This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

Closed-Loop Parameter Optimization for Robotic Machining Using Physics-Informed Machine Learning and Multiobjective Optimization

Guijun Ma, *Member*, *IEEE*, Zidong Wang, *Fellow*, *IEEE*, Weibo Liu, *Member*, *IEEE*, Zeyuan Yang, Desheng Huang, and Han Ding, *Senior Member*, *IEEE*

***Abstract*—In practical applications, the simultaneous optimization of numerous design parameters in time-consuming multi-objective optimization experiments is recognized as a significant bottleneck across various scientific and engineering disciplines. A prominent example is the optimization of machining parameters for achieving efficient and precise robotic belt grinding (RBG). This paper presents a closed-loop machining parameter optimization approach, which comprises two key stages: forward multi-task prediction and backward multi-objective parameter optimization. In the first stage, a physics-informed neural network (PINN) method is introduced, which integrates the multi-gate mixture-of-experts multi-task learning method with an RBG mechanism model to simultaneously predict material removal depth and averaged surface roughness. In the second stage, a powered multi-objective particle swarm optimization (MOPSO) method is developed, which combines a standard MOPSO method with a non-linear Powerball technique, to efficiently optimize the RBG machining parameters with a limited number of training iterations based on the learned PINN model. Two optimal machining parameter solutions are generated and recommended for the RBG machining process. The effectiveness and superiority of the proposed closed-loop parameter optimization method are validated through comparative experiments, which demonstrate its advantages in both coprediction accuracy and optimization efficiency.**

***Note to Practitioners*—This paper addresses the challenge of identifying robotic machining parameters that effectively balance machining efficiency and surface quality. Traditional trial-and-error methods for adjusting these parameters are both time-intensive and costly, given the vast number of possible combinations. To overcome these limitations, this paper proposes an intelligent optimization approach that leverages historical machining data to automatically determine optimal machining parameters. Our approach integrates artificial intelligence techniques with robotic machining mechanisms, ultimately recommending two sets of parameters for robotic machining. Preliminary experiments demonstrate the feasibility of this approach, but it has not yet been incorporated into a robotic machining system or tested in production. Future research will focus on dynamically optimizing robotic machining parameters by combining dynamic time-series signals with static machining parameters.**

***Index Terms*—Robotic belt grinding, parameter optimization, multi-task prediction, physics-informed neural network, multi-objective particle swarm optimization.**

This work was supported in part by the National Natural Science Foundation of China under Grant 52188102, Grant 62503185 and Grant 52505554; in part by the China Postdoctoral Science Foundation under Grant 2024M750991; and in part by the Postdoctor Project of Hubei Province of China under Grant 2024HBBHCXA010. (*Corresponding author: Zeyuan Yang*)

G. Ma, Z. Yang and H. Ding are with the State Key Laboratory of Intelligent Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mails: mgj@hust.edu.cn; yangzeyuan@hust.edu.cn; dinghan@hust.edu.cn).

Z. Wang and W. Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom (e-mails: Zidong.Wang@brunel.ac.uk; Weibo.Liu2@brunel.ac.uk).

D. Huang is with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: dsh@hust.edu.cn).

I. Introduction

Robots are increasingly recognized as emerging tools for machining large structural components and complex surfaces due to their large workspace, high flexibility, and low cost [1], [2]. Among the various robotic machining techniques, robotic belt grinding (RBG) is identified as an effective solution for workpiece finishing, garnering growing attention from both academia and industry [3], [4]. Achieving efficient and precise grinding remains a long-standing goal for RBG, which can be realized by adjusting grinding parameters to simultaneously enhance material removal ability and workpiece surface quality [5], [6].

Material removal depth (MRD) and averaged surface roughness (denoted as $\mathrm{R_a}$), widely regarded as representative indicators of material removal ability and workpiece surface quality, respectively, are frequently employed to assess the effectiveness of RBG. Nevertheless, the trial-and-error method for adjusting grinding parameters is both time-consuming and costly as there are thousands of possible parameter combinations [7]. A practical strategy involves designing a closed-loop optimization approach (see Fig. 1) that integrates forward MRD/$\mathrm{R_a}$ prediction with backward parameter optimization [8], [9].

In general, the forward MRD/$\mathrm{R_a}$ prediction methods for RBG can be categorized into mechanism methods, data-driven methods, and hybrid methods. Mechanism methods are centered on constructing mathematical or empirical models by analyzing the interaction pressure distribution between the tool and the workpiece. For MRD prediction, Preston's law [10] has been extensively applied to predict the MRD in RBG as a foundational model. Building on this, local nonlinear formulations have been developed to express MRD as a function of key process parameters, including belt speed, feed rate, and normal force [11]. The predictive accuracy of MRD has been further enhanced by incorporating additional physical variables such as the inclination angle and the half-width of the contact rectangle [6]. To improve the generalizability

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

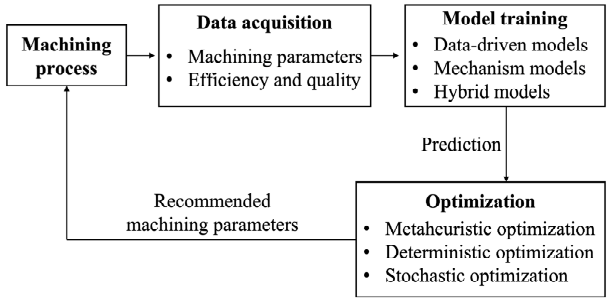


Fig. 1. Schematic of the closed-loop machining parameter optimization framework. The workflow begins with data acquisition during the machining process, collecting information on machining parameters along with corresponding efficiency and quality metrics. The acquired data is then used to train predictive models (such as data-driven, mechanism, and hybrid models) that accurately forecast machining outcomes. The predicted values are subsequently utilized to guide the optimization procedure, where metaheuristic, deterministic, or stochastic optimization algorithms are employed to optimize machining parameters to improve model performance.

across different grinding conditions, prediction models based on hierarchical mechanism have been proposed to characterize MRD behavior across under-grinding, theoretical grinding, and over-grinding modes [12]. In the context of $\mathrm{R_a}$ prediction, two primary modeling approaches have been explored [13], [14]. Proportional models based on chip thickness have been proposed to establish direct relationships between process variables and surface quality [13]. Concurrently, empirical polynomial models have been developed to capture the nonlinear effects of machining parameters on surface roughness [14]. Although mechanism methods offer valuable insights into the underlying physical processes and provide interpretable models, they often sacrifice predictive accuracy when exposed to real-world disturbances.

Data-driven methods, on the other hand, aim to develop black-box models that correlate grinding variables with MRD/$\mathrm{R_a}$ using machine learning techniques [15]–[19]. For example, a generalized regression neural network-based method has been proposed in [20] for predicting $\mathrm{R_a}$ in RBG, where the undeformed chip thickness, average abrasive grain size, and normal grinding force are used as input features. To further improve the prediction performance, the genetic algorithm has been employed to optimize the neural network for $\mathrm{R_a}$ prediction during the RBG process of aircraft engine blades [21]. In addition, a convolutional neural network-based multisensor fusion strategy has been introduced in [22] for MRD prediction, in which visual, acoustic, and tactile signals have been integrated to enhance prediction accuracy. While these methods typically achieve high prediction accuracy due to their strong learning capabilities, they heavily depend on large training datasets and struggle with poor interpretability.

Hybrid methods, which combine the strengths of mechanism and data-driven methods, offer a balance by maintaining prediction accuracy, generalization, and interpretability [23]. A promising example of a hybrid method is the use of physics-informed neural networks (PINN), which embed physical laws into neural networks to capture accurate solutions even with a limited number of training examples [24]. Based on the analysis conducted so far, the development of PINN appears to be a logical approach for addressing the limitations of both mechanism and data-driven methods, thereby enabling reliable and accurate MRD/$\mathrm{R_a}$ prediction in RBG.

It should be noted that MRD/$\mathrm{R_a}$ prediction methods typically require the training of two separate models for RBG, which neglects the potential correlation between MRD and $\mathrm{R_a}$ [4], [25], [26]. Multi-task learning (MTL) has emerged as a promising technique to address this limitation by enhancing the learning ability for each task through the sharing of knowledge across tasks. Examples of MTL approaches include the Shared-Bottom method, the tensor factorization method, and the multi-gate mixture-of-experts (MMoE) method. Among these, the MMoE method is particularly effective at balancing the correlation and discrepancy between multiple tasks while training a relatively small number of parameters [27], [28], which makes it a feasible and efficient approach for accurately copredicting MRD and $\mathrm{R_a}$, leveraging shared knowledge while maintaining task-specific learning [29].

To simultaneously minimize $\mathrm{R_a}$ and maximize MRD for RBG, backward parameter optimization methods are employed to automatically determine the optimal machining parameters by leveraging the learned coprediction model. Over the past few decades, evolutionary multi-objective optimization algorithms have gained widespread recognition for their ability to solve complex multi-objective optimization (MOO) problems by providing a diverse set of Pareto-optimal solutions, thereby supporting informed and flexible decision-making [30]–[33]. For instance, the non-dominated sorting genetic algorithm II (NSGA-II) has been applied to identify Pareto-optimal machining parameters in RBG [34]. Likewise, an adaptively controlled differential evolution algorithm has been utilized to optimize multiple performance aspects, including robotic kinematics, abrasive grit trajectory continuity, and tool wear control [35]. In addition, the multi-objective particle swarm optimization (MOPSO) algorithm has been adopted to determine optimal machining parameters in ultrasonic vibration-assisted grinding, taking into account both grinding forces and surface roughness [36]. Among these algorithms, the MOPSO algorithm is particularly notable for its ease of implementation and robust global search capabilities. MOPSO has also been successfully applied across a wide range of fields, including aerospace, chemistry, power systems, and mechanical manufacturing [37], [38]. Consequently, deploying the MOPSO algorithm to address the parameter optimization problem in RBG appears to be a promising approach. MOPSO provides a set of feasible solutions for decision-making, offering substantial potential to improve both grinding quality and efficiency.

It is worth mentioning that the MOPSO algorithm often suffers from slow convergence when generating a reliable Pareto front, resulting in high computational costs and delayed decision-making [39]. The powered stochastic gradient descent approach, which utilizes the non-linear Powerball technique to accelerate convergence, has been demonstrated to be effective both theoretically and experimentally in various applications [40]–[42]. Similar to stochastic gradient descent, the local and global updates for particle positions in the standard MOPSO algorithm are susceptible to gradient vanishing, which further

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

impedes its convergence. To address this issue, the introduction of the non-linear Powerball technique into the update process of the MOPSO algorithm emerges as a natural solution, significantly enhancing its convergence speed.

In this paper, a novel closed-loop parameter optimization approach is proposed for optimizing the machining parameters of RBG, consisting of two key stages: forward prediction and backward optimization. In the first stage, a PINN method, integrating an MMoE model with an RBG mechanism model, is introduced to accurately copredict MRD and $\mathrm{R_a}$. The MMoE model is employed for dual-task prediction, while the RBG mechanism model is embedded within the MMoE model to enhance the interpretability and accuracy of the predictions. An MOO problem is subsequently formulated based on the learned PINN model to simultaneously maximize MRD and minimize $\mathrm{R_a}$ in RBG. In the second stage, a powered MOPSO algorithm is developed to solve the MOO problem and identify a set of optimal machining parameters. To accelerate the convergence of the standard MOPSO algorithm, a non-linear Powerball technique is incorporated, and this proposed approach facilitates an efficient and precise RBG machining process.

The main contributions of this paper are outlined as follows.

1) A closed-loop machining parameter optimization framework is developed for the RBG machining process, consisting of i) a customized PINN for accurate and physically consistent coprediction of MRD and $\mathrm{R_a}$; and ii) a powered MOPSO algorithm for efficient machining parameter optimization based on the learned PINN model.
2) A customized PINN is designed by embedding an RBG-specific mechanism model into an MMoE architecture. By integrating domain knowledge with multi-task learning, this design substantially improves coprediction accuracy and model interpretability compared to the standard MMoE approach.
3) A powered MOPSO algorithm is introduced by integrating a non-linear Powerball transformation into the standard MOPSO update rules. The extension significantly improves the convergence speed and the diversity of Pareto-optimal solutions in machining parameter optimization.
4) The proposed closed-loop optimization framework is experimentally validated on a real-world robotic grinding setup. Comparative results show that the proposed method significantly outperforms baseline approaches in terms of both coprediction performance and optimization efficiency, which offers a practical solution for intelligent robotic machining.

The remaining sections of this paper are organized as follows, In Section II, the background of MTL, PINN, MOO, and the problem description of parameter optimization are presented. In Section III, the forward prediction method, the backward parameter optimization method, and their evaluation metrics are introduced. Section IV describes the self-developed RBG experimental platform and the experimental dataset. In Section V, the experimental results of prediction and optimization are presented and discussed in detail. Section VI draws the conclusions and addresses possible future research topics.

II. Preliminary

A. Background of MTL

Given l learning tasks $\left\{\mathcal{T}^i\right\}_{i=1}^{l}$ where all the learning tasks or a subset of them are related but not identical, MTL is dedicated to enhancing the learning capability of any model for $\mathcal{T}^i$ by leveraging shared knowledge across the l tasks [43].

Multi-task supervised learning (MTSL), a subset of MTL, focuses on supervised learning tasks aimed at constructing mapping functions from the feature space to the label space [43]. Given l supervised learning tasks $\{\mathcal{T}^i\}_{i=1}^{l}$, each task contains n_i labeled samples $\left\{(\mathbf{x}_j^i, \mathbf{y}_j^i)\right\}_{j=1}^{n_i}$, where $\mathbf{x}_j^i$ represents the input in a d-dimensional feature space $\mathcal{X}$, and $\mathbf{y}_j^i$ represents the corresponding label in a label space $\mathcal{Y}$. MTSL aims to simultaneously train l functions $\left\{f^i(\mathbf{x})\right\}_{i=1}^{l}$ for the l supervised learning tasks leveraging shared knowledge across the l tasks.

B. Background of PINN

PINN incorporates the knowledge of physical laws that govern the characteristics of a system into the learning process of neural networks [24]. These physical laws are typically expressed through partial differential equations, ordinary differential equations, integral equations, or algebraic equations. By embedding these laws, the learned model is guided to adhere more closely to the underlying physics or other domain-specific rules represented by the equations, thereby enhancing the model's reliability and interpretability.

Given a physical law $g\left(u(\mathbf{x}, t), \mathbf{x}, t\right) = 0$ governing a physical system, where $g(\cdot)$ is an operator applied to the real solution $u(\mathbf{x}, t)$, input data $\mathbf{x}$ and time t, a neural network $u_\theta(\mathbf{x}, t)$, parameterized by θ, is defined to approximate the solution $u(\mathbf{x}, t)$. The overall loss function of PINN is the weighted sum of a supervised loss function $\mathcal{L}_1 = \left\|u_\theta(\mathbf{x}, t) - u(\mathbf{x}, t)\right)\|^2 / n$ and a physics-informed loss function $\mathcal{L}_2 = g\left(u_\theta(\mathbf{x}, t), \mathbf{x}, t\right)$. The goal of PINN is to learn a prediction of $u_\theta(\mathbf{x}, t)$ by minimizing the overall loss function, thereby simultaneously approximating the real solution $u(\mathbf{x}, t)$ and satisfying the physical law.

Given l functions in MTSL, $\left\{f^i(\mathbf{x})\right\}_{i=1}^{l}$, a PINN aims to incorporate the physical law $g\left(f^1(\mathbf{x}), \cdots, f^l(\mathbf{x})\right) = 0$ into the training process of MTSL. The overall loss function for MTSL with PINN is defined as the weighted sum of an MTSL loss function $\mathcal{L}_1 = \sum_{i=1}^{l} \left\|\mathbf{y}^i - f^i(\mathbf{x})\right\|^2 / \sum_{i=1}^{l} n_i$ and a physics-informed loss function $\mathcal{L}_2 = g\left(f^1(\mathbf{x}), \cdots, f^l(\mathbf{x})\right)$. The goal of MTSL with the PINN is to train l prediction functions $\left\{f^i(\mathbf{x})\right\}_{i=1}^{l} : \mathbf{x} \rightarrow \mathbf{y}^i$ such that the predictions not only minimize the supervised loss $\mathcal{L}_1$ but also satisfy the physical law $g(\cdot)$ through $\mathcal{L}_2$.

C. Background of MOO

Consider the following MOO problem:

$$\min_{\mathbf{x}} \ \mathbf{f}(\mathbf{x}) = \left[f^1(\mathbf{x}), f^2(\mathbf{x}), \cdots, f^l(\mathbf{x})\right]^T \tag{1}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

subject to

$$B_i(\mathbf{x}) = 0 \quad \text{for } i = 1, 2, \ldots, p \tag{2}$$

$$C_j(\mathbf{x}) \leq 0 \quad \text{for } j = 1, 2, \ldots, q \tag{3}$$

$$x_k^{(L)} \leq x_k \leq x_k^{(U)} \quad \text{for } k = 1, 2, \ldots, n \tag{4}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_d]^T$ represents the vector of decision variables, and $\mathbf{f}(\mathbf{x}) = [f^1(\mathbf{x}), f^2(\mathbf{x}), \cdots, f^l(\mathbf{x})]^T$ denotes l objective functions to be minimized simultaneously. $B(\cdot)$ and $C(\cdot)$ are the equality constraints and inequality constraints, respectively. $x_k^{(L)}$ and $x_k^{(U)}$ are the lower and upper bounds on the decision variable x_k, respectively.

MOO algorithms are utilized to solve the optimization problem described above, where two or more objective functions are typically in conflict. The primary goal of MOO algorithms is to identify a set of solutions that effectively balance the various objectives, referred to as Pareto-optimal solutions [44]. In contrast to single-objective optimization algorithms, MOO algorithms offer a more comprehensive, flexible, and insightful approach to addressing optimization problems, which is particularly advantageous in complex scenarios involving multiple conflicting objectives, where trade-offs among objectives need to be carefully analyzed and resolved.

D. Problem Description

In the RBG machining process, MRD and $\mathrm{R_a}$ are strongly influenced by machining parameters, including normal force, robotic feed speed, belt linear speed, and equivalent radius of abrasive grains. To determine the optimal machining parameters, MTSL with a PINN is employed to copredict MRD and $\mathrm{R_a}$, while MOO algorithms are used to address the conflicting objectives of minimizing $\mathrm{R_a}$ and maximizing MRD. The machining parameter optimization problem, incorporating the PINN prediction model, can be formulated as follows:

$$\min_{\mathbf{x}} \ \hat{\mathbf{y}}_{\mathrm{R_a}} = f^1(\mathbf{x}) \tag{5}$$

$$\max_{\mathbf{x}} \ \hat{\mathbf{y}}_{\mathrm{MRD}} = f^2(\mathbf{x}) \tag{6}$$

subject to

$$\hat{\mathbf{y}}_{\mathrm{R_a}} - g(\hat{\mathbf{y}}_{\mathrm{MRD}}) = 0 \tag{7}$$

$$\mathbf{y}_{\mathrm{R_a}}^{(L)} \leq \hat{\mathbf{y}}_{\mathrm{R_a}} \leq \mathbf{y}_{\mathrm{R_a}}^{(U)} \tag{8}$$

$$\hat{\mathbf{y}}_{\mathrm{MRD}} \leq \mathbf{y}_{\mathrm{MRD}}^{(U)} \tag{9}$$

$$F_n^{(L)} \leq F_n \leq F_n^{(U)} \tag{10}$$

$$V_s^{(L)} \leq V_s \leq V_s^{(U)} \tag{11}$$

$$V_w^{(L)} \leq V_w \leq V_w^{(U)} \tag{12}$$

$$\widetilde{R}^{(L)} \leq \widetilde{R} \leq \widetilde{R}^{(U)} \tag{13}$$

where $\mathbf{x} = [\widetilde{R}, V_s, F_n, V_w]^T$, with $\widetilde{R}$ representing the equivalent radius of abrasive grains, V_s the belt linear speed, V_w the robotic feed speed, and F_n the normal force between the contact wheel and the workpiece.

To simultaneously enhance the workpiece surface quality and material removal ability in RBG, (5) is employed to minimize the predicted $\mathrm{R_a}$, while (6) is designed to maximize the predicted MRD. The equality constraint (7) ensures that the predicted $\mathrm{R_a}$ and MRD adhere to the mechanism model $g(\cdot)$. The inequality constraint (8) defines the allowable tolerance range for $\mathrm{R_a}$, while constraint (9) limits the MRD based on the thickness of the workpiece. Additionally, constraints (10)–(13) specify the lower and upper bounds on the machining parameters F_n, V_s, V_w, and $\widetilde{R}$, which correspond to the normal force, belt linear speed, robotic feed speed, and equivalent radius of abrasive grains, respectively.

The objective of the closed-loop parameter optimization process is to solve this MOO problem and derive a set of Pareto-optimal solutions that balance the conflicting objectives. These solutions aim to achieve an efficient and precise RBG machining process while simultaneously improving surface quality and material removal efficiency.

III. Methods

In this section, the proposed forward PINN prediction method and backward parameter optimization method are described in detail. The forward PINN prediction part begins with the presentation of the standard MMoE method, the utilized RBG mechanism model, and the proposed MMoE-based PINN method. Following this, the backward parameter optimization part introduces the standard MOPSO algorithm and the proposed powered MOPSO algorithm. Furthermore, the evaluation metrics used to assess the performance of PINN prediction and parameter optimization are elaborated. The schematic framework of the proposed closed-loop RBG machining parameter optimization method is illustrated in Fig. 2.

A. The Forward PINN Prediction Method

1) The Standard MMoE Method: The MMoE method is designed to model relationship and discrepancy across various tasks. The output of $i\mathrm{th}\,(i = 1, 2, \cdots, l)$ task in the MMoE method is formulated as follows:

$$\hat{\mathbf{y}}^i = h^i\left(\sum_{k=1}^{s} o_k^i(\mathbf{x}) e_k(\mathbf{x})\right), \tag{14}$$

where $\mathbf{x}$ represents the input data or its transformed features, s is the number of expert networks, and $e_k(\cdot)$ denotes the mapping function of $k\mathrm{th}$ expert network. $o^i(\mathbf{x}) = \mathrm{softmax}(W_o^i \mathbf{x})$ is the output of $i\mathrm{th}$ gating network, where $W_o^i \in \mathbb{R}^{s \times d}$ is the weight matrix and d represents the dimension of $\mathbf{x}$. $h^i(\cdot)$ denotes the mapping function of $i\mathrm{th}$ tower network. $\hat{\mathbf{y}}^i$ corresponds to the output of the tower network. The schematic architecture of MMoE method is illustrated in Fig. 3.

Remark 1: Compared to the conventional Shared-Bottom method [45], the MMoE method models relationships across tasks in a weighted manner by determining the extent to which outputs from different gating networks overlap. When the provided tasks are less related, the sharing of expert networks is penalized, prompting the gating networks for these tasks to prioritize selecting distinct expert networks instead [27].

2) Mechanism Model: As stated in previous work [13], [46], the relationship between $\mathrm{R_a}$ and MRD is formulated as follows:

$$\mathbf{y}_{\mathrm{R_a}} = \mathbb{E}(h_{\mathrm{ch}}) K_r \left[1 - K_w \eta^{p_1} \left(\sqrt{\frac{2}{\pi}} \mathbb{E}(h_{\mathrm{ch}})\right)^{p_2}\right], \tag{15}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

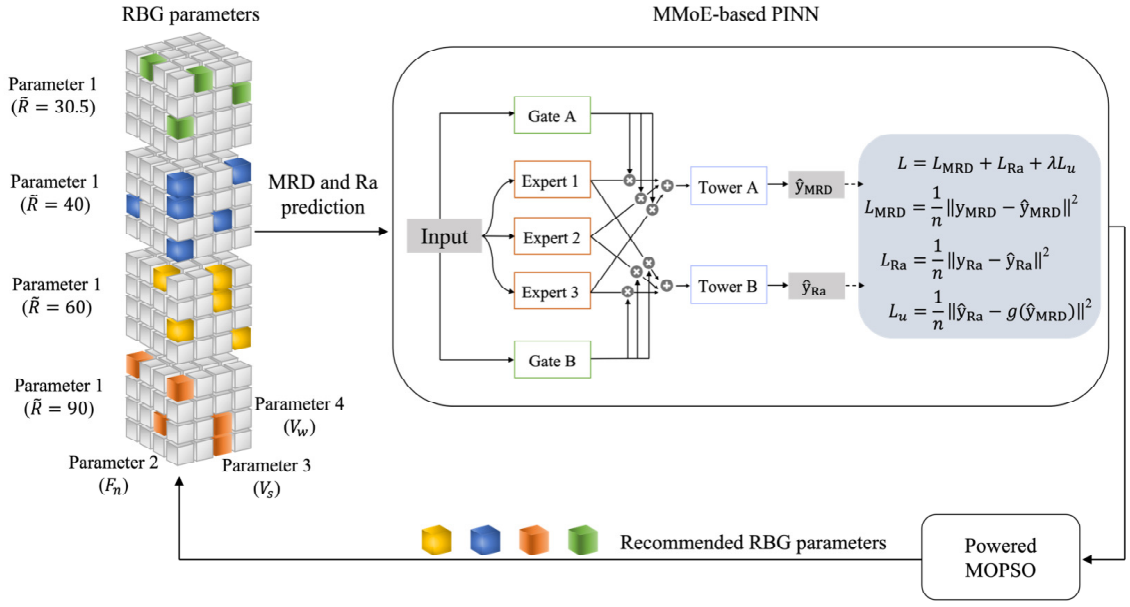


Fig. 2. The schematic framework for the closed-loop RBG machining parameter optimization method. First, four key machining parameters (i.e., normal force (F_n), belt linear speed (V_s), robotic feed speed (V_w), and equivalent radius of abrasive grains ($\bar{R}$)) are fed into the proposed MMoE-based PINN model to simultaneously predict MRD and $\mathrm{R_a}$. Then, a powered MOPSO algorithm is developed to recommend optimal RBG parameters with these predictions.

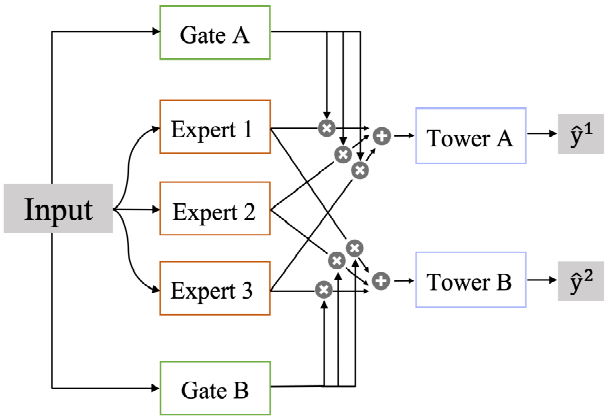


Fig. 3. The schematic architecture of MMoE method [27], where MMoE for two learning tasks is taken as a representative case.

$$\mathbb{E}\left(h_{\mathrm{ch}}\right) = \left(\frac{E_1}{E_2}\right)^{p} \left(\frac{V_w}{2R_r C_d V_s}\sqrt{\frac{\mathbf{y}_{\mathrm{MRD}}}{D_e}}\right)^{1/2}, \tag{16}$$

where $\mathbf{y}_{\mathrm{R_a}}$ and $\mathbf{y}_{\mathrm{MRD}}$ represent the observed $\mathrm{R_a}$ and MRD, respectively; $\mathbb{E}\left(h_{\mathrm{ch}}\right)$ is the expectation of maximum chip thickness h_{ch} in RBG; K_r and K_w are proportionality factors determined by the overlapping characteristics of belt abrasive grains; C_d is the effective cutting edge density and η is the percentage of effective abrasive grains; R_r represents the contact coefficient between workpiece and belt abrasive grains, characterizing the influence of contact deformation and contact surface morphology on the change in contact length; V_w and V_s denote robotic feed speed and belt linear speed, respectively; E_1 and E_2 represent the elastic moduli of contact wheel and workpiece, respectively; D_e is the equivalent diameter of the contact wheel; and p, p_1 and p_2 are exponential coefficients.

3) The MMoE-Based PINN Method: The proposed PINN method utilizes the MMoE architecture as its backbone while embedding the mechanism model (15)–(16) as a physical constraint to guide the training process. The physical loss function is defined as:

$$L_u = \frac{1}{n}\left\|\hat{\mathbf{y}}_{\mathrm{R_a}} - g(\hat{\mathbf{y}}_{\mathrm{MRD}})\right\|^2, \tag{17}$$

where $\hat{\mathbf{y}}_{\mathrm{R_a}}$ and $\hat{\mathbf{y}}_{\mathrm{MRD}}$ are the predicted values of $\mathrm{R_a}$ and MRD in RBG, respectively. $g(\cdot)$ denotes the mechanism model introduced in (15)–(16), and n is the number of training samples.

The overall loss function of the proposed PINN method is formulated as follows:

$$\begin{aligned} L &= L_{\mathrm{MRD}} + L_{\mathrm{R_a}} + \lambda L_u \\ &= \frac{1}{n}\left\|\mathbf{y}_{\mathrm{MRD}} - \hat{\mathbf{y}}_{\mathrm{MRD}}\right\|^2 + \frac{1}{n}\left\|\mathbf{y}_{\mathrm{R_a}} - \hat{\mathbf{y}}_{\mathrm{R_a}}\right\|^2 + \lambda L_u, \end{aligned} \tag{18}$$

where L_{MRD} and $L_{\mathrm{R_a}}$ represent the supervised mean squared error (MSE) loss functions for MRD and $\mathrm{R_a}$, respectively; λ is a hyperparameter that balances the two supervised MSE loss functions and the physical loss function. During the training process of the PINN method, the Adam algorithm is applied to update the model weights via minimizing L for a certain number of epochs.

B. The Backward Parameter Optimization Method

1) The Standard MOPSO Algorithm: The standard MOPSO algorithm is a heuristic-based MOO technique inspired by the social behavior of animals, such as insects, herds, birds, and fish. Over recent decades, it has been effectively applied to various MOO problems due to its simplicity of implementation and robust global search capabilities [47], [48]. This algorithm provides a set of non-dominated solutions, commonly referred

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

to as Pareto-optimal solutions, where no single objective can be improved without deteriorating at least one other objective.

As an innovative extension of the PSO algorithm [49], [50], the standard MOPSO algorithm employs similar learning mechanisms to solve MOO problems. Candidate solutions are represented by particles, each characterized by a D-dimensional velocity vector $\boldsymbol{v}$ and a D-dimensional position vector $\boldsymbol{s}$. The algorithm operates iteratively, with particle velocities and positions being updated at each iteration. The velocity of each particle is influenced by its personal best position *pbest* and the global best position *gbest* identified by the entire swarm.

The velocity and the position of ith particle in dth dimension are updated as follows:

$$\begin{aligned} v_{i,k+1}^d &= w v_{i,k}^d + c_1 r_1 (p_{i,k}^d - s_{i,k}^d) \\ &\quad + c_2 r_2 (p_{g,k}^d - s_{i,k}^d), \\ s_{i,k+1}^d &= s_{i,k}^d + v_{i,k+1}^d, \end{aligned} \tag{19}$$

where k denotes the iteration number; w is the inertia weight; c_1 and c_2 represent the cognitive acceleration coefficient and the social acceleration coefficient, respectively; r_1 and r_2 are two random numbers; $p_{i,k}^d$ is the personal best position updated by the ith particle itself; and $p_{g,k}^d$ is the global best position identified by the swarm.

It should be noted that the updates for $p_{i,k}^d$ and $p_{g,k}^d$ in the standard MOPSO algorithm differ from those in the standard PSO algorithm due to the consideration of multiple objective functions. In the standard MOPSO, $p_{i,k}^d$ (the personal best position) is selected from the non-dominated personal positions, and $p_{g,k}^d$ (the global best position) is chosen from the non-dominated global positions, identified using hypercube division of the search space and the roulette-wheel selection method [37].

Remark 2: Since the standard MOPSO algorithm addresses multiple optimization objectives simultaneously, the updates for *pbest* and *gbest* differ from those in the standard PSO algorithm. In MOPSO, *pbest* of any particle is selected from its non-dominated personal positions, which consist of its current and historical positions. Specifically, if the current position is dominated by the historical positions, the historical positions are retained; otherwise, the current position replaces the historical ones. If neither position dominates the other, one is randomly selected. *gbest* is chosen from the current Pareto-optimal solutions, which are divided into a defined number of hypercubes. To promote diversity among the Pareto-optimal solutions, hypercubes containing more than one particle are assigned a fitness value, determined by dividing a specific number by the number of particles within the hypercube. The roulette-wheel selection method [51] is then used to select a hypercube based on these fitness values. After determining a hypercube, *gbest* within that hypercube is randomly selected.

2) The Powered MOPSO Algorithm: Inspired by the powered stochastic gradient descent algorithm [40]–[42], which employs the non-linear Powerball technique to accelerate convergence and mitigate gradient vanishing, a promising strategy is to integrate the Powerball technique into the MOPSO algorithm to enhance the update of particle velocity. Consequently, the updates for particle velocity and position are modified as follows:

$$\begin{aligned} v_{i,k+1}^d &= w v_{i,k}^d + c_1 r_1 \sigma_\gamma (p_{i,k}^d - s_{i,k}^d) \\ &\quad + c_2 r_2 \sigma_\gamma (p_{g,k}^d - s_{i,k}^d), \\ s_{i,k+1}^d &= s_{i,k}^d + v_{i,k+1}^d, \end{aligned} \tag{20}$$

where $\sigma_\gamma(z) = \operatorname{sign}(z)\,|z|^\gamma$ is the non-linear Powerball function with an exponential hyperparameter $\gamma \in [0, 1]$. It should be noted that the powered MOPSO algorithm becomes equivalent to the standard MOPSO algorithm if $\gamma = 1$. This extension allows for accelerated convergence and improved performance in optimization scenarios.

3) Machining Parameter Optimization: In this paper, the powered MOPSO algorithm is deployed to optimize four machining parameters (namely, equivalent radius of abrasive grains $\widetilde{R}$, belt linear speed V_s, robotic feed speed V_w, and normal force F_n) through solving the MOO problem defined in (5)–(13). By taking the learned PINN model as the optimization objectives (see (5)–(6)), the powered MOPSO algorithm generates a set of Pareto-optimal solutions for the four machining parameters through an iterative training process. The pseudocode of the proposed closed-loop parameter optimization method is provided in Algorithm 1.

Algorithm 1: Pseudocode of the Proposed Closed-Loop Parameter Optimization Method

Input: Multi-task samples $\left\{(\mathbf{x}_j^i, \mathbf{y}_j^i)\right\}_{j=1}^{n_i}, i = 1, 2, \cdots, l$
Output: Pareto-optimal solutions $\mathbf{x}^*$
Initialize PINN weights of θ; hyperparameters of λ and γ; the number of PINN training epoch N_1; and the number of powered MOPSO iterations N_2.
for $1 : N_1$ **do**
  1) Input multi-task samples into the PINN;
  2) Predict $\hat{\mathbf{y}}^i$ for multiple tasks using (14);
  3) Calculate the physical loss using (15)-(17);
  4) Run the Adam optimization algorithm to minimize (18) and update the weights of θ.
end
for $1 : N_2$ **do**
  1) Run the Powered MOPSO algorithm to solve the MOO problem presented in (5)-(13);
  2) Update the velocity and position of each particle in the powered MOPSO algorithm using (20).
end

C. Evaluation Metrics

1) Evaluation Metrics for PINN Prediction: Three evaluation metrics, namely, the root mean square error (RMSE), the mean absolute percentage error (MAPE) and the coefficient of determination (R^2), are employed to assess the performance of the forward PINN prediction method. It should be mentioned that the RMSE is applied to calculate the deviation between the predicted and observed values, the MAPE is utilized

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

to measure the ratio of the absolute error to the observed values, and the R^2 is used to assess the correlation between the predicted and observed values. The aforementioned three evaluation metrics are formulated by

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \tag{21}$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%, \tag{22}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N}\left(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right)^2}, \tag{23}$$

where y_i and $\widehat{y_i}$ represent the observed $\mathrm{R_a}$/MRD and the predicted $\mathrm{R_a}$/MRD, respectively; and N is the number of testing samples.

2) Evaluation Metrics for Parameter Optimization: The maximum hypervolume (denoted as $\mathrm{HV_{max}}$) and the iteration number at $\mathrm{HV_{max}}$ (denoted as $T_{\mathrm{HV_{max}}}$) are employed to evaluate the performance of the powered MOPSO algorithm. The $\mathrm{HV_{max}}$ is a commonly used quality indicator for assessing the accuracy of MOO algorithms owing to its independence from the actual Pareto front, making it both practical and widely applicable [52]. The $T_{\mathrm{HV_{max}}}$ serves as a measure of the convergence speed of MOO algorithms. The two evaluation metrics for assessing parameter optimization are formulated as follows:

$$\mathrm{HV_{max}} = \max_{t=1}^{N_2}\left\{\ell\left(\bigcup_{\mathbf{a}\in\mathbf{A}_t}\{\mathbf{x} \mid \mathbf{a} \prec \mathbf{x} \prec \mathbf{r}\}\right)\right\}, \tag{24}$$

$$T_{\mathrm{HV_{max}}} = \mathrm{argmax}_{t=1}^{N_2}\left\{\ell\left(\bigcup_{\mathbf{a}\in\mathbf{A}_t}\{\mathbf{x} \mid \mathbf{a} \prec \mathbf{x} \prec \mathbf{r}\}\right)\right\}, \tag{25}$$

where $\mathbf{A}_t$ represents the Pareto-optimal solutions generated by MOO algorithms in tth iteration; N_2 is the total number of iterations of MOO algorithms; ℓ denotes the Lebesgue measure which is utilized to compute the hypervolume in d-dimensional space; $\mathbf{r}$ is a reference point; and the symbol $\prec$ indicates the dominance relationship among solutions.

IV. Experimental Platform and Dataset

A. Experimental Platform

The overview of the developed RBG experimental platform is presented in Fig. 4(a). This platform consists of an industrial robot (ABB IRB4400-60/1.96) and an abrasive belt grinder. The industrial robot is equipped with a six-dimensional force sensor (ATI Omega 160, as shown in Fig. 4(b)) mounted at its end. The experimental workpiece is a unidirectional carbon fiber-reinforced polymer (CFRP) T300 (depicted in Fig. 4(c)), which is widely used in the aerospace and automotive industries. The abrasive belt grinder comprises a contact wheel and a ceramic alumina abrasive belt ($3\mathrm{M}^{\mathrm{TM}}\mathrm{Cubitron}^{\mathrm{TM}}$ II 726 A).

After completing offline system calibration and robot path planning, an orthogonal grinding experiment with parallel grinding paths (Fig. 4(c)) is conducted using various machining parameters. The actual $\mathrm{R_a}$ values are measured using a digital surface roughness tester, while the actual MRD values are obtained through a self-developed measuring system with an accuracy of 1 μm and a sampling frequency of 4.33 kHz (Fig. 4(d)). To ensure the reliability of the measurements, the final values are determined by averaging the cross-sectional removal profile centers within the stable grinding area.

B. Experimental Dataset

In this experiment, four key controlled machining parameters (namely, equivalent radius of abrasive grains $\widetilde{R}$, belt linear speed V_s, normal force F_n and robotic feed speed V_w) are adjusted to investigate the MRD and $\mathrm{R_a}$. Specifically, four standard sizes of $\widetilde{R}$ (i.e., 30.5 μm, 40 μm, 60 μm and 90 μm) are chosen for the RBG experiment, while five different values of V_s (10 mm/s, 20 mm/s, 30 mm/s, 40 mm/s and 50 mm/s), F_n (10 N, 20 N, 30 N, 40 N and 50 N), and V_w (5.24 m/s, 9.42 m/s, 13.61 m/s, 17.8 m/s and 21.99 m/s) are set to exploit the RBG characteristics. The controlled machining parameters as well as their chosen values are presented in Table I. To significantly reduce the number of RBG experiments, an orthogonal grinding experiment with 80 sets of machining parameter combinations is carried out to investigate RBG characteristics. The corresponding MRD and $\mathrm{R_a}$ are carefully measured to evaluate the RBG performance.

TABLE I
The Controlled Machining Parameters and Their Chosen Values in the Orthogonal RBG Experiment

Machining parameter	Chosen value
Equivalent radius of abrasive grains $\widetilde{R}$	30.5, 40, 60, 90 μm
Belt linear speed V_s	5.24, 9.42, 13.61, 17.8, 21.99 m/s
Normal force F_n	10, 15, 20, 25, 30 N
Robotic feed speed V_w	10, 15, 20, 25, 30 mm/s

V. Results and Discussion

This section presents the closed-loop experimental results for forward copreddiction and backward parameter optimization. In the first stage, the results for MRD and $\mathrm{R_a}$ copreddiction using the proposed PINN method are introduced and compared with the baseline methods. In the second stage, the parameter optimization results using the developed powered MOPSO algorithm are illustrated in detail. Additionally, two recommended solutions of machining parameters are highlighted to enhance machining efficiency and precision in the RBG machining process.

A. Copreddiction of MRD and R_a

1) Experimental Setting: In the experiment, a five-fold cross-validation strategy is used to validate the effectiveness of the proposed PINN method. Specifically, 80% of the experimental dataset is used for training the PINN model, while the remaining 20% is reserved for testing the trained model in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

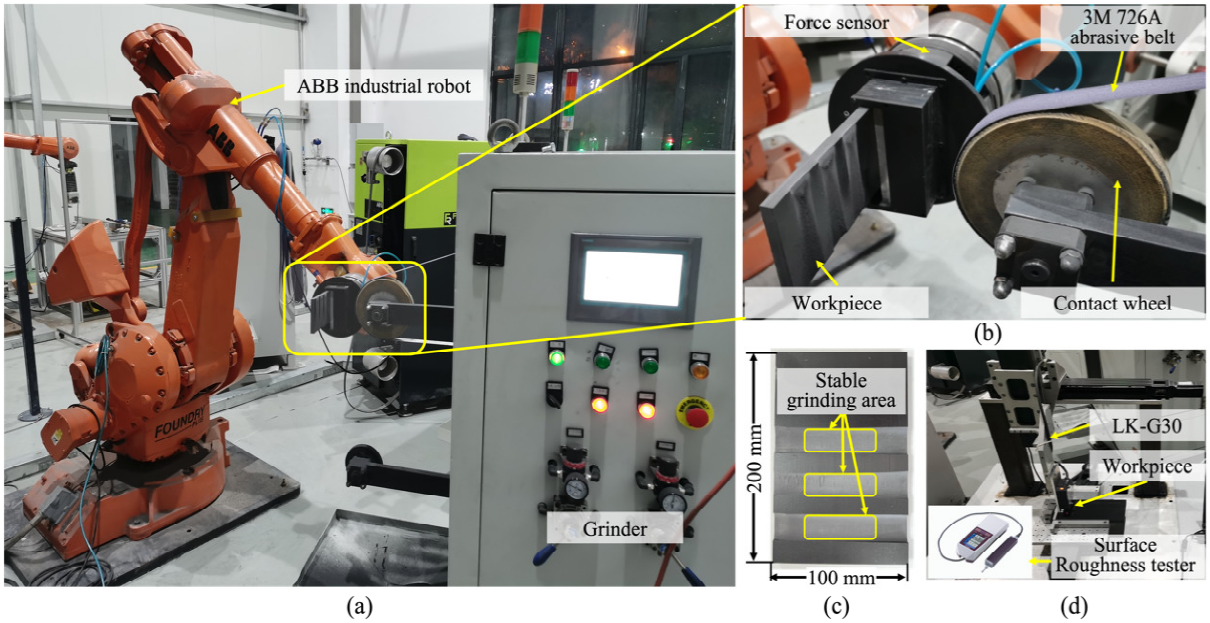


Fig. 4. (a) Overview of the experimental platform designed for an efficient and precise RBG machining process. Close-up views of (b) the RBG system and (c) the experimental workpiece. (d) The developed device for accurately measuring MRD and $\mathrm{R_a}$.

each cross-validation iteration. The input of training dataset is normalized using a min-max normalization technique, which is subsequently applied to scale the input of testing dataset. To facilitate convergence in the PINN training process, the labels of both training and testing datasets are divided by their maximum value, ensuring that the labels fall within the interval [0,1].

In the PINN architecture, the network input is the four controlled RBG machining parameters with a size of 1×4, and the labels are the measured MRD and $\mathrm{R_a}$ with a size of 1×2. To implicitly learn the shared knowledge across MRD and $\mathrm{R_a}$, two fully connected layers (denoted as FC1 and FC2), each with 8 neurons, are designed to map the machining parameters into a shared feature space. Subsequently, the mapped features are fed into an MMoE layer to balance the correlation and the discrepancy between MRD and $\mathrm{R_a}$. Motivated by the experimental setting in [27], the number of experts in the MMoE layer is set to 4 to balance learning capability and the number of parameters. The number of gates and towers in the MMoE layer is both set to 2, which is equal to the number of learning tasks. Additionally, the number of neurons in the expert, gate and tower layer is set to 4. Detailed parameter settings for the developed PINN architecture are recorded in Table II. To ensure a fair comparison, the Shared-Bottom method (which has no MMoE layer) and the standard MMoE method (which lacks the mechanism model) share the same network architecture and parameter settings as the PINN method.

Among the training parameters of the PINN method, some parameters in the mechanism model (15)-(16) are determined through system identification (i.e., p, p_1, p_2, η, C_d, K_r and K_w), and others are set based on the physical and geometric characteristics, with $E_1/E_2 = 0.01754$, $R_r = 2$ and $D_e =$ 180 mm. To balance the two supervised MSE loss functions and the physical loss function during the training process of the PINN method, λ in (18) is set to 0.6. The learning rate is set to 0.0008 with a decay of 1e-5 to ensure convergence, and the training epoch is set to 8000. The training parameters for the Shared-Bottom method and the standard MMoE method are the same as those used for the PINN method.

TABLE II
THE PARAMETER SETTINGS FOR THE DEVELOPED PINN ARCHITECTURE

Layer	Number	Neurons	Output size
Input	-	-	1×4
FC1	1	8	4×8
FC2	1	8	8×8
Gate in MMoE	2	4	2×1×4
Expert in MMoE	4	4	4×1×4
Tower in MMoE	2	4	2×1×1
Output A	-	-	1×1
Output B	-	-	1×1

2) Prediction Results: The predicted MRD and $\mathrm{R_a}$ compared to their observed values are illustrated in Fig. 5, where five different colors/markers represent five testing folds. The left figure exhibits the correlation between the predicted MRD and the observed MRD, and the right figure shows the predicted $\mathrm{R_a}$ versus observed $\mathrm{R_a}$. Both figures demonstrate a strong consistency between the predicted values and the corresponding observed values in terms of MRD and $\mathrm{R_a}$.

Herein, the evaluation metrics for MRD and $\mathrm{R_a}$ coprediction using the Shared-Bottom method, the standard MMoE method and the proposed PINN method are presented in Table III. The prediction results can be summarized in two key aspects. On the one hand, regarding MRD prediction, the Shared-Bottom method yields an average RMSE of 0.316 mm, an R^2 of 0.768

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

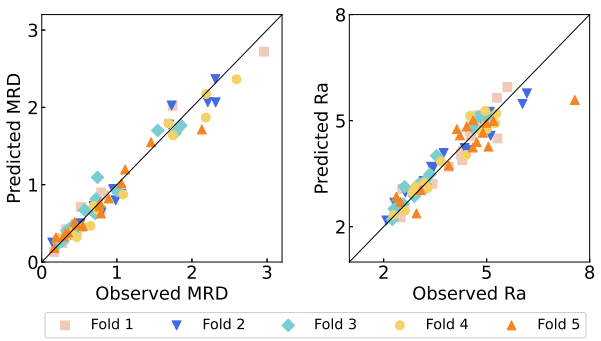


Fig. 5. The copredicted MRD and $\mathrm{R_a}$ versus the corresponding observed values using the proposed PINN method.

and a MAPE of 42.413%. Owing to the powerful multi-task learning capability, the standard MMoE method significantly reduces MRD prediction errors, achieving average RMSE, R^2 and MAPE values of 0.142 mm, 0.954 and 16.439%, respectively. Notably, the PINN method (with average RMSE, R^2 and MAPE values of 0.128 mm, 0.962 and 13.765%, respectively) outperforms the standard MMoE method, as the incorporation of the mechanism model further enhances prediction accuracy by mitigating overfitting and effectively guiding the training process.

TABLE III
THE COPREDICTION RESULTS OF MRD AND $\mathrm{R_a}$ USING DIFFERENT PREDICTION METHODS

Prediction method	RMSE		R^2		MAPE (%)	
	MRD	$\mathrm{R_a}$	MRD	$\mathrm{R_a}$	MRD	$\mathrm{R_a}$
Shared-Bottom	0.316	0.586	0.768	0.733	42.413	11.745
MMoE	0.142	0.386	0.954	0.884	16.439	**6.154**
PINN (Ours)	**0.128**	**0.382**	**0.962**	**0.887**	**13.765**	7.030

On the other hand, for $\mathrm{R_a}$ prediction, the Shared-Bottom method achieves an average RMSE of 0.586 μm, an R^2 of 0.733 and a MAPE of 11.745%. Similar to the MRD prediction results, the standard MMoE method substantially improves the performance of $\mathrm{R_a}$ prediction compared to the Shared-Bottom method, achieving average RMSE, R^2 and MAPE values of 0.386 mm, 0.884 and 6.154%, respectively. The PINN method surpasses both baseline methods in terms of RMSE and R^2, showcasing the superior effectiveness of the incorporated mechanism model in enhancing learning capability.

The impact of λ on the coprediction performance of the proposed PINN method is shown in Fig. 6. The value of λ = 0 corresponds to the standard MMoE method without incorporating the RBG mechanism, which exhibits high co-prediction errors for both MRD and $\mathrm{R_a}$. In contrast, the proposed PINN method with $\lambda \in$ (0,1] significantly improves the coprediction accuracy. Specifically, the selected value of λ = 0.6 in PINN can effectively balance the prediction of MRD and $\mathrm{R_a}$. Overall, the coprediction performance is insensitive to the values of $\lambda \in (0, 1]$, showcasing the robustness of the proposed PINN method.

The training and testing losses over epoch are depicted in Fig. 7. The PINN method shows a slightly higher training loss (solid red line) compared to the standard MMoE method (dashed red line). Nevertheless, the testing loss for the PINN method (solid orange line) is significantly lower than that for the standard MMoE method (dashed orange line) when the epoch exceeds 3000. The findings indicate that the incorporation of the mechanism model is able to effectively mitigate overfitting during the training process of the MMoE method.

B. Machining parameter optimization

1) Experimental Setting: In the experiment, the optimization objectives described in (5)-(6) are to simultaneously minimize the predicted $\mathrm{R_a}$ and maximize the predicted MRD. These predicted values are averaged across five models, each trained using a five-fold cross-validation strategy. The optimization constraints consist of one equality constraint and six inequality constraints. The equality constraint in (7) represents the mechanism model that characterizes the relationship between the predicted $\mathrm{R_a}$ and MRD. The inequality constraints in (8)–(13) are established based on practical requirements and experimental conditions. Specifically, $\mathbf{y}_{\mathrm{R_a}}^{(L)}$ and $\mathbf{y}_{\mathrm{R_a}}^{(U)}$ in (8) are set to 2.1 μm and 3.6 μm, respectively; the upper limit $\mathbf{y}_{\mathrm{MRD}}^{(U)}$ for MRD presented in (9) is set to 3 mm; $F_n^{(L)}$ and $F_n^{(U)}$ in (10) are 10 N and 30 N, respectively; the lower and upper limits for V_s in (11) are equal to 10 m/s and 25 m/s, respectively; $V_w^{(L)}$ and $V_w^{(U)}$ in (12) are set to 10 mm/s and 30 mm/s, respectively; and the belt abrasive grain size $\widetilde{R}$ in (13) is chosen from $[30, 90]$.

The hyperparameters for the powered MOPSO algorithm are empirically determined as follows: iteration number k = 200, the number of particles n = 200, inertia weight w = 0.9, cognitive acceleration coefficient c_1 = 0.1, and social acceleration coefficient c_2 = 0.11. Additionally, the exponential hyperparameter γ in (20) is chosen from $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, and the reference point for calculating the $\mathrm{HV_{max}}$ in (24) is set at (5.3, 2.1).

2) Parameters Optimization: The Pareto fronts produced by the powered MOPSO algorithm (γ = 0.8) and standard MOPSO algorithm (γ = 1) are shown in Fig. 8(a), where each point represents an optimal solution. The two Pareto fronts exhibit a significant overlap, indicating that the Powerball technique does not change the position of the optimal Pareto front. Owing to the capability of Powerball technique for alleviating gradient vanishing problem, the powered MOPSO algorithm produces a more extensive and uniform Pareto front compared to the standard MOPSO algorithm, highlighting the superiority of the proposed powered MOPSO algorithm.

The convergence curves of the powered MOPSO algorithm with different γ are displayed in Fig. 8(b), where $\gamma \in [0.1, 0.9]$ produces a faster convergence speed compared to the γ = 1 (i.e., the standard MOPSO algorithm). Table IV records the evaluation metrics for the powered MOPSO algorithm, the standard MOPSO algorithm, and three popular MOO algorithms including NSGA-II, multi-objective evolutionary algorithm based on decomposition (MOEA/D), and constrained two-archive evolutionary algorithm (C-TAEA). Our findings

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

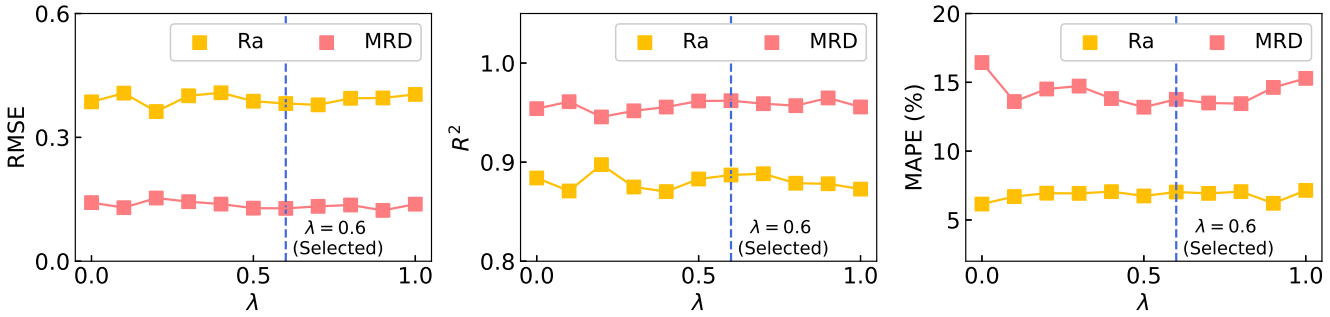


Fig. 6. The coprediction performance of the proposed PINN method with various λ values. A value of $\lambda = 0.6$ is ultimately selected for model training and subsequent machining parameter optimization.

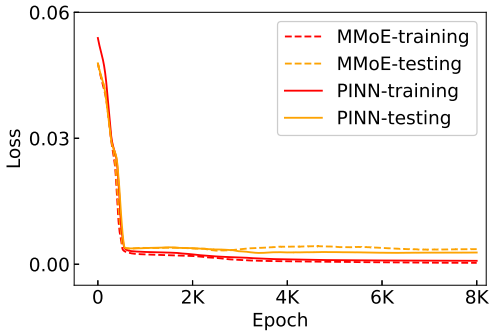


Fig. 7. Losses of the proposed PINN method and the standard MMoE method during the training and testing processes, which are averaged over five-fold cross-validation.

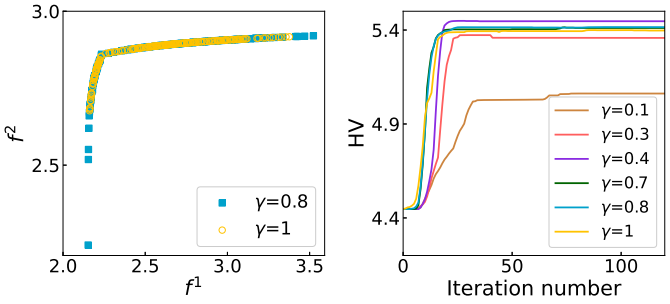


Fig. 8. (a) The Pareto fronts for machining parameter optimization in RBG using the standard MOPSO algorithm (γ=1) and the powered MOPSO algorithm (with $\gamma = 0.8$ as a representative case). (b) The convergence curves of the powered MOPSO algorithm with different values of γ.

show that $\gamma \in [0.1, 0.9]$ consistently results in faster convergence speed compared to $\gamma = 1$ ($T_{\mathrm{HV}_{\max}}$=197), and $\gamma \in \{0.4, 0.5, 0.7, 0.8\}$ achieves larger $\mathrm{HV}_{\max}$ values than the condition when $\gamma = 1$. To conclude, the experimental results illustrate that $\gamma \in \{0.4, 0.5, 0.7, 0.8\}$ is effective in optimizing the machining parameters of RBG while ensuring fast convergence, demonstrating the effectiveness of the proposed powered MOPSO algorithm compared to the standard MOPSO algorithm.

Moreover, NSGA-II, MOEA/D, and C-TAEA achieve $\mathrm{HV}_{\max}$ values of 5.400, 5.333, and 5.396, respectively. Compared to the selected MOO algorithms, the powered MOPSO algorithm exhibits superior performance in optimizing RBG machining parameters by enabling more efficient and effective exploration of the solution space. The flexibility introduced through the tunable parameter γ allows the algorithm to adapt to varying performance requirements, thereby validating both the robustness and practical applicability of the powered MOPSO algorithm.

Since the industrial belt abrasive grains typically come in a limited range of standard sizes ($\tilde{R} \in \{30.5, 40, 60, 90\}$), two optimal solutions are chosen from the Pareto-optimal solutions, which are ultimately applied to balance machining efficiency and precision in the RBG machining process. The two optimal solutions are detailed in Table V, where the predicted $\mathrm{R_a}$ and MRD corresponding to the optimal machining parameters are also presented. Among the two recommended solutions, one prioritizes higher efficiency (i.e., larger MRD), while the other emphasizes superior workpiece surface quality by minimizing $\mathrm{R_a}$. The recommended solutions provide flexibility to accommodate varying requirements in different RBG applications. The results reflect optimal trade-offs between MRD and $\mathrm{R_a}$, supporting effective decision-

TABLE IV
EVALUATION METRICS FOR THE POWERED MOPSO UNDER VARIOUS γ VALUES AND FOR OTHER MOO ALGORITHMS

Optimization method	γ	$\mathrm{HV}_{\max}$	$T_{\mathrm{HV}_{\max}}$
Standard MOPSO	$\gamma = 1.0$	5.400	197
Powered MOPSO	$\gamma = 0.1$	5.062	85
	$\gamma = 0.2$	5.195	33
	$\gamma = 0.3$	5.373	30
	$\gamma = 0.4$	5.449	28
	$\gamma = 0.5$	5.411	40
	$\gamma = 0.6$	5.362	25
	$\gamma = 0.7$	5.415	196
	$\gamma = 0.8$	5.418	184
	$\gamma = 0.9$	5.377	85
NSGA-II	—	5.400	199
MOEA/D	—	5.333	87
C-TAEA	—	5.396	198

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

making in the RBG process.

TABLE V
RECOMMENDED MACHINING PARAMETERS AND CORRESPONDING PREDICTIONS FOR THE RBG EXPERIMENT

$\widetilde{R}$ (μm)	V_s (m/s)	F_n (N)	V_w (mm/s)	$\hat{\mathbf{y}}_{\mathrm{R_a}}$ (μm)	$\hat{\mathbf{y}}_{\mathrm{MRD}}$ (mm)
30.5	25	24	10	2.17	2.70
40	25	30	10	2.73	2.90

VI. CONCLUSION

In this article, a closed-loop parameter optimization approach has been developed to identify optimal machining parameters for RBG machining process. The proposed approach has introduced a PINN method that integrates a standard MMoE model with an RBG mechanism model to accurately copredict MRD and $\mathrm{R_a}$. Building on the learned PINN model, a powered MOPSO method has been put forward to optimize the machining parameters efficiently with a small number of iterations. The proposed closed-loop parameter optimization method has been validated on a self-developed RBG system with various grinding parameters. Experimental results demonstrate two key findings: (1) the proposed PINN method significantly outperforms the standard Shared-Bottom method and the standard MMoE method in copredicting MRD and $\mathrm{R_a}$ for RBG machining process, and (2) the powered MOPSO method significantly enhances convergence speed compared to the standard MOPSO approach.

In the future, we plan to: 1) dynamically optimize the machining parameters for RBG [53], [54]; 2) incorporate transfer learning technique to enhance the accuracy of MRD and $\mathrm{R_a}$ copredict [55], [56]; 3) utilize various filtering and signal processing methods to pre-process the RBG dataset [57]–[59]; and 4) extend our closed-loop optimization method to other domains that involve process modeling with physical constraints and multi-objective parameter selection, including computer numerical control milling, battery management system, and object recognition [8], [60]–[62].

REFERENCES

[1] D. Li, J. Yang, H. Zhao, and H. Ding, Contact force plan and control of robotic grinding towards ensuring contour accuracy of curved surfaces, *International Journal of Mechanical Sciences*, vol. 227, art. no. 107449, 2022.
[2] C. Zheng, Y. An, Z. Wang, X. Qin, B. Eynard, M. Bricogne, J. Le Duigou, and Y. Zhang, Knowledge-based engineering approach for defining robotic manufacturing system architectures, *International Journal of Production Research*, vol. 61, no. 5, pp. 1436-1454, 2023.
[3] W. He, J. Mao, Y. Wang, Z. Li, H. Xie, H. Shao, and X. Zhao, Unified diagnostic and matching framework of fault and quality for robotic grinding system, *IEEE Transactions on Instrumentation and Measurement*, vol. 73, art. no. 3513612, 2024.
[4] X. Ren, X. Huang, K. Gao, L. Xu, H. Feng, X. Zhang, H. Chen, Z. Chai, and X. Chen, A review of recent advances in robotic belt grinding of superalloys, *The International Journal of Advanced Manufacturing Technology*, vol. 127, pp. 1447-1482, 2023.
[5] B. Li, J. Ni, J. Yang, and S. Y. Liang, Study on high-speed grinding mechanisms for quality and process efficiency, *The International Journal of Advanced Manufacturing Technology*, vol. 70, pp. 813-819, 2014.
[6] D. Li, J. Yang, and H. Ding, Process optimization of robotic grinding to guarantee material removal accuracy and surface quality simultaneously, *Journal of Manufacturing Science and Engineering*, vol. 146, no. 5, art. no. 051005, 2024.
[7] W. Wang, Q. Guo, Z. Yang, Y. Jiang, and J. Xu, A state-of-the-art review on robotic milling of complex parts with high efficiency and precision, *Robotics and Computer-Integrated Manufacturing*, vol. 79, art. no. 102436, 2023.
[8] P. M. Attia, A. Grover, N. Jin, K. A. Severson, T. M. Markov, Y.-H. Liao, M. H. Chen, B. Cheong, N. Perkins, Z. Yang, P. K. Herring, M. Aykol, S.J. Harris, R. D. Braatz, S. Ermon, and W. C. Chueh, Closed-loop optimization of fast-charging protocols for batteries with machine learning, *Nature*, vol. 578, no. 7795, pp. 397-402, 2020.
[9] T. Domhan, J. T. Springenberg, and F. Hutter, Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves, in *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, Buenos Aires, Argentina, Jul. 2015, pp. 3460-3468.
[10] F. W. Preston, The theory and design of plate glass polishing machines, *Journal of the Society of Glass Technology*, vol. 11, pp. 214-256, 1927.
[11] X. Ren, M. Cabaravdic, X. Zhang, and B. Kuhlenkötter, A local process model for simulation of robotic belt grinding, *International Journal of Machine Tools and Manufacture*, vol. 47, no. 6, pp. 962-970, 2007.
[12] K. Song, G. Xiao, S. Chen, X. Liu, and Y. Huang, A new force-depth model for robotic abrasive belt grinding and confirmation by grinding of the Inconel 718 alloy, *Robotics and Computer-Integrated Manufacturing*, vol. 80, art. no. 102483, 2023.
[13] S. Agarwal and P. V. Rao, Modeling and prediction of surface roughness in ceramic grinding, *International Journal of Machine Tools and Manufacture*, vol. 50, no. 12, pp. 1065-1076, 2010.
[14] W. Huai, H. Tang, Y. Shi, and X. Lin, Prediction of surface roughness ratio of polishing blade of abrasive cloth wheel and optimization of processing parameters, *The International Journal of Advanced Manufacturing Technology*, vol. 90, pp. 699-708, 2017.
[15] K. Deng, D. Gao, C. Zhao, and Y. Lu, Prediction of in-process frequency response function and chatter stability considering pose and feedrate in robotic milling, *Robotics and Computer-Integrated Manufacturing*, vol. 82, art. no. 102548, 2023.
[16] Z. Huang, J. Shao, W. Guo, W. Li, J. Zhu, Q. He, and D. Fang, Tool wear prediction based on multi-information fusion and genetic algorithm-optimized Gaussian process regression in milling, *IEEE Transactions on Instrumentation and Measurement*, vol. 72, art. no. 2516716, 2023.
[17] K. Deng, L. Yang, Y. Lu, and S. Ma, Multitype chatter detection via multichannelinternal and external signals in robotic milling, *Measurement*, vol. 229, art. no. 114417, 2024.
[18] P. Pawlus, R. Reizer, and G. M. Krolczyk, Modelling and prediction of surface textures after abrasive machining processes: A review, *Measurement*, vol. 220, art. no. 113337, 2023.
[19] C. Wei, C. He, G. Chen, Y. Sun, and C. Ren, Material removal mechanism and corresponding models in the grinding process: A critical review, *Journal of Manufacturing Processes*, vol. 103, pp. 354-392, 2023.
[20] Z. Tao, S. Li, L. Zhang, J. Qi, and D. Zhang, Surface roughness prediction in robotic belt grinding based on the undeformed chip thickness model and GRNN method, *The International Journal of Advanced Manufacturing Technology*, vol. 120, no. 9, pp. 6287-6299, 2022.
[21] P. Liu and P. Liu, Study on the abrasive belt grinding removal mechanism and surface roughness prediction of DD6 single crystal superalloy, *The International Journal of Advanced Manufacturing Technology*, vol. 139, pp. 175-195, 2025.
[22] N. Wang, G. Zhang, L. Ren, Y. Li, and Z. Yang, In-process material removal rate monitoring for abrasive belt grinding using multisensor fusion and 2D CNN algorithm, *The International Journal of Advanced Manufacturing Technology*, vol. 120, no. 1, pp. 599-613, 2022.
[23] Z. Yang, X. Xu, J. Li, D. Zhu, S. Yan, S. S. Ge, and H. Ding, Knowledge-wrapping method for prediction and evaluation of material removal behavior in robotic belt grinding, *Mechanical Systems and Signal Processing*, vol. 208, art. no. 110914, 2024.
[24] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, Physics-informed machine learning, *Nature Reviews Physics*, vol. 3, no. 6, pp. 422-440, 2021.
[25] Q. Yang, J. Zhou, and Z. Wei, Time perspective-enhanced suicidal ideation detection using multi-task learning, *International Journal of Network Dynamics and Intelligence*, vol. 3, no. 2, art. no. 100011, 2024.

[26] R. Prabhu and M. Kanthababu, Prediction of surface roughness and depth of cut in abrasive waterjet milling of alumina ceramic using machine learning algorithms, *Expert Systems with Applications*, vol. 246, art. no. 123168, 2024.

[27] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi, Modeling task relationships in multi-task learning with multi-gate mixture-of-experts, In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, London, UK, Aug. 2018, pp. 1930-1939.

[28] Y. Zhang, Y. Xin, Z.-W. Liu, M. Chi, and G. Ma, Health status assessment and remaining useful life prediction of aero-engine based on BiGRU and MMoE, *Reliability Engineering & System Safety*, vol. 220, art. no. 108263, 2022.

[29] G. Ma, Z. Wang, Z. Yang, R. Chen, W. Liu, Y. Zhang, and S. Yan, A novel pairwise domain-adaptation-assisted dual-task learning approach to coprediction of robotic machining efficiency and quality in new parameter spaces, *IEEE Transactions on Industrial Informatics*, vol. 21, no. 7, pp. 5150-5159, 2025.

[30] P. Fang, J. Yang, Q. Liao, R. Y. Zhong, and Y. Jiang, Flexible worker allocation in aircraft final assembly line using multiobjective evolutionary algorithms, *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7468-7478, 2021.

[31] W. Fang, B. Shen, A. Pan, L. Zou, and B. Song, A cooperative stochastic configuration network based on differential evolutionary sparrow search algorithm for prediction, *Systems Science & Control Engineering*, vol. 12, no. 1, art. no. 2314481, 2024.

[32] M. A. El-Shorbagy, A. Bouaouda, H. A. Nabwey, L. Abualigah, and F. A. Hashim, Bald eagle search algorithm: A comprehensive review with its variants and applications, *Systems Science & Control Engineering*, vol. 12, no. 1, art. no. 2385310, 2024.

[33] G. Ma, Z. Wang, W. Liu, J. Fang, Y. Zhang, H. Ding and Y. Yuan, Estimating the state of health for lithium-ion batteries: A particle swarm optimization-assisted deep domain adaptation approach, *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 7, pp. 1530-1543, 2023.

[34] R. Li, Z. Wang, and J. Yan, Multi-objective optimization of the process parameters of a grinding robot using LSTM-MLP-NSGAII, *Machines*, vol. 11, no. 9, art. no. 882, 2023.

[35] M. Li, W. Wang, L. Zou, C. Lv, J. Zhang, and Y. Huang, Robotic grinding of complex surfaces with an internal structured compliant tool: Multi-performance optimization in confined spaces, *Robotics and Computer-Integrated Manufacturing*, vol. 94, art. no. 102974, 2025.

[36] Q. Huang, B. Zhao, Y. Qiu, Y. Cao, Y. Fu, Q. Chen, M. Tang, M. Deng, G. Liu, and W. Ding, MOPSO process parameter optimization in ultrasonic vibration-assisted grinding of hardened steel, *The International Journal of Advanced Manufacturing Technology*, vol. 128, pp. 903-914, 2023.

[37] C. C. Coello and M. S. Lechuga, MOPSO: A proposal for multiple objective particle swarm optimization, in *Proceedings of the 2002 Congress on Evolutionary Computation*, HI, USA, May 2002, pp. 1051-1056.

[38] S. Lalwani, S. Singhal, R. Kumar, and N. Gupta, A comprehensive survey: Applications of multi-objective particle swarm optimization (MOPSO) algorithm, *Transactions on Combinatorics*, vol. 2, no. 1, pp. 39-101, 2013.

[39] X. Zheng and H. Liu, A hybrid vertical mutation and self-adaptation based MOPSO, *Computers & Mathematics with Applications*, vol. 57, no. 11-12, pp. 2030-2038, 2009.

[40] C. Cheng, Y. Wu, Q. Liu, F. Hua, Y. Zhang, and X. He, Fast high-dimensional parameter optimization for turbine blade manufacturing using the Powerball L-BFGS method under incomplete measurements, *IEEE Transactions on Instrumentation and Measurement*, vol. 73, art. no. 3524912, 2024.

[41] Y. Yuan, M. Li, J. Liu, and C. Tomlin, On the Powerball method: Variants of descent methods for accelerated optimization, *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 601-606, 2019.

[42] B. Zhou, J. Liu, W. Sun, R. Chen, C. Tomlin, and Y. Yuan, pbSGD: Powered stochastic gradient descent methods for accelerated non-convex optimization. in *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, Yokohama, Japan, Jan. 2020, pp. 3258-3266.

[43] Y. Zhang and Q. Yang, An overview of multi-task learning, *National Science Review*, vol. 5, no. 1, pp. 30-43, 2018.

[44] K. Sindhya, K. Miettinen, and K. Deb, A hybrid framework for evolutionary multi-objective optimization, *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 495-511, 2012.

[45] R. Caruana, Multitask learning, *Machine Learning*, vol. 28, pp. 41-75, 1997.

[46] W. Ding, C. Dai, T. Yu, J. Xu, and Y. Fu, Grinding performance of textured monolayer CBN wheels: Undeformed chip thickness nonuniformity modeling and ground surface topography prediction, *International Journal of Machine Tools and Manufacture*, vol. 122, pp. 66-80, 2017.

[47] L. Hu, Y. Yang, Z. Tang, Y. He, and X. Luo, FCAN-MOPSO: An improved fuzzy-based graph clustering algorithm for complex networks with multiobjective particle swarm optimization, *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 10, pp. 3470-3484, 2023.

[48] M. Reyes-Sierra and C. A. C. Coello, Multi-objective particle swarm optimizers: A survey of the state-of-the-art, *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 287-308, 2006.

[49] J. Fang, Z. Wang, W. Liu, S. Lauria, N. Zeng, C. Prieto, F. Sikström, and X. Liu, A new particle swarm optimization algorithm for outlier detection: Industrial data clustering in wire arc additive manufacturing, *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 2, pp. 1244-1257, 2022.

[50] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, and X. Liu, A novel sigmoid-function-based adaptive weighted particle swarm optimizer, *IEEE Transactions on Cybernetics*, vol. 51, no. 2, pp. 1085-1093, 2019.

[51] A. Lipowski and D. Lipowska, Roulette-wheel selection via stochastic acceptance, *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 6, pp. 2193-2196, 2012.

[52] M. Li and X. Yao, Quality evaluation of solution sets in multiobjective optimisation: A survey, *ACM Computing Surveys*, vol. 52, no. 2, art. no. 26, 2019.

[53] Y. Xue, M. Li, H. Arabnejad, D. Suleimenova, A. Jahani, B. C. Geiger, F. Boesjes, A. Anagnostou, S. J. E. Taylor, X. Liu, and D. Groen, Many-objective simulation optimization for camp location problems in humanitarian logistics, *International Journal of Network Dynamics and Intelligence*, vol. 3, no. 3, art. no. 100017, 2024.

[54] X. He, Y. Wang, and J. Jin, Bayesian inference and optimisation of stochastic dynamical networks, *International Journal of Systems Science*, vol. 55, no. 13, pp. 2589-2603, 2024.

[55] J. Fang, Z. Wang, W. Liu, L. Chen, and X. Liu, A new particle-swarm-optimization-assisted deep transfer learning framework with applications to outlier detection in additive manufacturing, *Engineering Applications of Artificial Intelligence*, vol. 131, art. no. 107700, 2024.

[56] G. Ma, S. Xu, B. Jiang, C. Cheng, X. Yang, Y. Shen, T. Yang, Y. Huang, H. Ding, and Y. Yuan, Real-time personalized health status prediction of lithium-ion batteries using deep transfer learning, *Energy & Environmental Science*, vol. 15, no. 10, pp. 4083-4094, 2022.

[57] X. Gao, F. Deng, W. Shang, X. Zhao, and S. Li, Attack-resilient asynchronous state estimation of interval type-2 fuzzy systems under stochastic protocols, *International Journal of Systems Science*, vol. 55, no. 13, pp. 2688-2700, 2024.

[58] Y. Wang, C. Wen, and X. Wu, Fault detection and isolation of floating wind turbine pitch system based on Kalman filter and multi-attention 1DCNN, *Systems Science & Control Engineering*, vol. 12, no. 1, art. no. 2362169, 2024.

[59] G. Li, Z. Wang, X. Bai, Z. Zhao, and H. Dong, Event-triggered set-membership filtering for active power distribution systems under fading channels: A zonotope-based approach, *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 1139-1151, 2024.

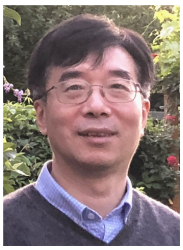
[60] Y. Yuan, G. Ma, C. Cheng, B. Zhou, H. Zhao, H.-T. Zhang, and H. Ding, A general end-to-end diagnosis framework for manufacturing systems, *National Science Review*, vol. 7, no. 2, pp. 418-429, 2020.

[61] D. Wang, C. Wen, and X. Feng, Deep variational Luenberger-type observer with dynamic objects channel-attention for stochastic video prediction, *International Journal of Systems Science*, vol. 55, no. 4, pp. 728-740, 2024.

[62] S. Hu, J. Lu, and S. Zhou, Learning regression distribution: Information diffusion from template to search for visual object tracking, *International Journal of Network Dynamics and Intelligence*, vol. 3, no. 1, art. no. 100006, 2024.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TASE.2025.3618242, IEEE Transactions on Automation Science and Engineering

Guijun Ma (Member, IEEE) received the B.Eng. degree in mechanical design, manufacturing and automation from Huazhong University of Science and Technology, Wuhan, China, in 2017, and the Ph.D. degree in mechanical engineering with the State Key Lab of Intelligent Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China, in 2023. From 2021 to 2022, he was a Visiting Scholar with the Department of Computer Science, Brunel University London, Uxbridge, U.K. He is currently with the State Key Laboratory of Intelligent Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China. His research interests focus on robotic dexterous manipulation and AI for science.



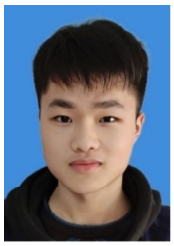
Zidong Wang (Fellow, IEEE) received the B.Sc. degree in mathematics in 1986 from Suzhou University, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology.

He is currently Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the U.K. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published a number of papers in international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for *International Journal of Systems Science*, the Editor-in-Chief for *Neurocomputing*, the Editor-in-Chief for *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals, including IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS–PART C. He is a Member of the Academia Europaea, a Member of the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, a Fellow of the IEEE, a Fellow of the Royal Statistical Society, and a member of program committee for many international conferences.

Weibo Liu (Member, IEEE) received the B.Eng. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, U.K, in 2015, and the Ph.D. degree in artificial intelligence in 2020 from the Department of Computer Science, Brunel University London, Uxbridge, U.K. He is currently a Lecturer in the Department of Computer Science, Brunel University London, Uxbridge, U.K. His research interests include intelligent data analysis, evolutionary computation, transfer learning, machine learning and deep learning. He serves as an Associate Editor for the JOURNAL OF AMBIENT INTELLIGENCE AND HUMANIZED COMPUTING and the COGNITIVE COMPUTATION. He is a very active reviewer for many international journals and conferences.

Zeyuan Yang received the Ph.D. degree in mechanical engineering with the State Key Lab of Intelligent Manufacturing Equipment and Technology, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2024. From 2021 to 2022, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He is currently a Postdoctoral Researcher with the State Key Lab of Intelligent Manufacturing Equipment and Technology, HUST, Wuhan, China. His research interests include robotic machining, dexterous manipulation and knowledge-based control.



Desheng Huang received the B.Eng. degree in School of Future Technology, Huazhong University of Science and Technology, Wuhan, China. He is currently pursuing the Ph.D. degree in information and communication engineering with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. His research interests focus on robotic dexterous manipulation and robotic simulation.

Han Ding (Senior Member, IEEE) received the Ph.D. degree in mechanical engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 1989.

Supported by the Alexander von Humboldt Foundation, he was with the University of Stuttgart, Stuttgart, Germany, from 1993 to 1994. He was with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore from 1994 to 1996. He has been a Professor at HUST since 1997. He has been a "Cheung Kong" Chair Professor of Shanghai Jiao Tong University since 2001 and was elected as a Member of the Chinese Academy of Sciences in 2013. His research interests include robotics, multiaxis machining, and equipment automation.

Dr. Ding was an Editor of IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING and a Senior Editor of IEEE ROBOTICS AND AUTOMATION LETTERS.