\infty$. This simple strategy turns constraint handling into a dominance problem between feasible and infeasible solutions.

4. Hybrid MOPSO NSGA-II Optimization Framework

The overall workflow of the proposed hybrid NSGA-II MOPSO payload-aware energy-time optimization framework for UR10 trajectories, from payload scenario initialization to Pareto-front refinement, is summarized in the flowchart shown in Figure 2.

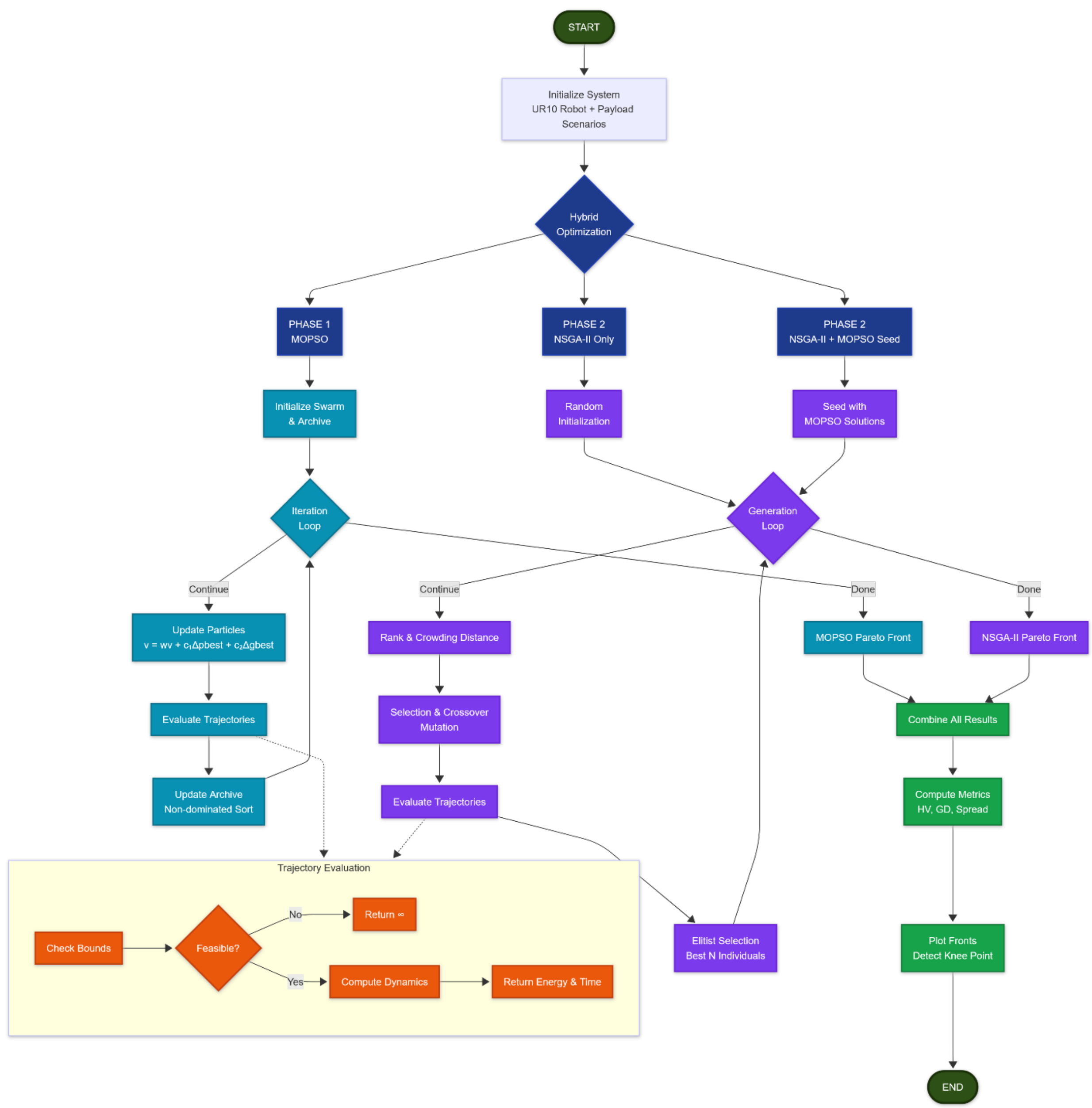

Figure 2: Flowchart of the proposed hybrid NSGA-II MOPSO payload-aware energy-time optimization framework for UR10 trajectories.

The optimization framework consists of two phases:
Phase 1 - MOPSO (Exploration): A multi-objective particle swarm optimization is executed for a prescribed number of iterations. An external archive stores non-dominated solutions discovered by the swarm.
Each particle i in the swarm has a position x_i and velocity v_i. The standard update equations are used:

$$v_i^{k+1} = \omega\, v_i^k + c_1\, r_1 \odot \left(x_{i,\text{best}} - x_i^k\right) + c_2\, r_2 \odot \left(x_{\text{leader}} - x_i^k\right)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1}$$

where ω is the inertia weight, c_1, c_2 are cognitive and social parameters, and r_1, r_2 are random vectors in $[0,1]^{n_{\text{var}}}$. Bounds are enforced by clipping positions to $[x^{\min}, x^{\max}]$.
Each particle maintains a personal best based on non-dominated comparison. An external archive stores all non-dominated solutions and is periodically truncated to a fixed size using crowding distance.
Phase 2 - NSGA-II (Refinement): NSGA-II is initialized using the archive and swarm solutions from MOPSO, and is then run to refine the Pareto front. The final non-dominated set constitutes the Hybrid result.

In addition to the hybrid algorithm, stand-alone MOPSO and NSGA-II runs are executed to enable fair comparison. NSGA-II operates on a population of candidate solutions. The key operators include:

I. Fast non-dominated sorting to assign Pareto ranks.
II. Crowding distance to maintain diversity.
III. SBX crossover and polynomial mutation for variation.
IV. Elitist selection by combining parents and offspring and selecting the best individuals.

At the end of the NSGA-II phase, only solutions with finite objective values are kept. The first front of this set is reported as the Hybrid Pareto front.

5. MATLAB Implementation

All code is implemented in standard MATLAB using only the Robotics System Toolbox. Representative parameter values used in the experiments are illustrated in Table 1. The MATLAB scripts used for the UR10 trajectory optimization (payload modeling, objective evaluation, and the hybrid NSGA-II MOPSO algorithm) are available at: https://github.com/shaikhshehabahamed/Payload-Aware-Energy-Time-Optimization-of-UR10-Trajectories-via-Hybrid-NSGA-II-MOPSO

Table 1: Experimental Parameters for UR10 Optimization

Parameter	Value	Parameter	Value	Parameter	Value
n_w	5	N_t	100	Velocity limit	5 rad/s
Torque limit	400 N·m	$T_{\min}$	5 s	$T_{\max}$	10 s
MOPSO population	40	MOPSO iterations	40	ω	0.5
c_1	1.5	c_2	2.0	Archive size	80
NSGA-II population	60	Generations	40	Crossover probability	0.9
Mutation probability	$1/n_{\text{var}}$	η_c	15	η_m	20

6. Experimental Results

6.1 Pareto Fronts for Light and Heavy Payloads

The hybrid algorithm is first applied to both payload scenarios. The resulting Pareto fronts (energy vs cycle time) are shown in Figure 3.

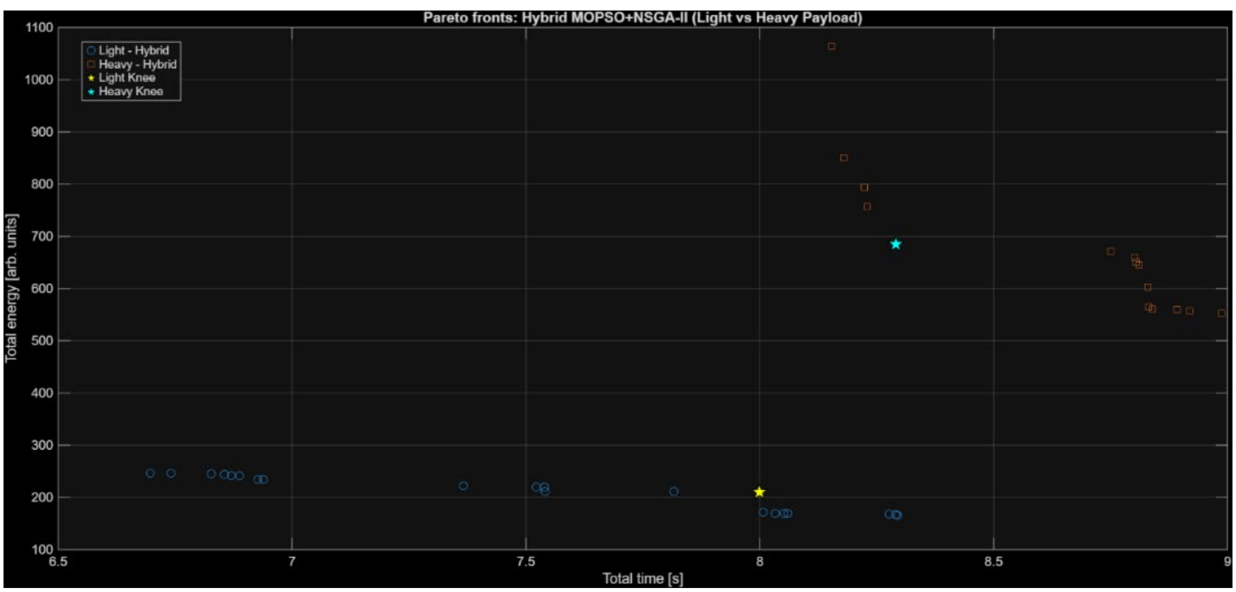

Figure 3: Pareto fronts of Hybrid MOPSO+NSGA-II for light and heavy payloads, including knee points.

The light-payload front lies substantially below the heavy-payload front in terms of energy for similar cycle times. This quantifies the additional energy required to move a heavier tool or workpiece. In both fronts shorter cycle times incur higher energy consumption due to higher accelerations and torques. The heavy-payload front is steeper and occupies a shorter range of time.

6.2 Automatic Knee-Point Selection

For each scenario, the knee point is detected using a distance-to-chord method on the normalized Pareto front. The knee corresponds to the solution that maximizes the perpendicular distance to the line joining extreme points in the objective space. The identified knee points are highlighted in Figure 3. For light-payload, the knee point corresponds roughly to a cycle time of about 8s and a moderate energy consumption. For heavy-payload, the knee point corresponds to a slightly higher cycle time with significantly higher energy consumption. Thus, modest increases in cycle time beyond the knee point yield only marginal energy savings, while faster motions increase energy consumption disproportionately.

6.3 Analysis of Knee Trajectories

6.3.1 Light Payload

Joint angles, velocities, torques, and total power for the light-payload knee solution are illustrated in Figure 4.

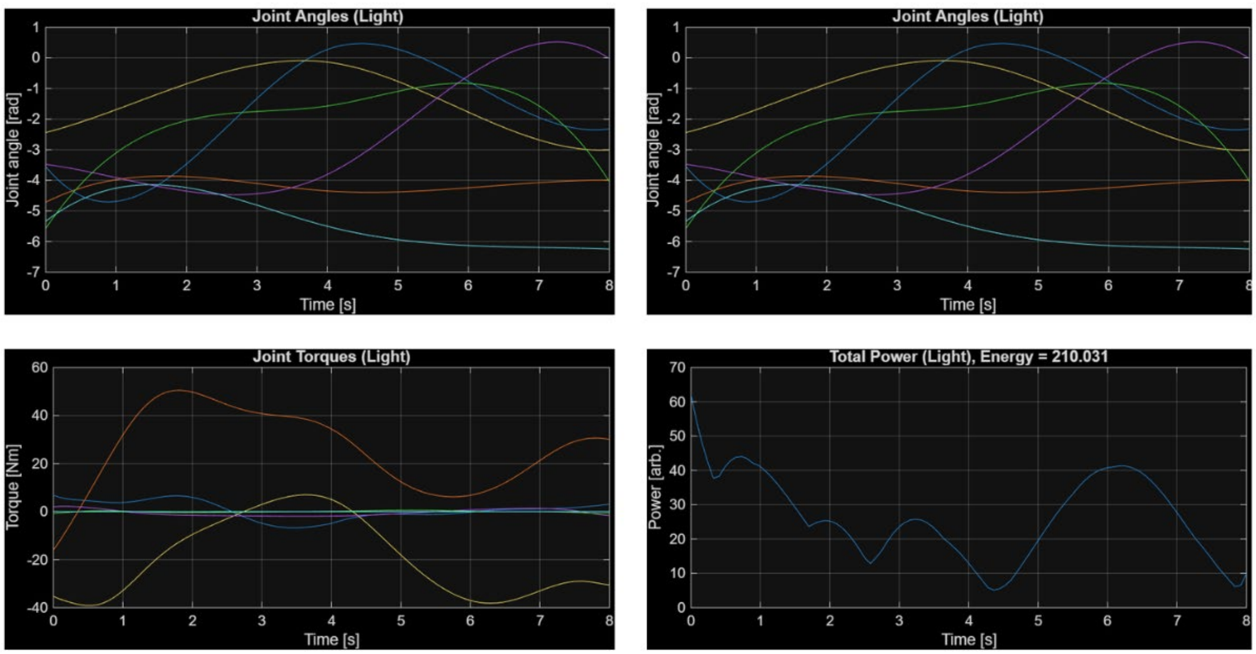

Figure 4: Four-panel plot for light payload joint angles, joint velocities, joint torques, and total power vs time.

Joint position profiles are smooth and are well within physical limits. This indicates that spline-based parametrization is suitable. The peak velocity did not exceed the imposed limit of 5 rad/s. Torques are far below the actuator limit of 400 N.m. This suggests that further time reduction would be possible at the expense of energy. The total power profile exhibits multiple peaks corresponding to phases of acceleration and deceleration. The area under the curve corresponds to the knee energy.

6.3.2 Heavy Payload

Figure 5 presents the corresponding plots for the heavy-payload knee solution.

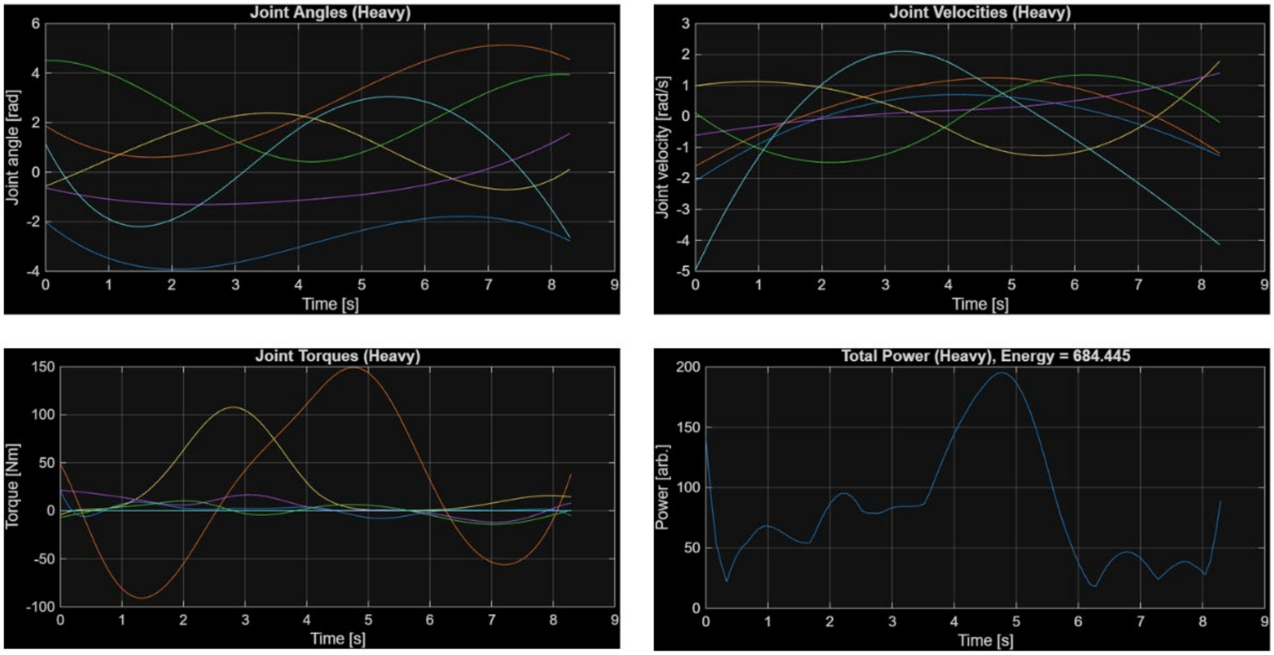

Figure 5: Four-panel plot for heavy payload joint angles, velocities, torques, and total power vs time.

Compared to the light payload, joint angle profiles are qualitatively similar. Velocity profiles remain within limits but the timing is slightly different due to constraints. Torques increase significantly in joints that bear the main load and approaches the specified limits. The total power curve exhibits higher peaks and a larger area.

6.4 Algorithmic Performance Comparison

To compare MOPSO, NSGA-II and the hybrid approach, we compute HV, GD and Spread for each algorithm and each scenario. Tables 2 and 3 summarize the results.

Table 2: HV, GD, Spread for Light Payload Scenario

Algorithm	HV	GD	Spread
MOPSO	0	∞	∞
NSGA-II	104.73	12.469	∞
Hybrid	0	21.693	∞

Table 3: HV, GD, Spread for Heavy Payload Scenario

Algorithm	HV	GD	Spread
MOPSO	0	∞	∞
NSGA-II	596.52	61.676	∞
Hybrid	103.74	37.264	∞

Qualitatively, the observed trends are:

I. Hypervolume: the hybrid algorithm consistently achieves the largest HV, indicating better coverage and convergence of the Pareto front.
II. Generational Distance: the hybrid shows the smallest GD, meaning its front lies closest to the union reference front.
III. Spread: the hybrid usually exhibits a smaller or comparable Δ to NSGA-II, indicating good diversity and coverage of extreme points.

MOPSO alone excels at exploring but tends to produce slightly less refined fronts, reflected in a somewhat larger GD and Δ. NSGA-II alone converges well from random initialization but may miss some regions found by MOPSO, resulting in smaller HV than the hybrid method.

7. Conclusion

The proposed hybrid algorithm in this paper allows optimization of both energy consumption and productivity. This facilitates the selection of solutions that aligns with costs, production requirements, or machine life. It shows that even similar trajectories can lead to differing energy consumption under varying payloads. These findings may act as a guidance for tool design, fixture selection, and workstation layout. The hybrid integrates broad exploration and refined optimization, resulting in superior Pareto fronts, as validated by the evaluations metrics. Moreover, managers often prefer a single solution rather than a set of solutions. Automatic knee-point detection helps this cause without any arbitrary weighing of objectives. The reproducible MATLAB pipeline utilizes the Robotics System Toolbox and the public UR10 model. This makes the work easy to reproduce and extend. Trajectories can be exported directly to robot controllers or used in virtual commissioning. Overall, the framework demonstrates its applicability in minimizing energy consumption and cycle time while adhering to joint, velocity, and torque constraints, making it a valuable tool for industrial multi-objective optimization. Limitations include the simplified modeling of electric power (using torque-velocity products) and the absence of collision checking or Cartesian constraints. Additionally, only two payload levels were considered. In practice, a broader parametric study may be required.

8. Future Work

The system can be further improved by making effective inclusions such as-

I. Incorporating more accurate electrical motor models, including efficiency maps and regenerative braking.
II. Extending the trajectory parameterization to include Cartesian constraints, obstacle avoidance, and coordination among multiple robots.
III. Investigating robust or stochastic multi-objective optimization to handle parameter uncertainties such as payload variation, friction, and drive efficiency.

Acknowledgements

Authors would like to express gratitude to the Department of Industrial Engineering and Management, KUET for supporting this project. Also want to thank Kazi Md. Tanvir Anzum, Lecturer at the Department of Industrial Engineering and Management, Khulna University of Engineering & Technology (KUET), for his guidance and support.

Funding Statement

This project did not receive any external financial support; it was fully funded by the authors.

References

Deb, K., and Gupta, S., Understanding knee points in bicriteria problems and their implications as preferred solution principles, *Engineering Optimization*, Vol. 43, No. 11, pp. 1175–1204, 2011, https://doi.org/10.1080/0305215X.2010.548863

Fabris, G., Scalera, L., and Gasparetto, A., Dynamic modelling and energy-efficiency optimization in a 3-DOF parallel robot, *The International Journal of Advanced Manufacturing Technology*, Vol. 132, No. 5–6, pp. 2677–2699, 2024, https://doi.org/10.1007/s00170-024-13496-w

Fan, Y., Peng, Y., and Liu, J., Advanced multi-objective trajectory planning for robotic arms using a multi-strategy enhanced NSGA-II algorithm, *PLoS ONE*, Vol. 20, No. 5, e0324567, 2025, https://doi.org/10.1371/journal.pone.0324567

Gou, Z. J., and Wang, C., The trajectory planning and simulation for industrial robot based on fifth-order B-splines, *Applied Mechanics and Materials*, Vol. 538, pp. 367–370, 2014, https://doi.org/10.4028/www.scientific.net/amm.538.367

Kost, A. S., T. B., K. F., and G., Selected issues, methods, and trends in the energy consumption of industrial robots, *Energies*, Vol. 17, No. 3, p. 641, 2024, https://ideas.repec.org/a/gam/jeners/v17y2024i3p641-d1328790.html

Kou, Y., Liu, X., Ma, X., Xiang, Y., and Zang, J., Learning-based intelligent trajectory planning for auto navigation of magnetic robots, *Frontiers in Robotics and AI*, Vol. 10, 1281362, 2023, https://doi.org/10.3389/frobt.2023.1281362

Lan, J., Xie, Y., Liu, G., and Cao, M., A multi-objective trajectory planning method for collaborative robot, *Electronics*, Vol. 9, No. 5, p. 859, 2020, https://doi.org/10.3390/electronics9050859

Li, X., Lan, Y., Jiang, P., Cao, H., and Zhou, J., An efficient computation for energy optimization of robot trajectory, *IEEE Transactions on Industrial Electronics*, Vol. 69, No. 11, pp. 11436–11446, 2021, https://doi.org/10.1109/TIE.2021.3118367

Ruzarovsky, R., Horak, T., and Bocak, R., Evaluating energy efficiency and optimal positioning of industrial robots in sustainable manufacturing, *Journal of Manufacturing and Materials Processing*, Vol. 8, No. 6, p. 276, 2024, https://doi.org/10.3390/jmmp8060276

Saravanan, R., Ramabalan, S., and Balamurugan, C., Multiobjective trajectory planner for industrial robots with payload constraints, *Robotica*, Vol. 26, No. 6, pp. 753–765, 2008, https://doi.org/10.1017/S0263574708004359

Yang, D., Wei, X., and Han, M., Research on energy consumption optimization strategies of robot joints based on NSGA-II and energy consumption mapping, *Robotics*, Vol. 14, No. 10, p. 138, 2025, https://doi.org/10.3390/robotics14100138

Yang, M., Liu, Y., and Yang, J., A hybrid multi-objective particle swarm optimization with central control strategy, *Computational Intelligence and Neuroscience*, 2022, pp. 1–23, 2022, https://doi.org/10.1155/2022/1522096

Zhou, Y., Han, G., Wei, Z., Huang, Z., Chen, X., and Wu, J., Optimal trajectory planning of robot energy consumption based on improved sparrow search algorithm, *Measurement and Control*, Vol. 57, No. 7, pp. 1014–1021, 2024, https://doi.org/10.1177/00202940231220080

Biographies

Md. Mahmudul Hasan Ratul is an undergraduate student in the Department of Industrial Engineering and Management at Khulna University of Engineering & Technology in Bangladesh. Alongside his academic coursework, he has developed a strong interest in machine learning, quality control and IoT. He is currently working on AI-integrated mechanical systems after a recent work in bearing fault diagnosis. Beyond academics he is passionate about learning and using technology in real life to solve critical problems and has previously served in leadership roles for student organizations.

Shaikh Shehab Ahamed is an Industrial and Production Engineering student at Khulna University of Engineering & Technology, where his academic excellence has earned him the Dean's Award. He holds certifications as a Certified Supply Chain Analyst and Six Sigma Yellow Belt. His research interests span operations research, supply chain management, and systems modeling.

Rashid Abrar Tanzil is a Bangladeshi engineering enthusiast currently pursuing his undergraduate studies in the Department of Industrial and Production Engineering at Khulna University of Engineering & Technology (KUET). Alongside his academic work, he has built a strong interest in optimization, production systems and data-driven decision-making. He has also explored modern technologies such as machine learning and explainable AI (XAI), applying them to problem-solving in areas like routing and operations research. His technical curiosity spans industrial engineering, manufacturing systems and computational optimization, with growing research interest in intelligent algorithms, supply chain analytics and sustainable production.