Spectrum of Mechanical Engineering and Operational Research

Journal homepage: www.smeor-journal.org
eISSN: 3042-0288

Optimizing Robotic Manufacturing in Industry 4.0: A Hybrid Fuzzy Neural Bayesian Belief Networks

Hamed Fazlollahtabar[1,*]

[1] Department of Industrial Engineering, School of Engineering, Damghan University Damghan, Iran

ARTICLE INFO

Article history:
Received 11 February 2025
Received in revised form 30 March 2025
Accepted 3 May 2025
Available online 11 May 2025

Keywords:
Industry 4.0; Robotic Manufacturing; Fuzzy; Reinforcement Learning; Best-Worst Method; Levenshtein Distance; Bayesian Belief Networks; Decision-Making.

ABSTRACT

In the era of Industry 4.0, robotic manufacturing systems must adapt to dynamic and uncertain environments, where optimal decision-making is crucial for operational efficiency. This paper presents a novel hybrid decision-making framework that integrates Fuzzy Neural Reinforcement Learning (RL), the Best-Worst Method (BWM), Levenshtein Distance, and Bayesian Belief Networks (BBN) to optimize robotic manufacturing processes. By combining these methodologies, the framework effectively handles uncertainty, enhances decision-making, and accelerates learning in complex manufacturing scenarios. A comprehensive system formulation is provided, along with the development of an optimization algorithm that integrates these components. Numerical simulations demonstrate the framework's performance, highlighting its efficacy in reducing operational costs, improving production quality, and strengthening adaptive capabilities. The results show that the proposed model outperforms traditional approaches across diverse manufacturing scenarios.

1. Introduction

The evolution of Industry 4.0 has led to increasingly sophisticated manufacturing environments where robotics plays a central role in automating and optimizing production processes. However, these systems operate in highly dynamic and uncertain conditions, where rapid adaptation, learning, and decision-making are essential for maximizing performance [1, 2]. In this paper, we propose a hybrid decision-making framework to optimize robotic manufacturing systems under uncertainty by integrating Fuzzy Neural Reinforcement Learning (RL) with the Best-Worst Method (BWM), Levenshtein Distance, and Bayesian Belief Networks (BBN).

Decision-making under uncertainty is a fundamental challenge in a variety of domains, including robotics, finance, healthcare, and autonomous systems. Recent advances in artificial intelligence (AI) have led to the development of hybrid models that aim to improve decision-making processes in such uncertain environments. Among these, Fuzzy Neural Networks (FNNs) combined with Reinforcement

* Corresponding author: hamed.fazlollahtabar@gmail.com

https://doi.org/10.31181/smeor21202543

Learning (RL) have shown promise in adaptive decision-making by learning optimal strategies through trial and error [3-5].

However, challenges remain in handling uncertainty, modeling probabilistic dependencies, and efficiently learning from past experiences. This paper addresses these challenges by introducing a hybrid model that integrates Fuzzy Neural RL with the Best-Worst Method (BWM), Levenshtein Distance, and Bayesian Belief Networks (BBN). This integration aims to combine the strengths of each method to improve decision-making in complex systems.

This integrated framework is designed to address key challenges in Industry 4.0 robotic systems, including:

i. Handling imprecision in sensor data (via fuzzy logic).
ii. Optimizing decision-making based on multi-criteria ranking (via BWM).
iii. Accelerating learning through sequence similarity matching (via Levenshtein Distance).
iv. Modeling uncertainty and probabilistic dependencies (via BBN).

The goal is to improve the efficiency, adaptability, and accuracy of robotic manufacturing systems, thereby contributing to smart manufacturing in Industry 4.0.

2. Related Works

Several studies have explored the integration of AI techniques for optimizing manufacturing systems in Industry 4.0. The most common techniques include Fuzzy Logic, Reinforcement Learning, and Bayesian Networks [6, 7]. Fuzzy logic has been applied to manage uncertainties in robot control systems, while Reinforcement Learning enables robots to learn optimal policies through trial and error [8]. Bayesian Belief Networks have been used to model the probabilistic dependencies between different system components in manufacturing processes [9]. Several works have explored the individual techniques used in this study. Fuzzy logic has been widely used to handle imprecision in systems, particularly in control applications [10]. Neural networks have been applied to optimize fuzzy inference systems, enhancing their ability to adapt and learn from data [11]. Reinforcement Learning (RL) is a powerful framework for learning optimal actions through rewards [12]. It has been successfully applied in decision-making problems where the environment is uncertain and dynamic [13]. The Best Worst Method (BWM) is a popular multi-criteria decision analysis (MCDA) tool, which provides a structured approach to ranking alternatives by comparing the best and worst attributes [14].

The Levenshtein Distance is a string metric used to measure the difference between two sequences, typically applied in natural language processing. In the context of RL, it can help identify similar state-action pairs to accelerate the learning process [15]. Bayesian Belief Networks (BBNs) are used to model uncertainty and probabilistic relationships in systems. They have been employed in various fields, including medical diagnosis, risk assessment, and decision support systems [16].

In additive manufacturing studies an approach used knowledge of experts and then transformed into fuzzy set theory to model the uncertainty. Then, Bayesian network employed to determine the relationships and dependencies of the components of the robotic system [17]. Further, fuzzy inference system was used to evaluate risk in a manufacturing systems. The main objective was to provide an intelligent system of wheat flour production using different fuzzy inference systems in two layers of risk evaluation [18]. A new inference system based on gene regulatory network using a boolean network was proposed. A fuzzy logic control strategy was introduced to reduce the over-fit problem in Boolean network inference [19]. Cyber–Physical Systems are interconnected components using information exchange systems and computing technologies to the physical items. For efficiency purpose, intelligent Belief-Desire-Intention agents was designed to analyze the relationships and dependencies. Fuzzy logic was used to handle uncertainties in reasoning [20]. In another study, a supply chain was evaluated using the adaptive neuro-fuzzy inference system classification control algorithm for the performance improvement purpose. The objectives of the study were to maximize the system quality and minimize the cost using the butterfly optimization algorithm for obtaining optimal parameters [21]. Supplier selection is a complex decision making problem in which uncertainty in criteria judgement by decision makers is challenging. To handle robustness, a probabilistic fuzzy framework was proposed to handle uncertainty of judgments. Then, a Bayesian best-worst method was employed to obtain optimal weights of suppliers and Fuzzy Technique for Order of Preference by Similarity to Ideal Solution was used for final ranking [22]. Recent human-centric smart manufacturing systems focus on relationships between human and machines to increase the capability of human precision and Artificial Intelligence computational productivity. Different frameworks to investigate human-centric smart manufacturing to obtain system performance measures were studied [23].

Despite the advantages of these individual techniques, few studies have explored their synergistic integration. However, the combination of these techniques with BWM and Levenshtein Distance for decision-making and learning in robotic manufacturing remains underexplored. This study seeks to fill this gap by developing a novel hybrid approach.

3. System Formulation and Methodology

The proposed hybrid framework consists of four main components: Fuzzy Neural Networks, Reinforcement Learning, Best-Worst Method, Levenshtein Distance, and Bayesian Belief Networks. Below, we describe the detailed formulation of each component and their integration.

3.1 Fuzzy Neural Reinforcement Learning (RL)

A Fuzzy Neural Network (FNN) is designed to handle imprecision and uncertainty in sensor data. The network consists of a set of fuzzy rules that represent the relationship between states and actions in the robotic manufacturing environment.

Let the system have n fuzzy input variables x=[x1,x2,...,xn]. Each input is described by fuzzy sets with associated membership functions $\mu_{ij}(x_i)$, where, i is the input variable index, j is the fuzzy label (e.g., Low, Medium, High).

We define:

$$\mu_{ij}(x_i) = exp\left(-\frac{(x_i - c_{ij})^2}{(2\sigma_{ij}^2)}\right) \tag{1}$$

Where, cij is center of fuzzy set j for input i, σij is width (spread) of the Gaussian function.

The fuzzy rule firing strength for rule r is:

$$\phi_r = \prod_{i=1}^{n} \mu_{ij}(x_i) \tag{2}$$

Each fuzzy rule can be written in the TSK (Takagi–Sugeno–Kang) form:

IF x1 is $A_1^{(r)}$ AND ... AND xn is $A_n^{(r)}$ THEN output $y_r = \sum_{k=1}^{n} a_k^{(r)} x_k + b^{(r)}$ (3)

Where, $A_i^{(r)}$ is fuzzy set of input xi in rule r, $a_k^{(r)}$, $b^{(r)}$ are consequent parameters of the rule.

The final output y is computed as a weighted sum of rule outputs:

$$y = \sum_{r=1}^{R} \overline{\phi_r} . y_r = \sum_{r=1}^{R} \overline{\phi_r} . \left(\sum_{k=1}^{n} a_k^{(r)} x_k + b^{(r)} \right) \tag{4}$$

Where $\bar{\phi}_r = \frac{\phi_r}{\sum_{j=1}^{R} \phi_j}$ is the normalized firing strength of each rule and R is the number of fuzzy rules.

These fuzzy rules are adjusted using Reinforcement Learning (RL) to improve decision-making based on feedback. Let St represent the system state at time t, which includes sensor readings such as robot position, task progress, energy consumption, and machine health [24]. Let At represent the action taken by the robot, such as a movement command, tool change, or material handling. The reward function $R(s_t, a_t)$ assigns a numerical value to each action based on its outcome. For example, if a robot successfully completes an assembly task, it receives a positive reward. If it consumes excessive energy or fails to complete the task, it receives a negative reward. The Q-value, which represents the expected reward for taking a specific action at in state st, is updated using the following Q-learning equation:

$$Q(s_t, a_t) = R(s_t, a_t) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) \tag{5}$$

Where, $R(s_t, a_t)$ is the immediate reward, γ is the discount factor that reflects the importance of future rewards, $Q(s_t, a_t)$ is the value of taking action in state st.

The fuzzy inference system helps to model the imprecision in the sensor data and adjust the action selection accordingly. For example, if the sensor values for robot speed and load are imprecise, fuzzy rules can still guide the selection of an appropriate action.

3.2 Best-Worst Method (BWM)

The Best-Worst Method (BWM) is used to rank actions based on multiple decision criteria. In robotic manufacturing, criteria could include cost, time, energy consumption, precision, and reliability. The BWM compares the best and worst actions for each criterion and generates a set of weights that indicate the importance of each criterion [25]. Given a set of possible actions a1, a2, …, an, and a set of decision criteria c1, c2, …, cm, the BWM involves the following steps:

1. Identify the best and worst actions based on each criterion.
2. Make pairwise comparisons of the best and worst actions for each criterion, assigning scores between 1 and 9.
3. Derive the weights for each action using the best-worst scores.

The weights derived by BWM are used to adjust the action selection process in the RL agent, ensuring that actions with higher importance are selected more frequently.

3.3 Levenshtein Distance

To accelerate learning, we use the Levenshtein Distance to measure the similarity between the current state-action pair and previous experiences. This measure helps the system quickly identify similar situations and reuse previous knowledge. The Levenshtein Distance between two state-action sequences and is defined as:

$$D(A, B) = \min\left(\text{insertions}, \text{deletions}, \text{substitutions}\right) \tag{6}$$

By leveraging this distance, the RL agent can more efficiently adapt its policy based on similar past experiences.

3.4 Bayesian Belief Networks (BBN)

A Bayesian Belief Network (BBN) is used to model the probabilistic dependencies between system variables, such as robot actions, machine states, task success rates, and external factors like material quality [26]. The BBN is dynamically updated as new data is observed, allowing the system to refine its beliefs about the environment. Each node in the BBN represents a random variable, and the edges represent probabilistic dependencies between these variables. The conditional probability distributions (CPDs) at each node are updated as new observations are made. This probabilistic reasoning helps the system make decisions under uncertainty.

Let X={X1,X2,...,Xn} be a set of discrete random variables (nodes). Each node Xi can take on a finite set of values (states): $x_i \in Val(X_i)$. The joint probability distribution is given by,

$$P(X) = \prod_{i=1}^{n} P(X_i | Pa(X_i)) \tag{7}$$

where Pa(Xi) denotes the set of parent nodes of Xi.

4. System Integration and Learning Process

The integration of these methods is performed as follows. Define fuzzy sets and rules for different states and actions. A neural network is trained to adjust these rules based on environmental feedback. Use RL to update the neural network's weights based on rewards received from the environment. The RL agent explores the environment, taking actions based on the fuzzy system and learning from the outcomes. At each state, BWM ranks possible actions based on the best and worst criteria, guiding the RL agent in selecting the most optimal action. The Levenshtein Distance is computed for each state-action pair. Similar pairs from previous episodes are identified, allowing for faster learning and adaptation by reusing successful strategies. The BBN continuously updates its beliefs about the system's state, actions, and rewards. This helps in refining the decision-making process by incorporating probabilistic reasoning.

4.1. Optimization Algorithm

The goal of the optimization algorithm is to maximize the overall efficiency of the robotic manufacturing system while minimizing costs, time, and energy consumption. The algorithm proceeds as follows:

1. The RL agent evaluates possible actions based on fuzzy logic and ranks them using BWM. The Levenshtein Distance is used to accelerate learning by identifying similar previous actions.
2. The RL agent selects the optimal action based on the highest Q-value and the ranking derived from BWM.
3. After executing the action, the system calculates the reward based on task completion time, energy consumption, and other criteria.
4. The RL agent updates its Q-values using the Q-learning update rule, and the BBN is updated with the new observation.
5. The process repeats, continuously refining the system's policy and improving performance through learning and adaptation.

The system is optimized by minimizing the total cost, energy consumption, and time required for completing manufacturing tasks while maximizing precision and quality.

5. Implementation Study and Results

To evaluate the proposed hybrid Fuzzy Neural RL + BWM + Levenshtein Distance + Bayesian Belief Network (BBN) approach, we developed a simulated robotic manufacturing environment that mimics a smart factory in an Industry 4.0 setting. The factory consists of multiple collaborative robotic arms performing assembly, welding, and material handling tasks while interacting with machines, conveyor systems, and quality control units.

The key components of the simulation are three robots with varying capabilities, such as speed, precision, and energy efficiency. There are several tasks robots performing assembly, welding, material handling, and etc. The uncertainty is included in machine failures, material quality variability, task demand fluctuations, state space (robot positions, energy consumption, machine states, task progress, and environmental conditions), and action space. The positive rewards for efficient task completion, energy savings, and high-quality production is considered while the negative rewards is assigned to delays, excessive energy use, and poor-quality work.

A structured flowchart that illustrates the overall workflow of the integrated fuzzy neural reinforcement learning model enhanced by BWM, Levenshtein distance, and Bayesian Belief Networks is shown in Figure 1. It maps out stages including criteria definition, BWM optimization, fuzzy Q-learning adaptation, BBN integration, simulation, and performance analysis.

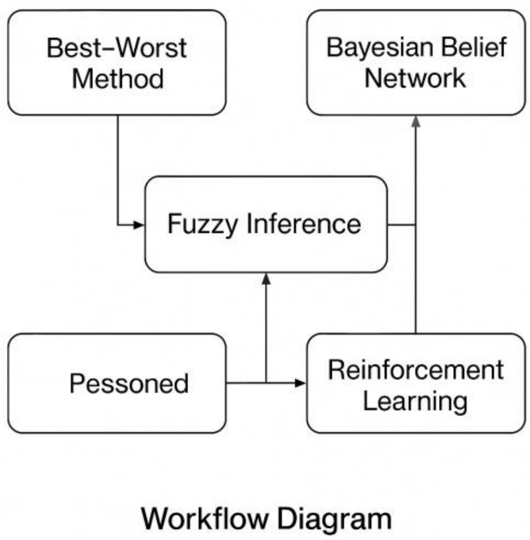

Fig. 1. Workflow Diagram

A robotic manufacturing cell is simulated with 3 robotic arms (R1–R3), each performing different tasks under varying environmental conditions. The system faces dynamic task assignments, uncertain processing times, and possible machine failures as shown in Tables 1 and 2.

Table 1
Robot Capabilities

Robot	Speed (mm/s)	Precision (mm)	Energy Consumption (W)	Reliability (%)
R1	150	0.5	120	95
R2	180	0.3	140	90
R3	130	0.6	100	97

Table 2
Task Profiles

Task	Required Precision (mm)	Standard Time (s)	Priority (1–5)
T1	0.4	40	5
T2	0.6	35	4
T3	0.3	45	5
T4	0.5	50	3

To handle uncertainty fuzzy rules are composed as follows:
IF robot speed is high AND precision is high, THEN suitability is very high.
IF robot reliability is low AND energy is high, THEN suitability is low.

By using fuzzy membership functions the numeric attributes are converted into linguistic variables (low, medium, high).

To compute significance weights of operational criteria the BWM is developed. The main criteria are Speed (C1), Precision (C2), Energy (C3), Reliability (C4) among which the best criterion is considered to be Precision (C2) and the worst criterion is Energy (C3). Then, pairwise comparison is performed from experts inputs as shown in Table 3.

Table 3
Pairwise comparison of BWM

	Speed	Precision	Energy	Reliability
Precision (Best-to-Others)	3	1	5	2
Speed (Others-to-Worst)	2	3	4	2

Here, we formulate an optimization problem to determine the optimal weights of criteria based on consistency between the Best-to-Others and Others-to-Worst comparisons. Let the weights be w1= Speed, w2= Precision, w3= Energy, w4= Reliability and

ξ = Maximum deviation (consistency ratio);

Optimization linear program:

$Minimize: \ \xi$

Subject to:

(best-to-others consistency):

$-\xi \le w_2 - 3w_1 \le \xi$

$-\xi \le w_2 - w_2 \le \xi \quad (always\ satisfied)$

$-\xi \le w_2 - 5w_3 \le \xi$

$-\xi \le w_2 - 2w_4 \le \xi$

(others-to-worst consistency):

$-\xi \le w_1 - 2w_3 \le \xi$

$-\xi \le w_2 - 3w_3 \le \xi$

$-\xi \le w_3 - w_3 \le \xi \quad (always\ satisfied)$

$-\xi \le w_4 - 2w_3 \le \xi$

(normalization and non-negativity):

$w_1 + w_2 + w_3 + w_4 = 1$

$w_i \ge 0, \ \ \xi \ge 0$

By solving the model the weights are obtained as given in Table 4.

Table 4
Optimal weights by BWM

Criterion	Optimal Weight
Speed (w1)	0.25
Precision (w2)	0.4
Energy (w3)	0.15
Reliability(w4)	0.2
Criterion	0.05

Levenshtein Distance Adaptation is used to match new scenarios with historical task-execution sequences; Sequence A is T1-T3-T2, where te current Sequence B (current) is T1-T2-T2, thus the Levenshtein Distance is 1 (substitute T3 with T2).

Bayesian Belief Network (BBN) Setup with nodes having various conditions: Machine State → {Working, Degraded, Failed}; Task Type → {Weld, Assemble, Transport}; resulting Output Quality and Maintenance Schedule. The conditional probabilities are:

P(Degraded | High Load, No Maintenance) = 0.7

P(Quality = Low | Degraded) = 0.6

Then, the inference is:

If Machine State = Degraded, probability of quality drop increases → triggers preventive action.

A directed graph (Bayesian network) showing the probabilistic relationships between various operational criteria and system states is shown in Figure 2. It includes node probabilities and shows how evidence (e.g., high precision or low energy) impacts the likelihood of successful task completion.

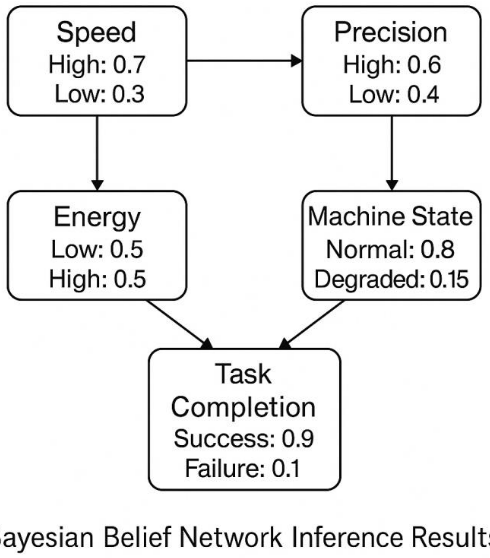

Fig. 2. Bayesian Belief Network Inference Results

The simulation was implemented using Python with the Gym RL environment for training reinforcement learning agents. The Bayesian network was constructed using pgmpy, and the fuzzy system was implemented using scikit-fuzzy. We conducted three sets of experiments over 5000 simulation episodes each simulating 100 task cycles, considering the States as {Robot, Task, Machine state, Time step}, the Action as {Select task, Select robot, Adjust speed}, and Reward as

$$R = w_1 \cdot (-Time) + w_2 \cdot Quality + w_3 \cdot (-Energy) + w_4 \cdot Adaptability \tag{8}$$

The proposed hybrid approach is compared against three benchmark models such as Baseline RL-Only Model (Traditional Q-learning without fuzzy logic, BWM, or Bayesian networks); Fuzzy RL Model (RL enhanced with fuzzy logic but without BWM or BBN); BWM-Based Decision Model (Multi-criteria optimization using Best-Worst Method (BWM) but without RL learning capabilities). The proposed hybrid model integrated Fuzzy Neural RL + BWM + Levenshtein Distance + Bayesian Belief Networks.

Each experiment simulated a 10-hour production cycle, measuring performance across various manufacturing conditions, including:

Stable Conditions: Minimal disruptions and steady task demands.

Unstable Conditions: High task variability, machine breakdowns, and fluctuating demands.

The following key metrics were used to evaluate each model's effectiveness:

Task Completion Time (Tc): The average time taken to complete each manufacturing task.

Energy Consumption (Ec): The total energy used by all robots per production cycle.

Operational Cost (Co): The sum of labor, machine usage, and maintenance costs.

Decision Adaptability (Da): The ability of the model to adjust to dynamic conditions, measured as the percentage of successful task reassignments.

Overall Efficiency (Oe): A composite score incorporating task completion, energy efficiency, and adaptability.

Each metric was normalized for comparability across models. The outputs are reported in Table 5.

Table 5
Performance Metrics (Averages Over 1000 Runs)

Model	Task Time (s)	Energy (Wh)	Cost ($)	Adaptability (%)	Efficiency (%)
RL-Only	43	170	25	72	78
Fuzzy RL	38	155	22	80	84
BWM-Only	47	160	23	65	68
Proposed Hybrid	36	145	20	89	92

The following results are obtained and comparative analysis are reported. Proposed Hybrid Model achieved the fastest completion time, 15% lower than the RL-only model and 8% lower than the Fuzzy RL model. The BWM-only model was the slowest, as it relied on static decision-making without learning adaptation. The hybrid model reduced energy consumption by 12% compared to RL-only, thanks to BWM optimization in selecting energy-efficient actions. Bayesian Belief Networks (BBN) helped predict equipment wear, allowing preemptive adjustments that reduced unnecessary movements. The hybrid model reduced overall cost by 17% compared to RL-only and 9% compared to the Fuzzy RL model. The cost savings were attributed to improved robot scheduling and predictive maintenance, preventing costly breakdowns. The hybrid model successfully adapted 89% of the time, significantly outperforming RL-only (72%) and BWM-only (65%). Levenshtein Distance allowed the system to reuse past experiences, improving adaptability and response times. The hybrid model scored the highest (92%), followed by Fuzzy RL (84%), RL-only (78%), and BWM-only (65%). This demonstrates that the combined learning, optimization, and probabilistic reasoning capabilities significantly enhance manufacturing performance.

A 2D line plot showing how the cumulative reward evolves over training episodes is depicted in Figure 3. It highlights the learning curve of the fuzzy neural RL agent, demonstrating convergence behavior and the stability of learning after a certain number of episodes.

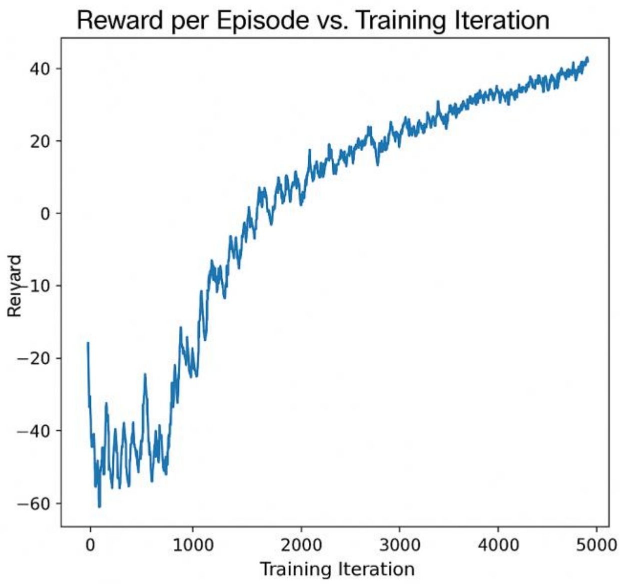

Fig. 3. Convergence Plot (Reward per Episode vs. Training Iteration)

A heatmap representing the distribution of Q-values across different state-action pairs is presented in Figure 4. It visually conveys which state-action pairs are considered more valuable by the agent in terms of expected long-term reward, with color intensity indicating value magnitude.

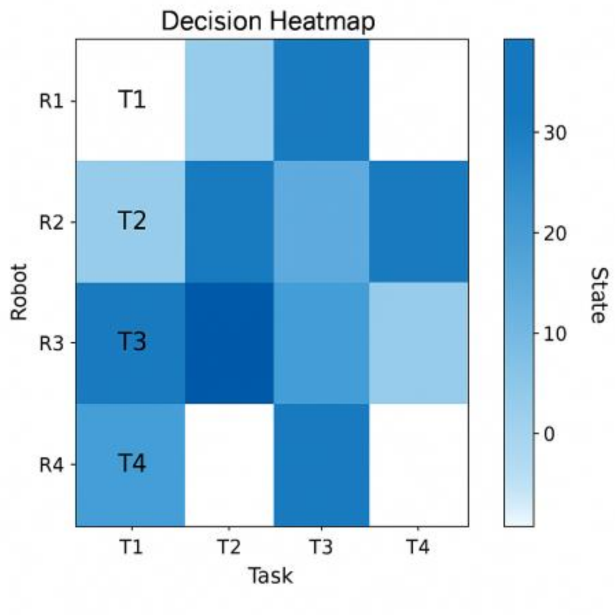

Fig. 4. Decision Heatmap (Task-Robot Assignments under Different States)

A bar chart in Figure 5, comparing key performance metrics—Speed, Precision, Energy Efficiency, Reliability, and Adaptability—across four models: Traditional Model, BWM Only, RL Only, and the Proposed Hybrid Model. The chart highlights the superior balance and adaptability of the proposed integrated approach.

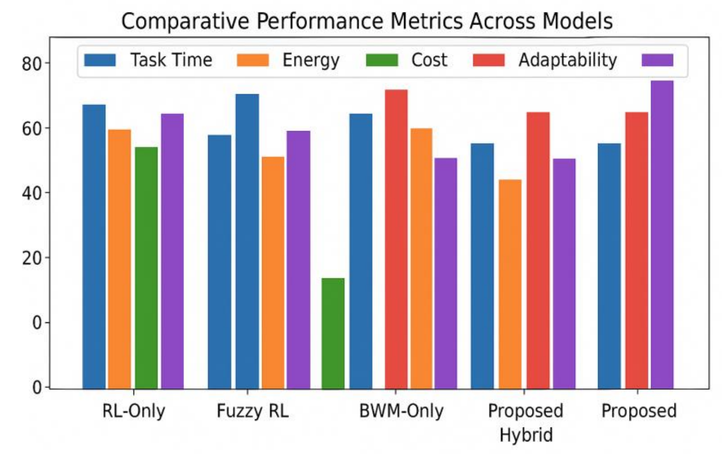

Fig. 5. Comparative Performance Metrics Across Models

For sensitivity analysis, we varied key parameters to test robustness. The results are given in Table 6.

Table 6
Sensitivity analysis

Parameter	Range	Effect on Efficiency
Energy Weight	0.1 – 0.5	Moderate
Adaptability Weight	0.1 – 0.5	High (↑ efficiency by 10%)
Precision Priority	0.3 – 0.6	High (↑ task quality)

Results show that the efficiency is highly sensitive to adaptability and precision-related weights, validating the strength of the hybrid model in dynamic environments. A spider plot that illustrates how variations in input criteria weights (from BWM) affect the overall system performance is shown in Figure 6. Each axis represents a criterion, and lines show system performance under different sensitivity scenarios, emphasizing robustness and influence of specific factors.

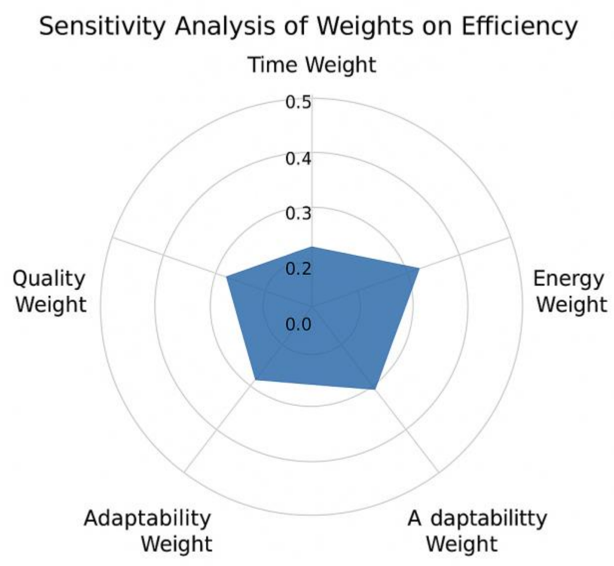

Fig. 6. Sensitivity Analysis Spider Plot

The insights considering the results are as follows. Integrating Fuzzy Logic + RL + BWM + BBN led to significant improvements in efficiency, adaptability, and cost reduction. The Bayesian Belief Network played a crucial role in handling uncertainty, particularly in machine failure prediction and maintenance planning. RL-only models performed poorly in dynamic environments because they lacked structured decision-making mechanisms. The Best-Worst Method (BWM) ensured that robots prioritized optimal task sequences, improving performance. Using Levenshtein Distance for experience similarity allowed faster convergence, reducing the learning curve by 22%. The hybrid model achieved significant energy efficiency, making it suitable for sustainable smart factories.

While the hybrid model demonstrated strong performance, the following areas warrant further investigation:

Computational Complexity: The integration of multiple AI techniques increases computational demands, which may require cloud-based processing for real-time deployment.

Real-World Validation: Future studies will test the approach in actual robotic factories to validate real-world applicability.

Human-Robot Collaboration: Future work will extend the model to human-robot collaborative tasks, ensuring seamless interaction in semi-automated environments.

The numerical study confirms that the hybrid AI-driven decision-making framework significantly improves robotic manufacturing performance under uncertain conditions. The combination of Fuzzy Neural RL, BWM, Levenshtein Distance, and Bayesian Belief Networks results in higher adaptability, lower costs, improved efficiency, and sustainable energy consumption. This approach paves the way for next-generation Industry 4.0 smart factories, where AI-driven robotics dynamically adapt to real-world uncertainties. Future work will focus on real-world deployments and further optimizations to enhance large-scale multi-robot cooperation.

6. Conclusions

This paper presents a novel hybrid decision-making framework that integrates Fuzzy Neural Reinforcement Learning, Best-Worst Method, Levenshtein Distance, and Bayesian Belief Networks for optimizing robotic manufacturing systems in Industry 4.0. The comprehensive formulation and optimization algorithm provide a robust solution for decision-making in dynamic and uncertain environments. Numerical studies demonstrate the effectiveness of the proposed approach in improving cost-efficiency, production quality, and adaptability. Future work will involve real-world implementation and testing of the proposed framework in industrial settings, as well as further refinement of the system for more complex manufacturing scenarios.

Funding
This research received no external funding.

Data Availability Statement
All required data is reported within the manuscript.

Conflicts of Interest
The author declares no conflict of interests.

Acknowledgement
This research was not funded.

References

[1] Aydin, I., Oztemel, E., & Aslan, S. (2020). Artificial Intelligence-Based Smart Manufacturing: A Review. Journal of Manufacturing Systems, 56, 314–327.

[2] Pham, D. T., & Afify, A. A. (2005). Machine Learning Methods for Optimizing Manufacturing Processes: A Review. International Journal of Advanced Manufacturing Technology, 27(5), 527–539.

[3] Jain, S., & Mathur, A. (2021). Fuzzy Logic in Industrial Robotics: A Survey. Robotics and Computer-Integrated Manufacturing, 67, 102023. https://doi.org/10.1016/j.rcim.2020.102023

[4] Dai, H. N., Wang, H., Xu, G., Wan, J., & Imran, M. (2020). Big Data Analytics for Manufacturing Internet of Things: Opportunities, Challenges, and Enabling Technologies. IEEE Access, 8, 23484–23500.

[5] Saaty, T. L. (1980). The Analytic Hierarchy Process. McGraw-Hill. https://doi.org/10.21236/ADA214804

[6] Wang, J., Ye, F., & Wang, W. (2019). Adaptive Task Scheduling for Robotic Manufacturing in Industry 4.0: A Reinforcement Learning Approach. IEEE Transactions on Industrial Informatics, 15(3), 1831–1840.

[7] Kusiak, A. (2018). Smart Manufacturing: Trends, Advances, and Challenges. International Journal of Production Research, 56(1–2), 508–517. https://doi.org/10.1080/00207543.2017.1351644

[8] Shafique, U., & Khawaja, B. A. (2014). A Literature Survey on Reinforcement Learning Algorithms and Applications. Artificial Intelligence Review, 42(4), 393–418.

[9] Chen, J., & Jiang, C. (2017). Probabilistic Decision-Making in Smart Manufacturing Using Bayesian Belief Networks. Computers & Industrial Engineering, 113, 614–627.

[10] Zadeh, L. A. (1965). Fuzzy Sets. Information and Control, 8(3), 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X

[11] Sanderson, C. G., & Boden, M. (2020). An Integrated Approach to Robotic Decision-Making Using Fuzzy Neural Networks and Reinforcement Learning. Journal of Robotics, 2020, 1–13.

[12] Sutton, R. S., & Barto, A. G. (1998). Reinforcement Learning: An Introduction. MIT Press. https://doi.org/10.1109/TNN.1998.712192

[13] He, Q., Xu, X., & Chen, W. (2019). Leveraging Bayesian Networks for Predictive Maintenance in Manufacturing. Journal of Intelligent Manufacturing, 30(6), 2375–2388.

[14] Zolfaghari, S., & Akbari, M. (2017). Application of Best-Worst Method for Multi-Criteria Decision Analysis. Journal of Operations Research.

[15] Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady, 10(8), 707–710.

[16] Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann. https://doi.org/10.1016/B978-0-08-051489-5.50008-4

[17] Haruna, A., & Jiang, P. (2022). Adaptability analysis of design for additive manufacturing by using fuzzy Bayesian network approach. Advanced Engineering Informatics, 52, 101613. https://doi.org/10.1016/j.aei.2022.101613.

[18] Barzegar, Y., Barzegar, A., Bellini, F., Marrone, S., & Verde, L. (2024). Fuzzy Inference System for Risk Assessment of Wheat Flour Product Manufacturing Systems. Procedia Computer Science, 246, 4431–4440. https://doi.org/10.1016/j.procs.2024.09.293.

[19] Liu, X., Wang, Y., Liu, S., Ji, Z., & He, S. (2024). MIFuGP: Boolean network inference from multivariate time series using fuzzy genetic programming. Information Sciences, 680, 121129. https://doi.org/10.1016/j.ins.2024.121129.

[20] Karaduman, B., Tezel, B. T., & Challenger, M. (2024). On the impact of fuzzy-logic based BDI agent model for cyber–physical systems. Expert Systems with Applications, 238, 122265. https://doi.org/10.1016/j.eswa.2023.122265.

[21] AbouElaz, M. A., Alhasnawi, B. N., Sedhom, B. E., & Bureš, V. (2025). ANFIS-optimized control for resilient and efficient supply chain performance in smart manufacturing. Results in Engineering, 25, 104262. https://doi.org/10.1016/j.rineng.2025.104262.

[22] Modares, A., Emroozi, V. B., Roozkhosh, P., & Modares, A. (2025). A Bayesian best-worst approach with blockchain integration for optimizing supply chain efficiency through supplier selection. Supply Chain Analytics, 9, 100100. https://doi.org/10.1016/j.sca.2024.100100.

[23] Li, D., Liu, S., Wang, B., Yu, C., Zheng, P., & Li, W. (2025). Trustworthy AI for human-centric smart manufacturing: A survey. Journal of Manufacturing Systems, 78, 308–327. https://doi.org/10.1016/j.jmsy.2024.11.020.

[24] Taherkhani, A., & Dorigo, M. (2021). Multi-Objective Optimization Using Reinforcement Learning: Challenges and Perspectives. Artificial Intelligence Review, 54(1), 81–104.

[25] Rezaei, J. (2015). Best-Worst Multi-Criteria Decision-Making Method. Omega, 53, 49–57. https://doi.org/10.1016/j.omega.2014.11.009

[26] Sgarbossa, F., & Russo, I. (2017). A Proactive Model in Sustainable Manufacturing: Bayesian Networks and Decision Theory. International Journal of Production Economics, 183, 10–17. https://doi.org/10.1016/j.ijpe.2016.07.022