Check for updates

Special issue: Advances in machining and milling technologies: Innovations, applications, management and process optimization

Institution of MECHANICAL ENGINEERS
Journal of PROCESS MECHANICAL ENGINEERING

A robust mechanistic–metaheuristic hybrid approach for predicting milling force coefficients

Proc IMechE Part E:
J Process Mechanical Engineering
1–15
© IMechE 2025
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/09544089251390127
journals.sagepub.com/home/pie
Sage

B. Sercan Bayram[1,2], Ümit Atila[3], Gültekin Uzun[2], Zafer Tekiner[2], İhsan Korkut[2] and Berzah Yavuzyegit[4,5]

Abstract

This study proposes an optimization-based methodology for predicting cutting forces in milling by eliminating the need for traditional offline calibration procedures. A mechanistic force model is employed, in which cutting force coefficients are identified using population-based metaheuristic algorithms, namely Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO). Cutting force data collected during machining are utilized to optimize the model parameters directly. The performance of each algorithm is systematically evaluated through 30 independent trials to ensure statistical reliability. The DE algorithm demonstrated the best performance, converging in all 30 runs with an average of 197 iterations and 5.4 s, followed by PSO (363 iterations, 9.8 s), while GA exhibited lower reliability (18 successful runs, 2108 iterations, 62.9 s). The optimized coefficients were validated against experimental data, yielding mean prediction errors of 2.82 N (F_x) and 4.35 N (F_y). The proposed method offers a fast, accurate, and scalable solution for cutting force prediction, supporting adaptive process control, and contributing to the development of intelligent manufacturing systems.

Keywords

Cutting force coefficients, optimization, metaheuristic algorithms, mechanistic method, milling

Received: 1 July 2025; accepted: 5 October 2025

Introduction

Machining is one of the fundamental manufacturing methods due to its capability to produce complex geometries with high precision across a wide range of materials.[1] Among these methods, milling is particularly critical in the aerospace, automotive, and medical sectors.[2] Advances in computer numerical control (CNC) machines and cutting tool technologies have enabled precise and repeatable manufacturing; however, the effective utilization of these technologies under appropriate operating conditions plays a crucial role in enhancing energy efficiency, product quality, and environmental sustainability.[3–7] Therefore, researchers have employed advanced techniques such as regression analysis, multi-objective optimization methods, artificial neural networks (ANN), and intelligent optimization algorithms to optimize cutting conditions.[8–12] For instance, Palanisamy et al.[13] performed a genetic algorithm (GA)–based multi-objective optimization in milling operations by considering multiple performance criteria such as cutting force, surface roughness, tool life, and vibration amplitude. The optimal cutting parameters obtained through GA — cutting speed of 38.71 m/min, feed rate of 0.30 mm/tooth, and depth of cut of 0.92 mm — were experimentally validated. The model predicted a surface roughness of 0.71 μm and cutting forces of Fx = 283.62 N and Fy = 457.96 N. Experimental results showed a surface roughness of 2.6 μm and cutting forces of Fx = 290 N and Fy = 465 N. Although some deviations were observed in surface roughness, the model accurately predicted cutting forces. In a similar study, Wu et al.[14] developed a multi-objective optimization approach by integrating deep learning, GA, and TOPSIS methods. This approach modeled the nonlinear relationships among energy consumption, surface quality, and material removal rate and identified optimal cutting parameters through Pareto optimization. The method was validated by milling experiments and improved surface quality while reducing energy consumption by 15%. In another study, Radhakrishnan and Nandan[15] employed a combination of artificial neural networks (ANN) and multiple regression analysis to model the relationship between cutting force and process parameters such as cutting speed, feed rate, and depth of cut. Initially, outliers were filtered through regression analysis, and the resulting refined dataset was used to train the ANN model. The developed empirical model

[1]Manufacturing Engineering, Graduate School of Natural and Applied Sciences, Gazi University, Ankara, Türkiye
[2]Manufacturing Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye
[3]Computer Engineering, Faculty of Engineering, Gazi University, Ankara, Türkiye
[4]Mechanical Engineering, Faculty of Engineering and Architecture, Recep Tayyip Erdoğan University, Rize, Türkiye
[5]School of Medicine, Pharmacy and Biomedical Sciences, Faculty of Science and Health, University of Portsmouth, Portsmouth, UK

Corresponding author:
Berzah Yavuzyegit, School of Medicine, Pharmacy and Biomedical Sciences, Faculty of Science and Health, University of Portsmouth, Portsmouth, PO1 2DT, UK.
Email: berzah.yavuzyegit@port.ac.uk

could predict cutting forces with an R^2 accuracy of 95.3%, enabling effective optimization of machining parameters. However, although methods such as regression and ANN[16,17] have produced successful results for optimizing cutting conditions, these approaches often require large amounts of experimental data and offline calibration experiments. When coupled with the necessity to generate separate datasets for each specific tool–material combination, this leads to significant limitations in terms of time and cost for industrial applications. The preparation of experimental data imposes considerable economic burdens depending on the tools and workpiece materials used, thereby restricting the broad practical applicability of these methods. As an alternative to this problem, Wang et al.[18] developed a transfer learning-based approach by combining simulation data obtained through Third Wave AdvantEdge software with a limited amount of experimental data. The model, developed using an ANN architecture with four hidden layers, predicted cutting forces with an error rate of 11.15%, even under low-data conditions. However, the authors emphasized that increasing the amount of experimental data adversely affected model performance. This finding indicates that simulation-based approaches cannot eliminate the need for experimental data and remain sensitive to data imbalance issues. Therefore, there remains a need to develop models with high generalization capability, reduced data requirements, and adaptability to various tool–material combinations for practical industrial applications.[19–21] Li et al.[22] developed a force prediction model by processing spindle current signals using deep learning algorithms, aiming to overcome the limitations of using expensive measurement systems and provide a feasible solution. Their study achieved successful predictions with error rates below 4% for forces in the feed direction and below 10% for forces perpendicular to the feed direction. However, the large number of experiments required to train the deep learning algorithm posed a significant limitation for practical applications of the method.

In this context, the exploration of hybrid approaches has become inevitable in order to overcome the data-dependent limitations of conventional methods and to develop reliable cutting force prediction models with high applicability in industrial settings. Such hybrid methods, which integrate analytical models with artificial intelligence and heuristic optimization techniques, not only address significant gaps in the literature but also contribute directly to the development of efficient solutions that operate with limited data, possess strong generalization capabilities, and can be integrated into online control systems.[23] To minimize the number of experiments or to achieve rapid and reliable solutions, it is essential to have a thorough understanding of cutting mechanics and to ensure that the developed models are grounded on solid physical and mechanical principles. Research on reliably modeling cutting forces has historically been built upon a well-established foundation.[24] One of the pioneering studies in this field was conducted by Martellotti in 1941,[25] followed by a more systematic modeling approach based on physical and mechanical principles introduced by Merchant in 1945.[26] Merchant developed a fundamental force model, expressed by equation (1), by relating the cutting forces generated during the material removal process to the process parameters such as chip thickness (h) and chip width (b):

$$F = K_t.h.b \tag{1}$$

Here, K_t represents the specific cutting force coefficient (SFC). This model provided a linear and simplified description of the material removal process. In 1951, Lee and Shaffer[27] extended the model by incorporating plastic deformation theory to account for the physical deformation effects in the shear zone. Subsequently, Oxley (1989)[28] further expanded the scope of the linear model by introducing a mechanistic approach that considered thermal effects and the properties of the workpiece material. In 1991, Altintaş and Budak[29–31] extended the linear cutting force model to milling operations using a mechanistic approach. In their study, cutting forces were defined in terms of tangential (Ft), radial (Fr), and axial (Fa) components and were correlated with the process parameters. The force components in milling operations were modeled as presented in equation (2):

$$\left.\begin{aligned} F_t &= K_t.h.b + K_{te}.b \\ F_r &= K_r.h.b + K_{re}.b \\ F_a &= K_a.h.b + K_{ae}.b \end{aligned}\right\} \tag{2}$$

Here, K_t, K_r, and K_a represent the tangential, radial, and axial cutting force coefficients, respectively, while K_{te}, K_{re}, and K_{ae} denote the edge cutting force coefficients. This model accurately related cutting forces to chip geometry and process parameters, enabling the prediction of stability lobes during milling operations. The work of Altintaş and Budak laid the foundation for modern linear cutting force models and marked a significant milestone in the analytical investigation of machining processes.

In the mechanistic approach, which is widely used for modeling cutting forces, the accuracy of force predictions depends on the precise determination of specific cutting force coefficients (SFCs). Budak[32] emphasizes that edge-cutting force coefficients are critical in accurately estimating the total cutting force, particularly at low chip thicknesses. However, in the mechanistic approach, cutting force coefficients are determined experimentally in offline conditions, which requires conducting separate experiments for each cutting tool geometry and process conditions.[33] The methodology for calculating SFCs within the mechanistic modeling framework is described in detail in the work of Altintaş.[21] A significant limitation of this approach lies in the necessity of performing a series of calibration experiments to determine SFCs for every new tool–material pair, which must be conducted offline.[33–35] Farhadmanesh and Ahmadi[36] proposed three different methods based on recursive least squares (RLS), Kalman filter (KF), and extended Kalman filter (EKF) to overcome the challenges in calculating SFCs and to enable their online estimation. Their study, where these methods were compared numerically and experimentally, emphasized that SFCs serve as a more sensitive indicator than force signals, particularly for monitoring tool wear. Similarly, Rosnan et al.[37] presented an FFT-based method for estimating SFCs, requiring fewer cutting tests than existing methods. Although the model demonstrated high accuracy and good fit in their study, the need for experimental data remains a limitation.

In this context, most studies in the literature have focused either on online estimation methods or on model calibration through uncertainty analysis; however, they have not proposed a systematic optimization approach capable of determining

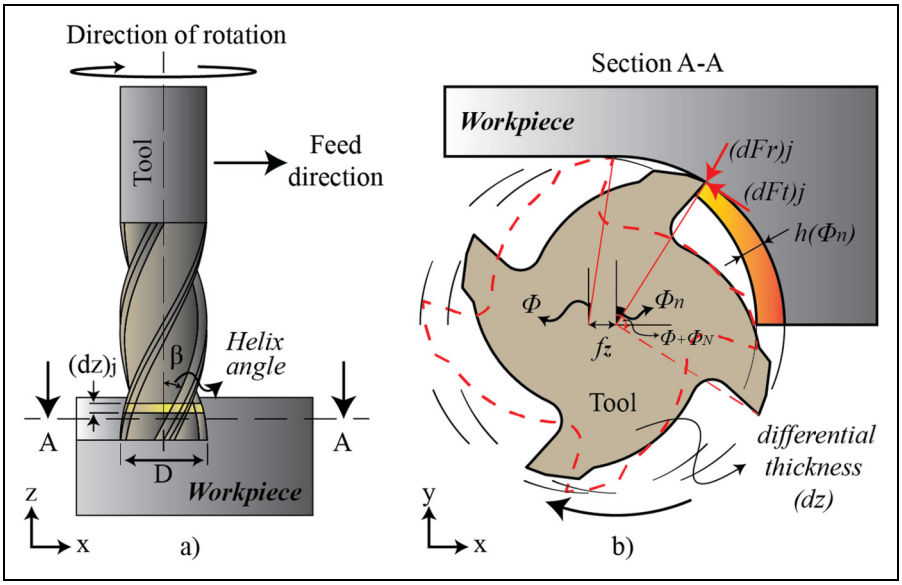


Figure 1. Schematized nomenclature for the analytical model. (a) Illustration of the cutting tool and workpiece interaction. (b) Cross-sectional view representing the engagement between the cutting tool and workpiece at a differential disk element corresponding to the axial depth of cut.

globally optimal values for SFCs in the highly multivariate and complex nature of milling operations. Moreover, the need for extensive offline experimental data collection to determine SFCs persists in existing studies, posing significant limitations regarding time and resource consumption. In the present study, while preserving the mathematical form of the conventional model, the parameter identification process is addressed as a multi-objective and multi-parameter optimization problem. To solve this problem, swarm intelligence-based algorithms such as Genetic Algorithm (GA),[38] Particle Swarm Optimization (PSO),[39] and Differential Evolution (DE)[40] are employed. The primary aim of this study is to demonstrate the applicability of optimization algorithms for solving the proposed problem. Accordingly, the development of new or hybrid algorithms specifically tailored for this problem is beyond the scope of this work. Instead, well-established reference algorithms such as GA, PSO, and DE are utilized, as they have proven to be effective in exploring complex, high-dimensional, and nonlinear search spaces while maintaining a balanced trade-off between exploration and exploitation. This selection supports the optimization-based solution strategy adopted in the study. The proposed approach provides cutting force prediction and achieves global model improvement, thereby enhancing model accuracy. Furthermore, this approach eliminates the offline experimental determination of SFCs, allowing these coefficients to be calculated directly during machining. In this respect, the study provides a reference for future research focused on cutting force prediction, enabling more precise and autonomous tool condition monitoring and process control. Unlike many prior works that employ these algorithms in isolation or rely on offline calibration procedures, this study integrates GA, PSO, and DE into an in-process optimization framework that dynamically updates cutting force coefficients based on real-time machining data. This unique implementation significantly enhances the method's efficiency and eliminates the need for time-consuming offline calibration experiments. Furthermore, the proposed methodology is scalable for a range of practical scenarios, including different tool–material combinations, cutting tool geometries, and machining parameter settings. While certain limitations remain in terms of real-time deployment, the framework holds strong potential for various practical applications during machining, including tool wear monitoring, prediction of surface roughness and integrity, and identification of stable machining conditions.

Materials and methods

Mechanistic force modeling in helical milling

This study utilized a mechanistic modeling approach to predict the cutting forces generated during helical end milling operations. The mechanistic model assumes a linear relationship between the cutting forces and the chip volume. Initially, the cutting tool is segmented into differential disk elements along its axial direction. The forces acting on each disk element are mathematically expressed using the following equations[21]:

$$\left.\begin{aligned} dF_t &= K_{tc}hdz + K_{te}dz \\ dF_r &= K_{rc}hdz + K_{re}dz \\ dF_a &= K_{ac}hdz + K_{ae}dz \end{aligned}\right\} \quad (3)$$

In these equations, dF_t, dF_r and dF_a represent the tangential, radial, and axial differential cutting forces, respectively. The coefficients K_{tc}, K_{rc}, and K_{ac} and K_{te}, K_{re}, and K_{ae} denote the specific cutting and edge coefficients, while h and dz indicate the instantaneous chip thickness and axial element thickness, respectively. This incremental computational method allows for detailed modeling of force distribution within the cutting zone, which is crucial for accurately characterizing instantaneous force variations across different axial positions and tool engagement angles.

Geometric considerations and lag angle formulation

A schematic representation of the process is provided in Figure 1 to enhance understanding of the analytical procedure

for calculating cutting forces described in the steps above. The geometry of the cutting tool significantly influences the distribution and magnitude of cutting forces. In this study, an end mill with diameter D, helix angle β, and a zero tip radius was considered. A key geometric parameter in helical milling is the lag angle (ψ), which arises due to the helical inclination of the cutting edges. When the helix angle β is greater than zero, points along the cutting-edge exhibit axial and circumferential positional differences, resulting in varying chip thicknesses (dz).

A coordinate system, used to represent the directions of the cutting forces, is established based on the center of the tool's flat-end face. In this system, the x and y axes lie in the surface plane, while the z axis is oriented upward along the tool axis.

The angular positioning of the cutting edges is governed by the step angle ϕ_Z, determined as follows:

$$\phi_Z = \frac{2\pi}{Z} \tag{4}$$

where Z represents the number of cutting edges. The positional angle of the ith cutting edge at the jth disk element and kth axial depth, $\phi(i,\ j,\ k)$ is expressed as follows:

$$\phi(i,\ j,\ k) = \phi + \sum_{n=0}^{i-1} \phi_Z - \psi(j,\ k) \tag{5}$$

The lag angle $\psi(j,\ k)$ is calculated as follows:

$$\psi(j,\ k) = \frac{2tan\beta}{D} \sum_{n=0}^{j} \Delta z(k) \tag{6}$$

Given these formulations, the instantaneous chip thickness $h_{(i,j)}(\phi,\ k)$ along the cutting edge varies periodically due to changes in rotational angle and feed rate, and is computed as follows:

$$h_{(i,j)}(\phi,\ k) = f_z sin\phi_{i,j}(k) \tag{7}$$

Cutting force component calculation

Differential forces acting on the infinitesimal cutting edges of thickness *dz* are first computed individually and then summed to find the total cutting forces. Tangential (dF_t) and radial (dF_r) force components are transformed into Cartesian coordinates using trigonometric relationships according to the defined coordinate system as follows:

$$\left.\begin{aligned} dF_{x(i,j)}(\phi,\ k) &= -dF_{t(i,j)}(\phi,\ k)cos\phi_j(k) - dF_{r(i,j)}(\phi,\ k)sin\phi_j(k) \\ dF_{y(i,j)}(\phi,\ k) &= dF_{t(i,j)}(\phi,\ k)sin\phi_j(k) - dF_{r(i,j)}(\phi,\ k)cos\phi_j(k) \end{aligned}\right\} \tag{8}$$

Integrating these differential force components along the axial cutting depth, the instantaneous total cutting forces acting on the milling tool can be obtained as follows:

$$F_{xi,j}(\phi,\ k) = \left[\frac{f_z}{4z_\beta}(-K_{tc}cos2\phi_j(k) + K_{rc}[2\phi_j(k) - sin2\phi_j(k)]) + \frac{1}{z_\beta}(K_{te}sin\phi_j(k) - K_{re}cos\phi_j(k))\right]_{k_{j,1}(\phi_j(k))}^{k_{j,2}(\phi_j(k))} \tag{9}$$

$$F_{yi,j}(\phi,\ k) = \left[\frac{-f_z}{4z_\beta}(K_{tc}[2\phi_j(k) - sin2\phi_j(k)] + K_{rc}cos2\phi_j(k)) + \frac{1}{z_\beta}(K_{te}cos\phi_j(k) + K_{re}sin\phi_j(k))\right]_{k_{j,1}(\phi_j(k))}^{k_{j,2}(\phi_j(k))} \tag{10}$$

In Equations (8) and (9), the cutting forces are calculated in a discrete manner based on the instantaneous angular position of each cutting edge. The cutting tool is divided into disk elements with differential thickness along the axial depth of cut. For each of these disk elements (j^{th}), the effects of the i^{th} cutting edges that are engaged in the cutting zone at angular position ϕ_j are individually computed using a numerical approach. In this way, the total cutting force is obtained by summing the contributions of all cutting edges engaged in the cutting zone along the axial depth of cut at the given angular position ϕ_j.

Where $z_\beta = \frac{2tan\beta}{D}$, $\phi_j = \phi + \phi_Z$ represents the angular positioning of each cutting edge. $k_{j,1}(\phi_j(k))$ and $k_{j,2}(\phi_j(k)$ are the lower and upper intersection boundaries of the cutting portion of the i^{th} cutting edge on the j^{th} disk element. This systematic formulation enables precise calculation and modeling of instantaneous cutting forces during milling operations.

Optimization algorithms and model formulation

Various mechanistic models have been developed based on fundamental physical principles to predict cutting forces in milling operations. The primary distinction among these models lies in the methodology employed for identifying the cutting force coefficients, which are critical for accurately characterizing the material removal process. Conventional approaches typically rely on empirical formulations derived from extensive experimental trials and linear regression-based parameter estimation. While these traditional methods can provide reliable results, they are associated with significant experimental costs, require time-intensive data collection, and are constrained by the inherent assumptions of linearity.

To address these challenges, this study employs population-based metaheuristic optimization algorithms, including GA, PSO, and DE. These algorithms optimize the cutting coefficients by minimizing the deviation between analytically predicted forces and in-process force

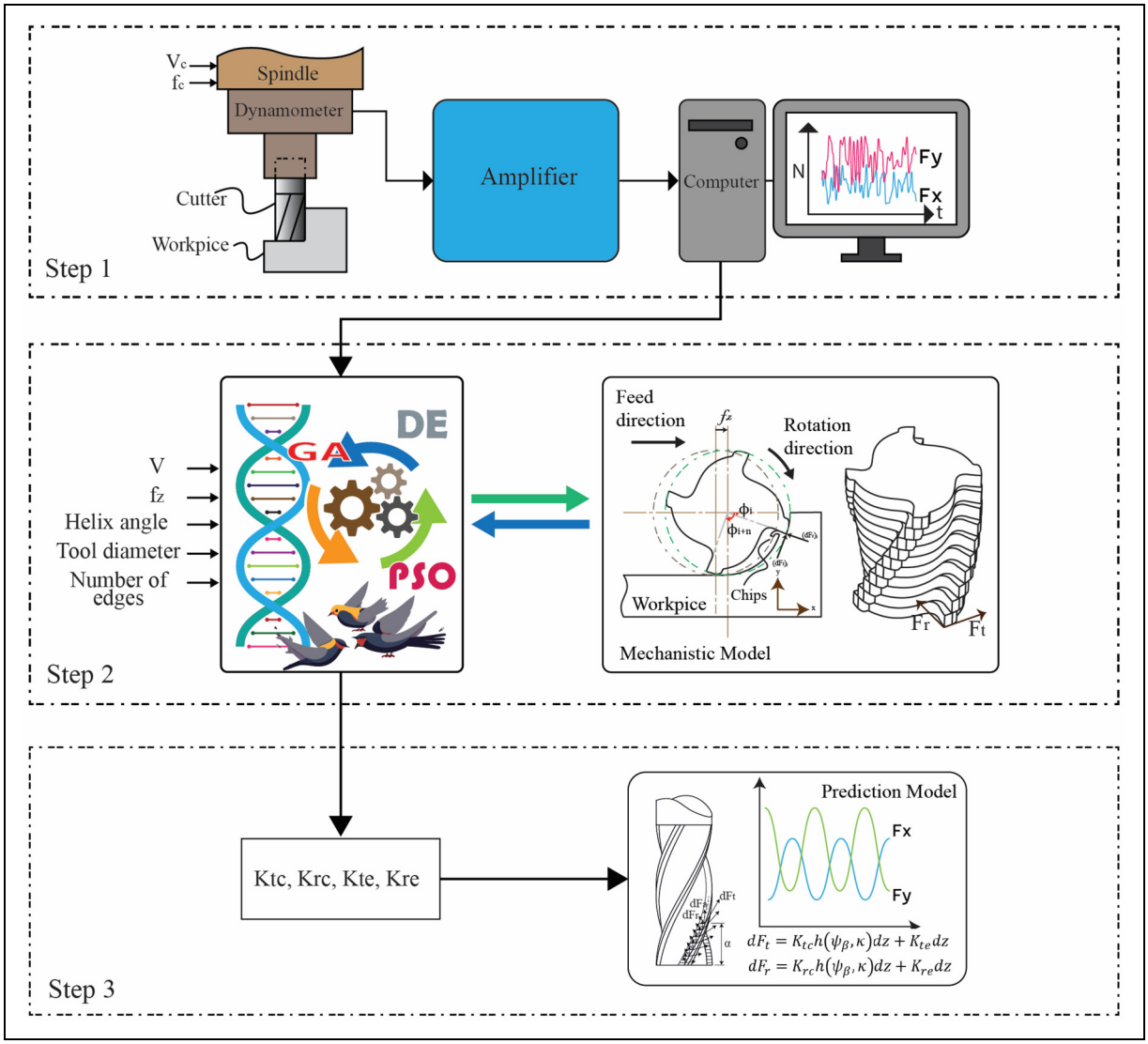


Figure 2. Flow diagram of the method.

measurements. This approach eliminates the need for multiple offline experiments typically required by conventional calibration methods that rely on average cutting force data, offering a more practical computational framework. In addition, it allows for tracking the variations of cutting coefficients throughout the machining process.

The operational steps of the proposed methodology are illustrated schematically in Figure 2.

As shown in Figure 2, the optimization methodology involves three main steps:

- **Data acquisition:** Collection of experimental measurements for cutting force components F_x and F_y.
- **Model optimization:** Utilization of these experimental measurements to optimize an objective function derived from the analytical cutting force model.
- **Prediction Model Formulation:** Construction of a predictive mathematical model using optimized cutting coefficients.

Metaheuristic optimization formulation. In applying metaheuristic algorithms, the optimization problem is rigorously defined as the minimization of the squared error between the measured and predicted cutting force components. This objective function can be expressed as follows:

$$F_{(x,y)_\theta} = f_{(x,y)_\theta}(K_{tc},\ K_{rc},\ K_{te},\ K_{re}) \tag{11}$$

where the function $f_{(x,y)_\theta}$ describes the analytical relationship between the process parameters and the cutting force coefficients. The operational logic of the optimization framework is detailed in the pseudo-code in Table 1, which demonstrates the calculation of predicted forces for a given set of coefficients and the computation of the objective error function. During the optimization, the cutting coefficients are iteratively updated as candidate solutions within the defined search space, guided by the respective metaheuristic algorithm. The coefficients yielding the minimum error represent the optimal solution, thus allowing precise force estimation without the need for extensive offline calibration.

To clarify, the squared error minimized in the objective function represents the local discrepancy between the measured and predicted pointwise cutting forces over angular segments, and it is expressed in a non-dimensional form due to normalization and alignment procedures. As presented in Table 3, these values reflect the fitness criteria utilized by

Table 1. PSEUDO code representation of the object-oriented cutting force simulation model for optimization algorithms.

Optimization Problem

Initialize:
  Parameters: bounds, cutting field, a_p, dz, β, D, f_z, Vc, Z, K_{tc}, K_{te}, K_{rc}, K_{re}
  Computed Values: $\phi_Z \leftarrow \frac{2\pi}{Z}$
Function cutting force (angle, solutions):
  Fx, Fy ← 0 (Initialize forces)
  Start loop for each tooth: LOOP i ← 0 TO Z-1
    Calculate the angular position of the tooth: Φ ← angle + ϕ_Z x i
    K_{tc}, K_{te}, K_{rc}, K_{re} ← solutions
    Star loop for each axial-depth: LOOP k ← 0 **TO** a_p: (from 0 to `$a_p$` with step size `dz`)
      Calculate the lag angle: $\psi \leftarrow (\Phi - ((2 \times \tan(\beta)) \div D) \times k) \text{ MOD } (2 \times \pi)$
      If the lag angle is within the cutting region: IF ψ in cutting field
        Calculate the chip thickness: $h \leftarrow f_z \times \sin(\psi)$
        Integrate tangential and radial forces:
        $F_t \leftarrow dz \times (K_{tc} \times h + K_{te})$
        $F_r \leftarrow dz \times (K_{rc} \times h + Kre)$
        Calculate Cartesian force components:
        $F_x \leftarrow F_x + (-F_t \times \cos(\psi)) - F_r \times \sin(\psi)$
        $F_y \leftarrow F_y + F_t \times \sin(\psi) - F_r \times \cos(\psi)$
  Return the total X and Y forces: **RETURN** F_x, F_y
Function objective function (angle, measured, solutions):
  forces ← cutting force(angle, solutions)
  error ← error(forces, measured)
  RETURN error

the optimization algorithms during the convergence process. However, in order to evaluate the real-world predictive performance of the model, a separate error analysis is performed on the full dataset using the cutting coefficients obtained from the optimization. In this analysis, the mean absolute difference between the measured and predicted forces is computed in Newtons (N) to assess the model's accuracy.

Implementation details of the proposed approach. Traditional methods for determining cutting force coefficients involve a series of machining trials at various feed rates, followed by the application of linear regression to the resulting data.[41] The slope and intercept of the resulting linear equation provide estimates for the cutting force and edge force coefficients, respectively.[21] Although robust, these methods are time-consuming, resource-intensive, and may not be practical for real-time or adaptive manufacturing environments.

In contrast, the methodology introduced in this study enables direct, in-process optimization of cutting coefficients using force data acquired during actual machining. After collecting the force measurements, the data is segmented into fixed-length windows (typically corresponding to a complete revolution of the cutting tool) to ensure temporal resolution and computational efficiency (Figure 3). Each data window is then fed to the selected metaheuristic optimization algorithm (GA, DE, or PSO), which generates candidate solutions equal to the defined population size (40 in this study).

The unknown entry angle of the force signals within each window introduces a phase shift between the measured and theoretical forces. This phase shift is calculated by multiplying the angular shift, obtained through cross-correlation, by the sampling step size. Angular alignment is achieved using the computed phase shift, and the objective function converges, based on the squared error at the closest force point, to optimize the cutting coefficients.

$$SE = (F_{measured} - F_{calculated})^2 \tag{12}$$

The optimization process continues iteratively, with candidate solutions evolving through reproduction, mutation, or positional updates, depending on the algorithm. The search proceeds until a solution yielding an error below the predefined threshold is found, resulting in the optimal set of cutting force coefficients.

This continuous and adaptive identification process enables the cutting force model to be updated dynamically throughout the machining operation, eliminating the need for separate offline calibration experiments. As a result, the proposed approach offers a practical, efficient, and scalable solution for real-world machining applications, supporting the development of intelligent and autonomous manufacturing systems.

Experimental setup and parameters

Milling experiments were conducted to validate and test the proposed cutting force prediction models and optimization algorithms. The cutting tool selected for the experiments was a 4-flute Tungsten Carbide (WC) end mill with an 8 mm diameter, a 35-degree helix angle, and an AlCrN physical vapor deposition (PVD) coating. The workpiece material used was Ti6Al4 V alloy, a widely used aerospace-grade titanium alloy noted for its machinability challenges.

Milling experiments were conducted on a Haas VF 2 SS five-axis computer numerical control (CNC) machine. Cutting forces were recorded using a Kistler Type 9123C rotary dynamometer. The milling and measurement setup is illustrated in Figure 4.

In the milling experiments, the cutting parameters were set as follows: cutting speed (V) of 100 m/min, feed per tooth (f_z) of 0.1 mm/tooth, radial depth of cut (a_e) at 50% of the tool diameter, and axial depth of cut (a_p) of 4 mm. Cutting force measurements were conducted using a dynamometer operating at a 10 kHz sampling rate. The experimental conditions were deliberately held constant to focus explicitly on evaluating and optimizing the cutting force coefficients, thereby ensuring reproducibility and clarity of the analysis. However, this constraint constitutes a limitation concerning the generalizability of the obtained results.

The optimization algorithms (GA, DE, PSO) were executed independently for 30 repetitions under standardized conditions to ensure statistically significant results. The details regarding algorithm-specific parameters, such as population size, iteration numbers, and boundary conditions, are listed comprehensively in Table 2. The selection of these parameters was guided by both a preliminary sensitivity analysis and relevant literature. The primary goal of metaheuristic algorithms is to provide effective alternatives for solving complex real-world optimization problems. Such algorithms are generally characterized by three key features: (a) the

Table 2. Optimization parameters used for GA, DE, and PSO algorithms.

Algorithm	Optimization parameters	Population number	Iteration	Boundary range
GA	Crossover probability (Pc): 0.8 Mutation probability (Pm): 0.1 Crossover rate (σ): 0.9 Mutation rate (α): 0.07	40	3000	[−10.000, 10.000]
DE	Mutation: DE/rand/1 Mutation factor (F): 0.8 Crossover probability (Pc): 0.7	40	3000	[−10.000, 10.000]
PSO	Cognitive intelligence (c1): 2 Social intelligence (c2): 2 Max. coefficient of inertia (ωmax): 0.1 Min. coefficient of inertia (ωmin): 0.01	40	3000	[−10.000, 10.000]

balance between exploration (global search) and exploitation (local search), (b) adaptation and diversity, and (c) the use of critical control operators.[42] Achieving a well-balanced exploration–exploitation trade-off is inherently challenging and depends not only on the algorithm's structure but also on the nature of the optimization problem. Belmont-Moreno[43] reported that typical values for mutation rates generally fall between 0.01 and 0.1, while population size should be chosen based on evaluation depth: smaller populations (10–30) are more suitable for quick evaluations, whereas larger populations (100+) are recommended for extensive optimization. However, he also noted that the commonly suggested population size of 100 might be unnecessarily large in many cases. Similarly, Dhal et al.[44] emphasized that the optimal values of these parameters should be tailored to both the algorithm and the problem type. Their study indicated that population size in PSO, in particular, has an unpredictable effect on performance, and that a population range between 10 and 40 is generally more suitable—larger sizes tended to reduce performance. In the present study, a range of parameter configurations was evaluated during the initial testing phase to assess their impact on convergence speed and solution accuracy. A population size of 40 and a maximum iteration limit of 3000 were found to provide a practical balance between computational cost and optimization performance. While it is acknowledged that further tuning of these parameters, potentially via automated methods or hybrid strategies, could improve convergence speed, such parameter optimization is considered beyond the scope of the current study and may constitute a standalone research effort.

The performance of metaheuristic algorithms is known to be sensitive to their parameter settings. Thus, a preliminary sensitivity analysis was conducted to identify suitable parameter values ensuring optimal convergence performance for each algorithm. The key findings from this analysis can be summarized as follows:

- **Genetic Algorithm (GA)**: GA demonstrated consistent convergence behavior when the crossover probability (P_c) was set at or above 0.8, and the mutation probability (P_m) was maintained below 0.1. This balance prevented premature convergence and promoted a thorough search of the solution space.
- **Differential Evolution (DE)**: DE algorithm performance was optimal with a mutation factor (F) of 0.8 and crossover probability (CR) of 0.7. Deviating from these settings, especially higher values of F, led to unstable divergence, while lower values caused early convergence and suboptimal exploration.
- **Particle Swarm Optimization (PSO)**: PSO showed superior stability and convergence robustness when inertia weight was linearly reduced throughout iterations, and cognitive (c_1) and social (c_2) parameters were equally balanced at a value of 2.0.

These findings suggest that tuning algorithm parameters has a significant impact on both convergence speed and solution quality. Default parameters were selected based on these observations and confirmed through repeated trials.

Results and discussion

This study proposes an optimization-based approach for predicting cutting forces in the milling process, eliminating the need for conventional offline calibration procedures. Utilizing a mechanistic force model, the cutting force coefficients are optimized through metaheuristic algorithms, including GA, DE, and PSO. The performance of each algorithm was assessed through 30 independent runs. The optimized cutting force coefficients were validated against experimental force measurements, demonstrating the high predictive accuracy of the developed model. The results indicate that the proposed method offers significant advantages in implementation simplicity, time efficiency, and cost-effectiveness.

Evaluation based on tolerance criterion and fixed iteration criterion

The performance of optimization algorithms is typically evaluated based on specific criteria such as convergence speed, solution accuracy, computational cost, and the algorithm's adaptability to the problem. Effective optimization algorithms should consistently deliver accurate solutions rapidly and with minimal computational resources, even when handling complex or high-dimensional problems.[45,46]

In this study, the performance of the algorithms was evaluated based on two main criteria. The first criterion examines the algorithm's capability to achieve a cost function value equal to or below the given tolerance level (cost ≤ 0.001). This criterion reflects the algorithm's efficiency and reliability in reaching an acceptable solution quality. The second criterion assesses the algorithm's performance by considering the best solutions obtained throughout a fixed iteration process (3000 iterations), providing insights to compare the performance of the algorithms at the point where the cost function falls below the tolerance value and their success after 3000 iterations, providing insights into each algorithm's behavior and

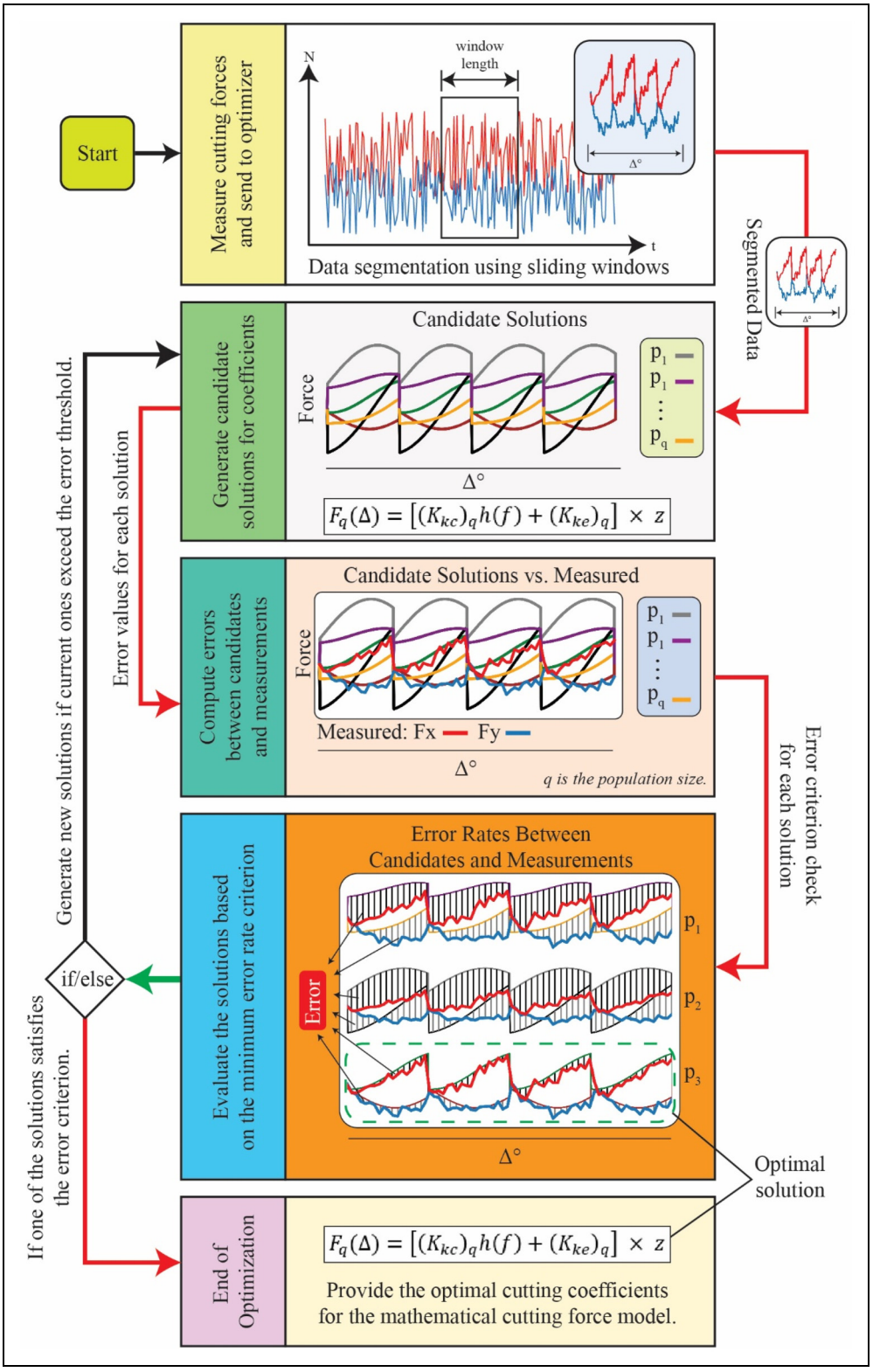


Figure 3. Schematic representation of the purposed methodology for cutting force coefficient estimation.

robustness under standardized computational constraints. The experimental evaluations were conducted using GA, DE, and PSO, selected due to their proven effectiveness in solving complex, nonlinear optimization problems with high-dimensional search spaces. Each algorithm was independently executed 30 times to ensure statistical reliability and robustness in the analysis.

The performance of the optimization algorithms was assessed based on error minimization, robustness, and computational efficiency under two criteria: achieving a predefined tolerance (cost ≤ 0.001) and operating under a fixed iteration limit (3000 iterations). As summarized in Table 3, DE achieved a 100% success rate with the lowest mean error (6.13×10^{-4}), shortest average iteration count (197), and minimal standard deviation across all runs, highlighting its effectiveness in both accuracy and consistency.

PSO also demonstrated strong reliability, converging successfully in all runs with slightly higher mean error ($7.92 \times$

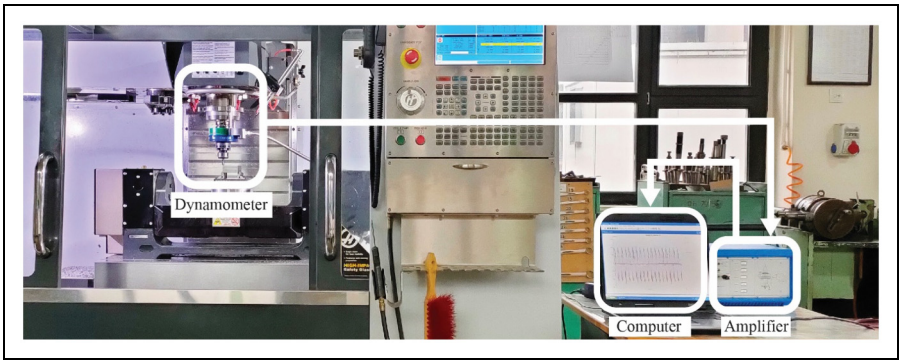


Figure 4. Experimental setup for milling operations and cutting force measurements.

Table 3. Comparative error and computational performance of optimization algorithms under two different stopping criteria.

Stopping criterion: Cost $\leq$ 0.001

Performance Error Analysis

Algo	*Min. Error*	*Max. Error*	*Mean Error*	*STD*
GA	1.713E-4	1.848E + 4	7.229E + 2	3.368E + 3
DE	2.051E-5	9.808E-4	6.128E-4	2.548E-4
PSO	1.426E-4	9.983E-4	7.924E-4	2.351E-4

Evaluation of Computational Speed

Algo	*Successful Trials*	*Average Iterations*	*Iterations (Std. Dev.)*	*Min. Iterations*	*Max. Iterations*	*Average Time [s]*	*Time (Std. Dev.)*	*Min. Time [s]*	*Max. Time [s]*
DE	30	197	8.561	179	223	5.4	0.2	4.9	6.1
PSO	30	363	74.699	198	489	9.8	2.0	5.4	13.1
GA	18	2108	616.442	735	2894	62.9	18.2	22.3	86.1

Stopping criterion: iter. = 3000

Evaluation of Algorithms Based on Error Metrics and Execution Performance

Algo	*Min. Error*	*Max. Error*	*Mean Error*	*STD Error*	*Average Iterations*	*Average Time [s]*
GA	1.865E-6	1.848E + 4	7.229E + 2	3.368E + 3	2947	87.8
DE	1.292E-26	1.292E-26	1.292E-26	0	673	18.3
PSO	1.292E-26	6.462E-26	1.486E-26	9.434E-27	1838	49.1

10^{-4}) and longer convergence time (9.8 s), but still maintaining relatively low variability. In contrast, GA displayed notable performance degradation, with only 18 successful runs out of 30, a mean error over three orders of magnitude higher (7.23×10^{2}), and the largest variance in both error and iteration counts, which indicates reduced stability and convergence reliability.

When evaluated under the fixed iteration constraint, DE continued to outperform by converging early (average: 673 iterations, 18.3 s) and achieving near-zero objective function values (1.29×10^{-26}) with zero run-to-run variance. PSO reached comparable accuracy (mean: 1.49×10^{-26}), though at a later stage (~1838 iterations). GA again lagged behind, converging near the end of the iteration window (~2947) with significantly higher computational cost (87.8 s) and error spread. These findings demonstrate that while DE and PSO are both viable for accurate and robust parameter estimation, GA requires further refinement to ensure reliable performance under constrained conditions.

The relatively low success rate and high computational cost of the GA algorithm observed in this study can be attributed to both the inherent characteristics of the algorithm and the nature of the optimization problem. A key factor behind this underperformance is GA's tendency toward premature convergence; the rapid loss of genetic diversity within the population often leads to entrapment in local minima during the early stages of the search process. Unlike GA, which explores the solution space in a relatively unguided manner, DE and PSO employ more directed search strategies, using difference vectors (DE) or velocity updates (PSO) to guide candidate solutions toward promising regions. This distinction becomes especially critical in high-dimensional, continuous, and multimodal optimization landscapes, where GA's exploration capability is limited, while DE and PSO are more likely to maintain search diversity and achieve consistent convergence.

Statistical analysis of solution sets

Z-scores were computed for each algorithm to evaluate the distribution and consistency of the fitness values obtained from 30 independent runs. The Z-score is defined as follows:

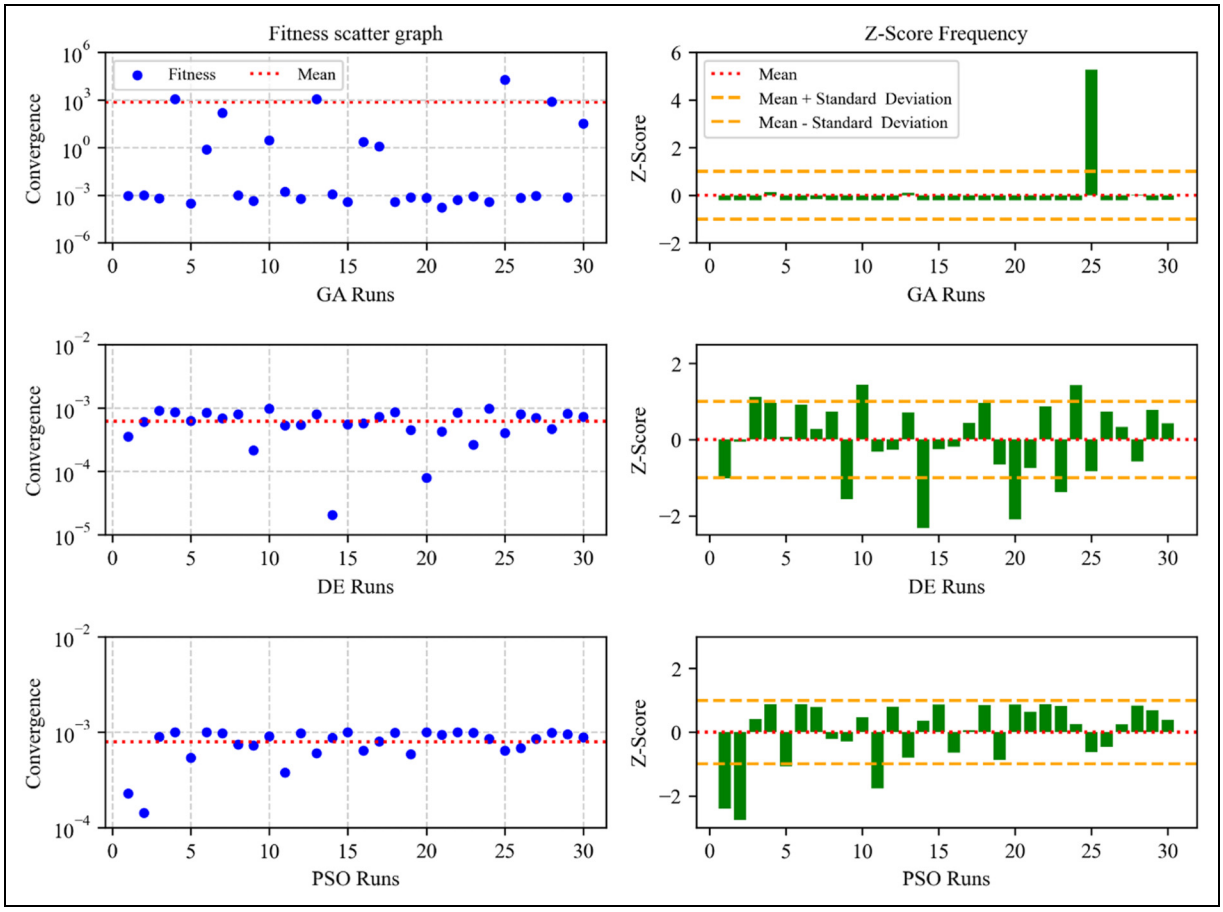


Figure 5. Solution set and Z-score values.

$$Z = \frac{(X - \mu)}{\sigma} \tag{13}$$

expresses how many standard deviations a particular fitness value X deviates from the mean μ, with σ being the standard deviation. Positive Z-scores indicate above-average performance, while negative values denote below-average results. This metric is particularly effective for detecting outliers and assessing solution robustness.

Figure 5 presents the convergence performance of 30 independent runs for each algorithm (GA, DE, and PSO), evaluated using both raw cost values (left) and their corresponding Z-scores (right). The GA results show a high degree of variability across runs. The convergence values span several orders of magnitude (from ~1e-6 to ~1e6), with the majority of runs clustering around mid-range cost values and a single extreme outlier observed at run 25. The corresponding Z-score plot confirms this inconsistency, with a notable deviation at the 25th run exceeding +5 standard deviations from the mean. Such a high Z-score, far outside the $\pm 1\sigma$ range, indicates that GA's performance is highly sensitive to initial conditions and lacks repeatability.

The PSO results also show relatively consistent convergence behavior, though with slightly more variation than DE. While the convergence values remain within a narrow logarithmic band (~1e-4 to ~1e-3), a few runs exhibit mild deviations. The Z-score distribution remains mostly within $\pm 1\sigma$, with a small number of values approaching $\pm 2\sigma$, indicating moderate sensitivity but generally reliable performance.

The DE algorithm exhibits tightly packed convergence values, all within a narrow range around 1e-3. The absence of large fluctuations across runs reflects high stability. Z-scores for DE remain well within the ± 1 standard deviation band for all 30 runs, confirming that the algorithm yields consistent results regardless of initialization. This narrow distribution indicates low variance and high robustness in performance.

The comparative Z-score analysis highlights the DE algorithm as the most statistically stable, with minimal run-to-run variation. PSO also shows consistent results but with slightly higher variability. GA, on the other hand, displays a broad and erratic performance distribution, with at least one statistically significant outlier. These observations are consistent with the convergence trends observed in Figure 6 and summarized in Table 3.

Comparative convergence analysis

Figure 6 displays the convergence profiles of the best-performing runs from each algorithm (DE, PSO, and GA), with the cost function plotted on a logarithmic scale against the number of iterations, up to the fixed limit of 3000 iterations.

The DE algorithm demonstrates the most immediate reduction in cost function value. A significant decline occurs between iterations 200 and 500, spanning several orders of magnitude. The PSO algorithm initially progresses more slowly. A noticeable decrease begins after approximately

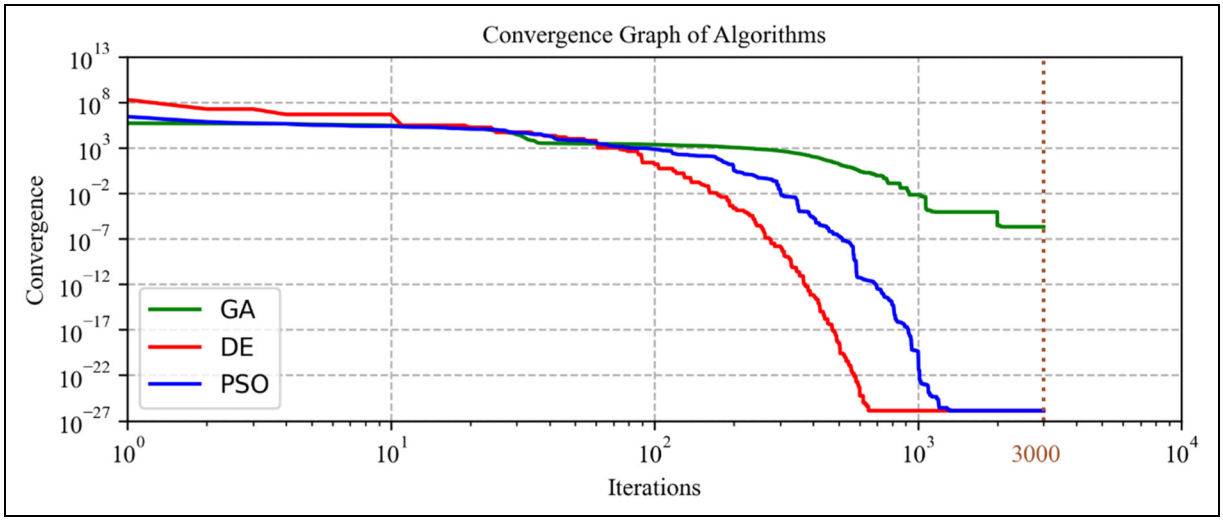


Figure 6. GA, DE and PSO convergence graph.

300 iterations, with the convergence accelerating between iterations 400 and 800. The curve approaches the final cost level of DE, but this occurs later in the iteration window. PSO thus achieves comparable final performance, albeit requiring more iterations to reach that level. The GA exhibits relatively inconsistent convergence behavior. While there is some initial improvement in the first 200 iterations, the cost function stagnates thereafter. No significant reduction is observed beyond this point. The final cost value remains multiple orders of magnitude above those achieved by DE and PSO. This behavior indicates early convergence to a suboptimal solution and a lack of effective exploration in later iterations.

In summary, within the tested configuration:

- DE achieves the lowest cost value in the fewest iterations.
- PSO converges more slowly but ultimately reaches a comparable cost level.
- GA shows limited convergence and stalls early, resulting in higher final error.

Optimization, verification, and validation of cutting coefficients

The final stage of the analysis involved identifying the optimal cutting force coefficients, which were successfully estimated by all three algorithms. The optimized values were found to be 1740.9 N/mm² for the tangential cutting coefficient (K_{tc}), 33.7 N/mm for the tangential edge coefficient (K_{te}), 510.6 N/mm² for the radial cutting coefficient (K_{rc}), and 40.1 N/mm for the radial edge coefficient (K_{re}).

Among the algorithms, DE demonstrated the highest effectiveness, achieving the optimal coefficients with the greatest consistency and speed. Its superior convergence behavior and low variability validate its suitability for high-dimensional, nonlinear parameter estimation problems such as cutting force modeling. The PSO algorithm also yielded accurate and stable results, closely matching DE's performance metrics. Its comparable robustness positions PSO as a reliable alternative in similar optimization tasks. Although the GA performed relatively poorly in speed and consistency, it converged to feasible coefficient values. With algorithmic enhancements, such as adaptive mutation strategies or hybridization with local search methods, GA's effectiveness could be significantly improved.

To assess the validity of the optimized cutting coefficients, theoretical cutting force predictions were performed and compared with experimental measurements. As shown in Figure 7, the predicted and measured force values exhibit a close match, supporting the reliability of the identified model parameters. The statistical evaluation of prediction errors is summarized in Table S1. The force component Fx showed a mean deviation of 2.823 N with relatively high variability (standard deviation: 4.954 N), and a range of −5.17 N to 9.95 N. The component Fy yielded a slightly higher mean deviation (4.347 N) but lower variability (3.634 N), with error bounds between −0.136 N and 11.76 N. These results confirm that the proposed approach can produce accurate and consistent force predictions across different directions, thereby validating the effectiveness of the optimization-based methodology. Importantly, this study focuses on the application of established metaheuristic algorithms (GA, DE, and PSO) without proposing novel variants. These methods were chosen for their demonstrated ability to handle nonlinear and high-dimensional search spaces while maintaining a balanced exploration–exploitation profile.

The findings highlight that data-driven estimation of cutting coefficients can be achieved without extensive offline calibration, offering significant time savings and process flexibility. However, the observed computational costs may hinder real-time applications. Future work should address this limitation through algorithmic acceleration techniques or hardware-based parallelization to enable online implementation under industrial constraints.

Discussion on algorithm applicability and future directions

The primary objective of this study was to evaluate the applicability of well-established metaheuristic algorithms,

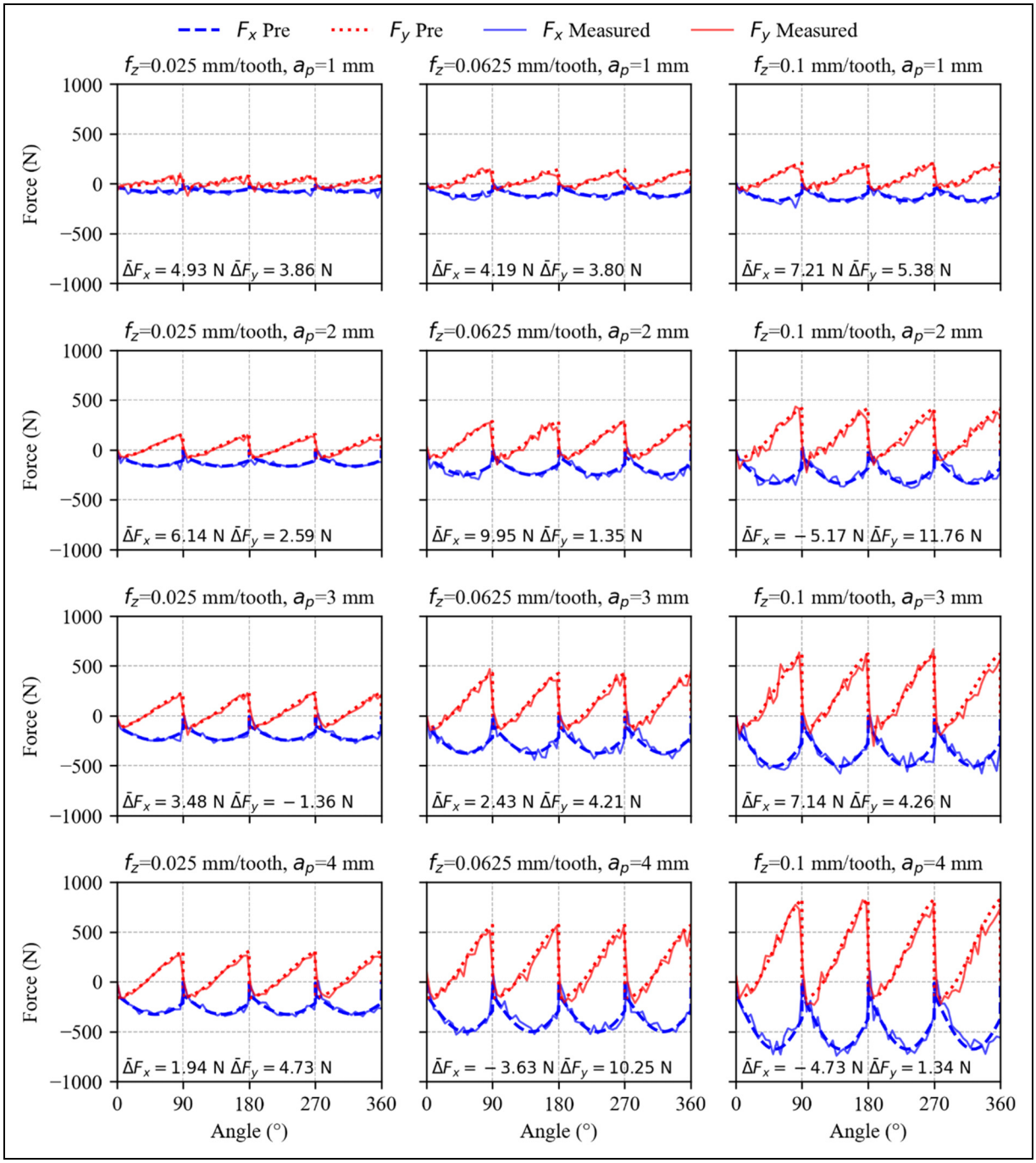


Figure 7. Comparison of measurement results and theoretical calculations.

GA, DE, and PSO, in optimizing cutting force coefficients during milling. These algorithms were selected based on their proven capability to effectively explore complex, nonlinear, and high-dimensional search spaces and their balanced exploration-exploitation mechanisms.

The findings demonstrate that all three algorithms can estimate accurate cutting coefficients without labor-intensive offline calibration, which is traditionally required for each unique tool–material pairing. This capability significantly enhances the flexibility and scalability of process modeling in machining applications. Among the tested methods, DE exhibited superior convergence speed, robustness, and predictive accuracy, while PSO provided a reliable alternative with consistent results. Although less efficient, GA converged to reasonable solutions, suggesting that its performance could be improved through algorithmic enhancements such as hybridization or adaptive parameter control.

Despite the promising results, it is important to emphasize that the current computational cost of the proposed optimization framework (while acceptable for offline analysis) may present a critical barrier for real-time industrial applications such as online adaptive control or closed-loop process monitoring. In high-throughput manufacturing environments, the latency introduced by iterative optimization can limit integration into production lines where decisions must be made within milliseconds. As such, this limitation may restrict the immediate adoption of the proposed approach in fully autonomous systems. To address this challenge, future

work should not only focus on algorithmic enhancements (e.g., surrogate modeling, adaptive stopping criteria) but also on hardware-level acceleration strategies such as GPU-based parallel computing or FPGA implementation. These directions would help bridge the gap between theoretical optimization models and their practical deployment in industrial digital twins or smart CNC systems. Future studies should focus on:

- Reducing computational time through algorithm acceleration (e.g., surrogate modeling, dimensionality reduction),
- Leveraging parallel processing or GPU computing,
- Integrating real-time sensor feedback to enable adaptive optimization,
- Extending the methodology to multi-objective frameworks, where surface quality, tool wear, or energy consumption may also be considered alongside force prediction.

Furthermore, the experimental setup employed a fixed cutting speed and a single tool–material combination (WC end mill and Ti6Al4 V alloy). This controlled configuration was chosen to minimize measurement noise under potentially unstable cutting conditions. However, this experimental constraint also limits the generalizability of the results. To enhance the robustness and broader applicability of the proposed framework, future studies should aim to validate the methodology under a wider range of cutting conditions (e.g., varying cutting speeds, feed rates, and depths of cut) and across multiple tool–material configurations. Such validation efforts are crucial to demonstrate the scalability and effectiveness of the method in diverse industrial environments.

Conclusions

This study presents a practical approach for predicting milling forces by optimizing process-specific cutting coefficients using population-based metaheuristic algorithms. The proposed strategy eliminates the need for offline pre-experimental calibration procedures by optimizing specific cutting coefficients using force measurements collected during machining. It integrates intelligent optimization algorithms with a semi-analytical mechanistic model to estimate these coefficients. This approach offers a flexible and practical method for force prediction across various cutting conditions. Moreover, monitoring the variation of cutting coefficients during machining enables real-time tracking of tool conditions. Each algorithm was executed 30 times under the stopping criterion of cost < 0.001, and their performance was evaluated in terms of convergence speed and robustness. Among the tested algorithms, the DE algorithm achieved the highest success rate, with all 30 runs converging successfully, yielding an average iteration count of 197 and an average computation time of 5.4 s. PSO similarly showed robust performance with 30 successful runs, completing optimization in 363 iterations and 9.8 s on average. In contrast, the GA algorithm achieved convergence in only 18 out of 30 runs, with a significantly higher average iteration count (2108) and computation time (62.9 s), along with considerable variability.

The cutting force prediction model developed using the optimized coefficients was validated against experimental measurements. The mean difference between the predicted and measured forces was calculated as 2.82 N for the F_x component and 4.35 N for the F_y component, with standard deviations of 4.95 N and 3.63 N, respectively. These results indicate that the model can closely approximate actual cutting forces across varying cutting conditions.

The proposed method demonstrates that metaheuristic optimization algorithms can effectively determine cutting coefficients without requiring prior calibration trials, thereby offering a flexible and scalable alternative for machining process modeling. What distinguishes this study is not only the selection of well-established algorithms (GA, PSO, DE), but also the novel integration strategy: these algorithms are embedded within a unified, in-process framework tailored for dynamic machining environments. Rather than serving as mere parameter-tuning tools, the algorithms are implemented in a way that supports real-time adaptation to machining data, ensuring robustness across different tool–material configurations. This implementation significantly enhances practical applicability, paving the way for integration into adaptive control systems and intelligent manufacturing platforms. The approach also enables online stability analysis and contributes to the development of autonomous and self-optimizing manufacturing systems. Future research may explore further improvements in computational efficiency through algorithmic refinement and hardware acceleration, thereby advancing the scalability of this framework for industrial digital twins and smart manufacturing ecosystems.

Acknowledgements

The authors would like to express their gratitude to the Gazi University Scientific Research Projects Coordination Unit for their financial support provided under the project number FGA-2024-8937.

ORCID iDs

B. Sercan Bayram https://orcid.org/0000-0002-9092-6274
Ümit Atila https://orcid.org/0000-0002-1576-9977
Gültekin Uzun https://orcid.org/0000-0002-6820-8209
Zafer Tekiner https://orcid.org/0000-0003-1400-614X
İhsan Korkut https://orcid.org/0000-0002-5001-4449
Berzah Yavuzyegit https://orcid.org/0000-0003-0759-780X

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Supplemental material

Supplemental material for this article is available online.

References

1. Chauhan S, Trehan R and Singh RP. State of the art in finite element approaches for milling process: a review. *Adv Manuf* 2023; 11: 708–751.
2. Kahya M, Ozbayoglu M and Unver HO. Precision and energy-efficient ball-end milling of Ti6Al4 V turbine blades

using particle swarm optimization. *Int J Comput Integr Manuf* 2021; 34: 110–133.
3. Zhang T, Liu Z, Sun X, et al. Investigation on specific milling energy and energy efficiency in high-speed milling based on energy flow theory. *Energy* 2020; 192: 116596.
4. Vara Prasad VU, Rao KV and Murthy P. Mechanistic models for prediction of cutting forces and power consumption considering chip geometry. *Proc Inst Mech Eng Part E J Process Mech Eng* 2021; 235: 479–488.
5. Yang J, Zhang Y, Huang Y, et al. Multi-objective optimization of milling process: exploring trade-off among energy consumption, time consumption and surface roughness. *Int J Comput Integr Manuf* 2023; 36: 219–238.
6. Berenji KR, Tehranizadeh F and Budak E. Chatter stability of orthogonal turn-milling process in frequency and discrete-time domains. *ASME J Manuf Sci Eng* 2024; 146: 091003. https://doi.org/10.1115/1.4065485
7. Mohanraj T, Uddin M and Thangarasu SK. Review on sensor design for cutting force measurement. *Proc Inst Mech Eng Part E J Process Mech Eng* 2023; 237: 455–466.
8. Mukherjee I and Ray PK. A review of optimization techniques in metal cutting processes. *Comput Ind Eng* 2006; 50: 15–34.
9. Deepak SSK. Applications of different optimization methods for metal cutting operation–A review. *Res J Eng Sci* 2012; 2278: 9472.
10. Zhang F, Xu F, Zhou X, et al. Data-driven and knowledge-guided prediction model of milling tool life grade. *Int J Comput Integr Manuf* 2024; 37: 669–684.
11. Bhandari B and Park G. Non-contact surface roughness evaluation of milling surface using CNN-deep learning models. *Int J Comput Integr Manuf* 2024; 37: 423–437.
12. Prakash Marimuthu K, Sri Kurumurthy G, Thenarasu M, et al. Minimizing residual stresses in AISI 1045 steel through optimization of cutting parameters: a particle swarm optimization approach. *Proc Inst Mech Eng Part E J Process Mech Eng* 2024: 09544089251318779. https://doi.org/10.1177/09544089251318779
13. Palanisamy P, Rajendran I and Shanmugasundaram S. Optimization of machining parameters using genetic algorithm and experimental validation for end-milling operations. *Int J Adv Manuf Technol* 2007; 32: 644–655.
14. Wu P, He Y, Li Y, et al. Multi-objective optimisation of machining process parameters using deep learning-based data-driven genetic algorithm and TOPSIS. *J Manuf Syst* 2022; 64: 40–52.
15. Radhakrishnan T and Nandan U. Milling force prediction using regression and neural networks. *J Intell Manuf* 2005; 16: 93–102.
16. Liu M, Xie H, Pan W, et al. Prediction of cutting force via machine learning: state of the art, challenges and potentials. *J Intell Manuf* 2023; 36: 1–62.
17. Omole S, Dogan H, Lunt AJG, et al. Using machine learning for cutting tool condition monitoring and prediction during machining of tungsten. *Int J Comput Integr Manuf* 2024; 37: 747–771.
18. Wang J, Zou B, Liu M, et al. Milling force prediction model based on transfer learning and neural network. *J Intell Manuf* 2021; 32: 947–956.
19. Wang B, Liu Z, Cai Y, et al. Advancements in material removal mechanism and surface integrity of high speed metal cutting: a review. *Int J Mach Tools Manuf* 2021; 166: 103744.
20. Zhang X, Pan T, Ma A, et al. High efficiency orientated milling parameter optimization with tool wear monitoring in roughing operation. *Mech Syst Signal Process* 2022; 165: 108394.
21. Altintas Y. *Manufacturing automation: Metal cutting mechanics, machine tool vibrations, and CNC design, 2nd Edition*. Second. Cambridge, UK: Cambridge Univ Press, 2012.
22. Li G, Jiang R, Sun L, et al. Prediction of bidirectional milling forces based on spindle current signals by using deep learning algorithms. *Proc Inst Mech Eng Part E J Process Mech Eng* 2025; 09544089241308122. https://doi.org/10.1177/09544089241308122
23. Zhou T, Cui H, Wang Y, et al. Multi-physics analytical modeling of the primary shear zone and milling force prediction. *J Mater Process Technol* 2023; 316: 117949.
24. Paysarvi O, Abootorabi MM and Jalili MM. Dynamic modeling and experimental investigation of hole making accuracy in the helical milling process. *Proc Inst Mech Eng Part E J Process Mech Eng* 2022; 236: 1176–1184.
25. Martellotti ME. An analysis of the milling process. *Transactions of ASME* 1941; 63: 677–700.
26. Merchant ME. Mechanics of the metal cutting process. I. Orthogonal cutting and a type 2 chip. *J Appl Phys* 1945; 16: 267–275.
27. Lee EH and Shaffer BW. The theory of plasticity applied to a problem of machining. *J Appl Mech* 1952; 18: 405–413.
28. Oxley PLB. *The mechanics of machining: An analytical approach to assessing machinability*. England: E. Horwood, https://books.google.com.tr/books?id=PaFjQgAACAAJ (1989).
29. Altintas Y and Engin S. Generalized modeling of mechanics and dynamics of milling cutters. *CIRP Ann Manuf Technol* 2001; 50: 25–30.
30. Engin S and Altintas Y. Mechanics and dynamics of general milling cutters. Part I: helical end mills. *Int J Mach Tools Manuf* 2001; 41: 2195–2212.
31. Altintaş Y and Budak E. Analytical prediction of stability lobes in milling. *CIRP Ann Manuf Technol* 1995; 44: 357–362.
32. Budak E, Altintaş Y and Armarego EJA. Prediction of milling force coefficients from orthogonal cutting data. *J Manuf Sci Eng Trans ASME* 1996; 118: 216–224.
33. Kazemi F, Song C, Clare AT, et al. A new cutting mechanics model for improved shear angle prediction in orthogonal cutting process. *ASME J Manuf Sci Eng* 2025; 147: 041003. https://doi.org/10.1115/1.4066976
34. Huang W, Li X, Zhang Y, et al. Prediction of ball end milling forces based on special turning experiment data. *Int J Adv Manuf Technol* 2018; 96: 4423–4430.
35. Tehranizadeh F, Koca R and Budak E. Investigating effects of serration geometry on milling forces and chatter stability for their optimal selection. *Int J Mach Tools Manuf* 2019; 144: 103425.
36. Farhadmanesh M and Ahmadi K. Online identification of mechanistic milling force models. *Mech Syst Signal Process* 2021; 149: 107318.

37. Mammadov B, Mohammadi Y, Farahani ND, et al. A unified FFT-based mechanistic force coefficient identification model for isotropic and anisotropic materials. *CIRP J Manuf Sci Technol* 2024; 49: 216–229.
38. Michalewicz Z, Janikow CZ and Krawczyk JB. A modified genetic algorithm for optimal control problems. *Comput Math Appl* 1992; 23: 83–94.
39. Kennedy J and Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, 1995, pp.1942–1948: IEEE.
40. Stron R and Price K. Differential evolution–a simple and efficient adaptive scheme for global optimization over continuous spaces. *J Glob Optim* 1995; 11: 341–359.
41. Paliwal V and Babu NR. Prediction of stability lobe diagrams in high-speed milling by operational modal analysis. *Procedia Manuf* 2020; 48: 283–293.
42. Yang X-S and He X. Nature-inspired optimization algorithms in engineering: overview and applications. *Nat-Inspired Comput Eng* 2016; 637: 1–20.
43. Belmont-Moreno E. The role of mutation and population size in genetic algorithms applied to physics problems. *Int J Mod Phys C* 2001; 12: 1345–1355.
44. Dhal KG, Das A, Sahoo S, et al. Measuring the curse of population size over swarm intelligence based algorithms. *Evol Syst* 2021; 12: 779–826.
45. Warnakulasooriya K and Segev A. Comparative analysis of accuracy and computational complexity across 21 swarm intelligence algorithms. *Evol Intell* 2025; 18: 18. https://doi.org/10.1007/s12065-024-00997-6
46. Kuang T, Hu Z and Xu M. A genetic optimization algorithm based on adaptive dimensionality reduction. *Math Probl Eng* 2020; 2020: 8598543.