ISSN:2633-4828 Vol. 6, No. 1, January, 2024

International Journal of Applied Engineering & Technology

A Systematic Framework Utilizing Dynamic Programming Approach for Enhanced Operational Integrity in Machine Tools

Olugbenga Adegbemisola Aderoba
Tshwane University of Technology, Pretoria, South Africa, aderobaoa@tut.ac.za, olugbenga.aderoba@gmail.com
Landmark University, Omu-Aran, Nigeria,

Khumbulani Mpofu
Tshwane University of Technology, Pretoria, South Africa, mpofu@tut.ac.za

Ilesanmi Afolabi Daniyan
Achievers University, Owo, Nigeria, afolabiilesanmi@yahoo.com

Buliaminu Kareem
Federal University of Technology, Akure, Nigeria, bkareem@futa.edu.ng

Date of Submission:22 November 2023 Revised: 02 January 2024 Accepted: 12 January 2024

How to Cite: **Aderoba, O. A, *et. al.*, (2024): A Systematic Framework Utilizing Dynamic Programming Approach for Enhanced Operational Integrity in Machine Tools, International Journal of Applied Engineering & Technology 6(1), pp.1490-1502**

Abstract - **Many machine tools experience fluctuations in performance, leading to inconsistencies in product quality and production output. The absence of a systematic framework exacerbates this issue, hindering the attainment of reliable and consistent operational integrity. This research introduces a systematic framework leveraging a dynamic programming approach to enhance operational integrity in machine tools. The framework addresses challenges related to operational performance variations, adaptability to changing operational requirements, and unplanned downtime. By integrating dynamic programming algorithms, the framework optimises operational machine tool parameters in real-time, ensuring responsiveness to fluctuating production demands. Results from rigorous testing demonstrate the efficacy of the systematic framework, showcasing notable enhancements in machine tool performance and overall operational integrity. This research contributes to the ongoing discourse on advanced methodologies for optimising manufacturing processes, with implications for industries seeking to achieve sustained operational excellence in their machine tool operations.**

Index Terms - **Machine tools; operational integrity; dynamic programming; systematic framework; optimisation**

INTRODUCTION

Machine tools, serving as the backbone of manufacturing, are pivotal in shaping end-product quality and efficiency through their operational integrity. They are used for manufacturing parts in various industries and they include lathes, drilling machines, milling machines, grinders, and more [1], [2]. Machine tools play a fundamental role in achieving precision and accuracy in manufacturing, while operational integrity ensures that the machine tools maintain their precision and accuracy over time. Any deviation from the intended specifications can lead to defective products, increased scrap rates, and compromised quality. Operational integrity, a cornerstone of manufacturing excellence, is particularly critical for machine tools as it ensures the consistent and reliable performance necessary for the efficiency, reliability, and accuracy of manufacturing processes. Reliability ensures consistent operation of machine tools by minimizing breakdowns or errors, while efficiency optimizes operational aspects to reduce waste, energy consumption, and operation time. Accuracy, in turn, enhances the precision of machine tools, ensuring a consistent output that meets desired specifications. These concepts underscore the importance of machine tools being in a state of completeness, wholeness, and good working conditions, which is crucial for maintaining manufacturing operations' overall functionality and effectiveness.

Nevertheless, the constraints posed by dynamic operational requirements, uncertainties in production demands, and the necessity for adaptability have highlighted the shortcomings of conventional, static approaches to machine tools. Challenges such as unplanned downtime, performance variations, suboptimal decision-making and insufficient responsiveness to changing conditions present significant obstacles to attaining high operational integrity. Low operational integrity can introduce variations in product dimensions, surface finish, and other critical parameters, potentially leading to quality issues and customer dissatisfaction. The maintenance of high operational integrity is crucial to ensure that machine tools consistently generate high-quality outputs, which is imperative for meeting quality standards and fulfilling customer expectations. The lack of a systematic approach to operational integrity not only jeopardizes the consistent fulfilment of customer expectations but also poses a threat to the erosion of trust and market competitiveness. To ensure sustainable machining processes, it is imperative to embrace an innovative approach that involves a comprehensive assessment for identifying optimal parameters [3]. Hence, there is a pressing need for the development and implementation of a systematic framework in the form of a structured approach that leverages dynamic programming principles in the context of machine tool operation. This study presents a framework which aims to optimize decision-making processes, enhance adaptability to evolving manufacturing environments, and ultimately elevate operational integrity in machine tools. By systematically integrating dynamic programming, the objective is to optimise process parameters, enhance integrity and improve precision, thereby establishing a robust foundation for reliable and efficient manufacturing operations.

In contemporary manufacturing systems, the predominant focus is on adjusting machine tool parameters rather than just the process variables determining the machined workpiece [4]. Model-based self-optimization systems emerge as a fundamental concept aimed at governing product quality by establishing connections between setting parameters, process variables, and ultimately, the machined surface quality. Hence, optimizing the process variables in machine tools is crucial because it impacts the quality of machining, efficiency, and the cutting lifespan of machine tools [5]. Achieving desired levels of machining quality and efficiency in process optimization necessitates adaptive control, wherein machining parameters are adjusted. This adjustment considers the operational integrity parameters of machine tools to ensure optimal performance. In a machine controller, the operation planning process can be genuinely adaptive, capable of dynamically altering the process plan in response to the dynamics observed during the actual machining process [6]. By implementing a process model, the system gains the ability to anticipate its future behaviour and promptly adapt, resulting in an improved process design that effectively enhances and optimizes the manufacturing process. This improvement results in reduced manufacturing costs, increased economic and production efficiency, and greater repeatability of the manufacturing process [4].

The dynamic behaviour of machine tools plays a pivotal role in influencing key machining outcomes, including reliability, efficiency, and accuracy [7]. To ensure consistency and effectiveness in manufacturing processes, it is essential to develop a systematic framework augmented with dynamic programming. This approach becomes critical for managing complex systems, standardizing processes, reducing waste, and facilitating continuous improvement. Consequently, it contributes to enhancing quality, precision, and operational integrity in machine tools. Dynamic programming plays a key role within this framework by offering optimization tools that enhance decision-making, resource allocation, and adaptability to dynamic operational conditions, ultimately contributing to an enhanced operational integrity of machine tools. Dynamic programming is a mathematical optimization technique used to solve problems that can be broken down into smaller, overlapping subproblems [8]. Consistent operational integrity results in reliable production schedules and high-quality products, fostering customer satisfaction and trust. The motivation behind developing a systematic framework is to provide a structured approach to managing complexity, mitigating risks, and improving operational efficiency.

Significant research efforts have focused on enhancing self-tuning processes through evaluated sensor signals for quantitative analysis of monitoring quality [9]. This system was further improved at a turning station by employing an intelligent production method, which incorporates smart IoT sensors and artificial intelligence techniques to facilitate digital sensing, trend analysis, and informed operational decision-making. [10]. An autonomous control system was also developed for maintaining the condition of metal-cutting machines [11]. Autonomous optimization of parameters is key in observing industrial systems, as they significantly contribute to Industry 4.0 advancement [12]. By using information from the machine and the process, self-optimizing machining systems can autonomously modify process and machine settings. This automated adjustment not only meets but can also improve upon predefined assessment criteria, thereby removing the necessity for an operator's manual intervention. Other works of literature also focused on enhancing machine tool efficiency and accuracy, but there's a lesser focus on the dynamic adaptation of these systems to evolving operational requirements and production demands.

This includes [13] using a Dynamic Programming algorithm for optimal power split trajectory, [14] establishing a degradation trajectory model for CNC machine tools, and [15] improving high-performance cutting with adaptronic systems. [12] emphasized optimizing within control systems for adaptability, while [16], [17] discussed dynamic programming for sequential decision problems. [4] noted optimization time reliance on algorithms, models, and computing hardware. [18], [19] addressed optimal machining parameters and tool selection using micro-morphological characteristics. Additionally, genetic algorithms and actor-critic reinforcement learning frameworks studied by [18], [20] are used for optimizing cutting conditions and energy-efficient batch machining. Despite these advancements, the challenge remains in how these systems can dynamically adjust to changing operational needs and production demands. The integration of dynamic programming to enhance adaptability, integrity and responsiveness in machine tools is an area that is not extensively covered. Also, the specific role of dynamic programming in continuously improving and sustaining operational integrity in machine tools is an area that needs more exploration. Therefore, the primary study's focus is to create a systematic framework based on dynamic programming principles to enhance the integrity of machine tools and optimize decision-making processes regarding their operation and maintenance.

The primary objective is to utilise dynamic programming principles to optimize decision-making processes related to maintenance, operational parameters, and resource allocation. This framework aims to enhance the operational integrity of machine tools, mitigating the identified challenges and fostering a manufacturing environment characterized by reliability, efficiency, and accuracy. Addressing the above challenges through the proposed systematic framework will not only optimize machine tool performance but also contribute to the broader goals of reducing operational costs, improving product quality and enhancing overall customer satisfaction. The development of such a framework represents a crucial step toward achieving sustained operational excellence in the realm of machine tools within modern manufacturing contexts. Other specific objectives achieved from this study include

a. the correlation between varying operational parameters and machine tool performance, identifies key parameters influencing integrity, quality, and overall effectiveness and uses a case study to understand the dynamic nature of operational parameters in manufacturing environments.
b. Formulate a mathematical model leveraging dynamic programming principles for real-time decision-making and integrate key operational parameters into the decision-making framework.
c. Develop algorithms capable of optimizing operational settings based on dynamic conditions, considering factors such as production demand and technological changes.
d. Evaluate the performance of the Framework using a case study by testing the decision-making framework with traditional static approaches, and assess the framework's ability to adjust operational parameters in response to different environmental factors.

The systematic framework utilizing dynamic programming proposed in this research significantly contributes to advancing our understanding and practices related to enhancing operational integrity in machine tools. The key contributions of this study are:

a. This framework facilitates a more efficient response to varying production demands and technological advancements, ensuring that machine tools operate at their optimal settings in a dynamic manufacturing environment.
b. The research emphasizes the integration of proactive maintenance strategies within the framework. By employing dynamic programming to recommend remedial actions in the form of proactive maintenance activities based on real-time conditions, the framework minimizes unplanned downtime, reducing operational disruptions and enhancing the overall reliability of machine tools.

The systematic framework contributes to establishing industry best practices for maintaining and optimizing machine tool performance. By emphasizing the importance of dynamic decision-making, the research provides a foundation for manufacturing enterprises to enhance their operational integrity, reduce costs, and improve overall competitiveness for industries seeking to achieve sustained operational excellence in their machine tool operations.

METHODOLOGY

Establishing a baseline for machine tool performance is a critical step in ensuring enhanced operational integrity and detecting deviations in a manufacturing environment. This process involves creating a reference point or standard against which a machine tool's performance can be measured [21]. The methodology in this study systematically identifies parameters that significantly affect productivity performance metrics.

2.1 Machine Tool Operational Integrity

2.1.1 Key Machine Tool Operations

Identifying and monitoring key operational parameters is fundamental to the success of the systematic framework. The aspects of machine tool operations that could benefit from optimization are cutting speed and feed rates, cutting tool wear and life management, vibration and stability, coolant and lubrication management, energy efficiency, and tool path planning, amongst others as shown in Figure 1. The selection process entails a thorough analysis that encompasses the machine tool's performance requirements, historical data, and industry standards, focusing on key parameters.

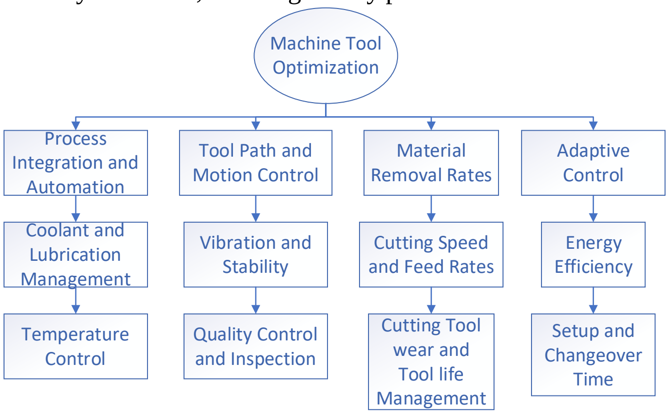


FIGURE 1
AREAS OF MACHINE TOOLS OPTIMISATION

In this study, signals from the state sensors to the executive mechanisms are sent through distributed units for component control in the machine tool. These units are connected to the central autonomous control unit, which analyzes the signals regarding the state of the components, assesses the state, and makes decisions regarding the best actions to support system functioning [11]. Improving the state control of machine tools crucially involves monitoring parameters to assess machine tool integrity in real-time, compare it with industry standards, and automate decision-making for necessary restorative measures through remedial actions. The output quality, together with the dimensional accuracy of the machine tool are the most critical factors in determining the performance of machine tools as determined by experimental tests such as surface roughness. The combination of all the factors can affect the output quality and dimensional accuracy of the machine tool and invariably its integrity as indicated in Figure 2.

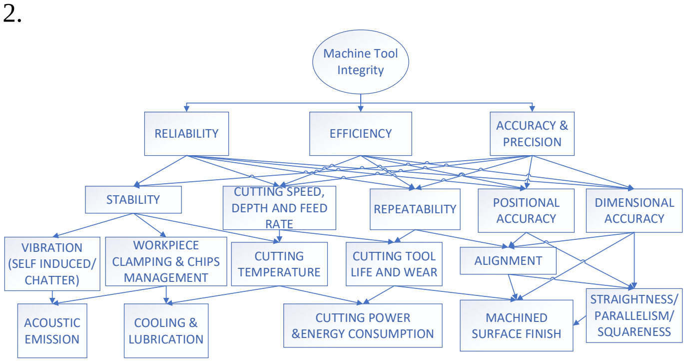


FIGURE 2
COMBINATION OF FACTORS THAT AFFECT MACHINE TOOL INTEGRITY

2.1.2 *Formulation of Machine Tool Integrity*

Suppose a parameter measures P_{an} and its acceptable benchmark is P_s. Its Integrity, I_N is

$$I_N = \frac{P_{an}}{P_s} \quad (where\ P_s > P_{an})$$

If P_{an} is greater than P_s, the Integrity automatically becomes, $I_N = 1$

Let the total integrity of a machine tool be, I_T and let there be N parameters for measurement and M of those parameters be in the category of death-knell. Therefore, N-M will be in the Pseudo Deathknell/living characterisation. The total machine tool integrity, I_T is as shown in Equation 1

$$I_T = I_{M1} \times I_{M2} \times \left[\frac{I_{(M+1)} + I_{(M+2)} + I_{(M+3)} + \cdots I_{(N-M)}}{(N-M)}\right] \tag{1}$$

2.1.3 *Key Performance Indices*

This is crucial for machine tools and their intended applications may include accuracy, repeatability, surface finish, cutting speed, and tool life. These performance metrics include parameters as indicated in Figure 2. Figure 3 presents the cause-and-effect fishbone diagram of a machine tool using surface roughness as a performance metric. The diagram highlights the major factors that promote machine tool functionality as cutting tool properties, machining parameters, workpiece properties and machining phenomenon and the effect.

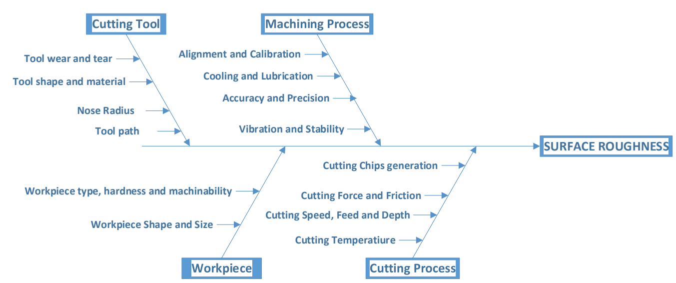


FIGURE 3
THE CAUSE-AND-EFFECT FISHBONE DIAGRAM OF A MACHINE TOOL USING SURFACE ROUGHNESS AS PERFORMANCE METRIC

Online process control enhances stability by accounting for runtime uncertainties and disturbances, facilitating safe operation at the machine's technological limits. Although it allows adaptation to changing process conditions, this doesn't assure the optimal functioning that self-optimization aims for. In contrast, advanced control methodologies leverage explicit process models for real-time optimization, achieving autonomous optimality.

2.2 Multi-Stage Decision Problem Description

According to [17], A multistage decision problem consists of a series of linked single-stage processes, where the output of each process becomes the input for the next, with no recycling of these outputs to earlier stages. In this scenario, a decision made at a specific time, t, is shaped by prior decisions and typically impacts subsequent decisions. The following are the major problems that the proposed framework seeks to solve:

i. **Optimization of Operation:** Dynamically determine the optimal operating parameters for different tasks to enhance precision and minimize wear and tear.
ii. **Adaptability and Learning:** The system could learn from past operations to continuously improve its performance over time.
iii. **Reducing downtime, enhancing precision, improving maintenance protocols, and ensuring consistent performance.**
iv. **Fault Diagnosis and Recovery:** Quickly identifying and rectifying operational issues, possibly through a decision-making process informed by dynamic programming.
v. **Prescriptive Maintenance:** Using dynamic programming to recommend remedial action in machine tools to enhance their integrity

2.3 Dynamic Programming Approach

Dynamic programming is a problem-solving method that effectively tackles complex issues by decomposing them into simpler subproblems. This approach is especially well-suited for optimization problems characterized by sequential decision-making, where the problem can be segmented into stages exhibiting a recursive relationship. Dynamic programming is a bottom-up algorithmic approach that solves problems by merging the solutions of overlapping subproblems. It effectively solves each subproblem once, storing the results in a table to prevent redundant recalculations each time the same subproblem is encountered [22].

In dynamic programming, breaking down the operational process of machine tools into smaller, manageable subproblems is crucial, as solving these subproblems is key to addressing the overall problem. The development of recursive algorithms that establish a relationship between the solution of each subproblem and its successor is central, to building up solutions for these subproblems towards solving the entire problem. This study applies a break of a complex multistage decision problem into simpler single-variable problems as shown in Figure 4. Decomposing the problem into sub-problems enables the optimal solution of resolving each smaller problem. This method is based on the idea that smaller problems are more manageable than the complex original ones. The approach is theoretically grounded in Bellman's principle of optimality through Bellman's equation. For effective optimization, the models must accurately represent the entire process, connecting parameter settings directly to product quality via detailed process parameters [4].

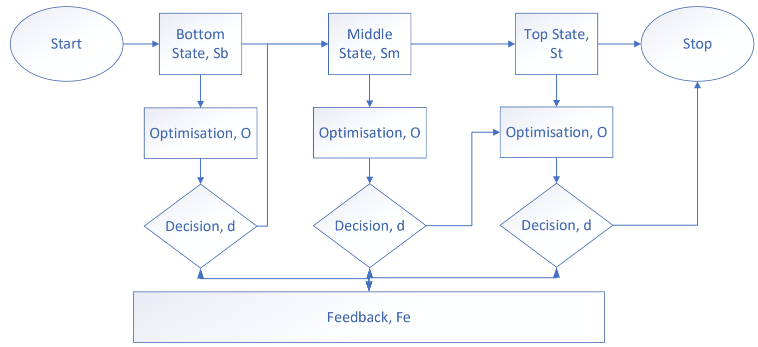


FIGURE 4
MULTI-STAGE DYNAMIC PROBLEM OPTIMISATION

The development of the dynamic programming model involves identifying relevant state and decision variables, formulating a comprehensive cost function, and implementing dynamic programming principles to optimize decision-making processes for enhanced operational integrity in machine tools. The modelling of the operational process entails the creation of mathematical models that represent the machine tool operations.

2.3.1 Identification of State Variables

State variables represent the key factors that characterize the machine tool's current state, ω and the manufacturing environment. In the context of machine tools and operational integrity, the state variables include Machine Health Status or condition of the machine tool, which may involve factors such as wear, temperature, vibration, or other diagnostic measures. Production-related variables reflecting the workload of the machine tool are expected to handle external factors influencing machine tool performance, such as ambient temperature, humidity, or specific material characteristics in the manufacturing process. In this sequential system, the output of stage t+1 corresponds to the input of stage t in a forward sequence, and similarly, the output of stage t aligns with the input of stage t−1 in a backward sequence as indicated in Figure 4. The transformations of states and the associated returns functions in this system are represented accordingly in equations 2 to 5.

$$\omega_{t+1} = g_t(\omega_t, \delta_t) \quad or \quad \omega_t = g_t(\omega_{t-1}, \delta_t) \tag{2}$$

$$H_{t+1} = h_t(\omega_t, \delta_t) \quad or \quad H_t = h_t(\omega_{t-1}, \delta_t) \tag{3}$$

$$\omega_{t-1} = g_t(\omega_t, \delta_t) \quad or \quad \omega_t = g_t(\omega_{t+1}, \delta_t) \tag{4}$$

$$H_{t-1} = h_t(\omega_t, \delta_t) \quad or \quad H_t = h_t(\omega_{t+1}, \delta_t) \tag{5}$$

Where equations 2 and 3 represent forward recursion and equations 4 and 5 represent backward recursion. t signifies the period from $t = 1, 2, \dots, T$., $\delta_i \in \Delta$ represents the vector of control variables used in decision-making, H is the return function, and ω_t is the state variable at stage t.

2.3.2 Identification of Decision Variables

Decision variables are the parameters that can be adjusted or controlled to influence the system's state in the form of choices or actions that can be taken at each stage to influence the system's state. In the context of machine tools, decision variables may include ***Maintenance Schedules*** which determine when, how and what remedial actions can be taken during monitoring and inspections or component replacements; ***Operational Settings:*** Adjusting parameters like speed, feed rate, or tool changes to optimize performance in response to changing conditions, and ***Resource Allocation*** which decides how resources, including manpower and spare parts, should be allocated to address maintenance needs or unexpected failures. However, this study focuses on the maintenance schedules and operational settings.

Due to the influence of inputs on the decisions made within the system, the return function, H is indicated in (6)

$$H = h(\Omega, \Delta) \tag{6}$$

Where Δ *=the decision space vector* and Ω is *the state space vector*.

In a t-multistage decision process, the input state vectors for the forward and backward t^{th} stages are represented by Ω_{t+1} and Ω_{t-1}, respectively, while the output state vectors are denoted by ω_i.

2.3.3 Formulation of the Cost Function

The cost function quantifies the objective to be minimized or maximized over time. In the context of machine tools, it represents the overall cost associated with the chosen decision variables. These components include the operational cost in the form of maintenance cost which encompasses, downtime cost which quantifies the financial impact of machine downtime, and quality cost which incorporates factors related to product quality, including defects and rework. The goal is to minimize the cost function over time to maximize the operational integrity. This cost function is represented in (7) while the constraints of the multistage problem is indicated in (8)

$$f(H_t) = \sum_{t=1}^{T} f_t(H_t) = \sum_{t=1}^{T} H_t(\omega_{t+1}, \delta_t) \tag{7}$$

$$Minimize\ Z = \sum_{t=1}^{T} f_t(\delta_t)$$

$$toSubject\ to: \sum_{t=1}^{T} (\delta_t \le b) \quad where\ \delta i \in \Delta, \delta i \ge 0, t = 1,2,3, \dots T \tag{8}$$

δ_t signifies the decision variables, each contributing to the overall return. b denotes the potential values representing the quantity of resources available for allocation.

2.3.4 Dynamic Programming Model Formulation:

The dynamic programming model integrates the identified state variables, decision variables, and cost function into a mathematical framework. It involves defining transition equations that describe how the system's state evolves based on the decisions made. The decision variables, objective function, and constraints specific to enhancing operational integrity in machine tools include:

i. State Transition Equation: this describes how the state variables change over time, considering the impact of decision variables and external factors as shown in equation (9).

$$f_T(\omega_T, \delta_T) = Minimize\ Z = \sum_{t=1}^{T} [f_t(\omega_t, \delta_t) + f_T(\omega_t)] \tag{9}$$

$$Subject\ to: \quad \sum_{t=1}^{T} \delta_t \le \omega, \omega > 0$$

$$\omega_{t+1} = f_t(\omega_t, \delta_t) \quad or \quad \omega_{t-1} = f_t(\omega_t, \delta_t)$$
$$\omega_{t+1} = (\omega_t - \delta_t) \quad or \quad f_t(\omega_t, \delta_t) = \omega_t - \delta_t$$
$$\omega_t \in \Omega_t, \delta_t \in \Delta_t(\Omega_t), \delta_t \ge 0, t = 1,2, \dots. T.$$

ii. Bellman Equation: Fundamental to dynamic programming, it expresses the cost function's best possible value in any given state as a function of the value of the cost function in subsequent states. The goal in a multistage decision problem is to identify δ_t within Δ to optimize a function, f, which is formulated based on the returns of each stage, expressed as $f(H_t)$ for stages $t = 1, 2, \dots, T$ as indicated in equation (9)

The decision maker's goal, given that available actions are dependent on the initial system state, is to choose a suitable decision variable, represented as ($\delta_1, \delta_2, \dots, \delta_n$), within an allowable range of actions (specifically, $\delta t \in \Delta$). This choice aims to improve the system's performance over a planning period covering T stages. This set of decisions or remedial actions is termed a policy which effectively reduces the objective function subject to specified constraints as indicated in (9).

In the dynamic programming model, the objective function encompasses both the cost function and the transition dynamics, thereby reflecting the model's overall goal. An optimal policy is characterized by a unique attribute that remains constant regardless of the current state. The decisions that follow should collectively form an optimal policy relative to the resulting state from the initial decision [13], [17]. The primary objective is to determine the optimal machining process settings by comparing the efficacy of various strategies on a single machine tool, using a specific approach to enhance the integrity of machine tools. The focus within this system is on the machined product, whose quality is evaluated using a surface roughness performance analysis model.

2.4 Optimization Algorithm and Performance Index Calculation

2.4.1 Optimization Algorithm:

The dynamic programming algorithm aims to find the sequence of decision variables that minimizes the cost function and maximizes the objective function of enhancing the integrity, thus optimizing the system's performance over time. The dynamic programming algorithm employs four systematic steps, which involve characterizing the structure of an optimal solution, recursively determining its value, generating the optimal value, and constructing an optimal solution as shown in Figure 5.

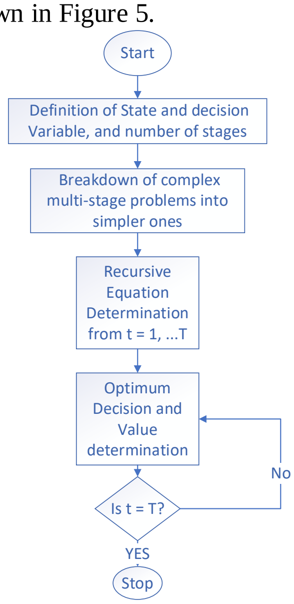


FIGURE 5
FLOWCHART OF THE DYNAMIC PROGRAMMING ALGORITHM

The construction of the cost function, which is based on the estimated distribution of failure times, is aimed at tackling the optimization problem [23]. Self-optimization in a technical machining system refers to its capability to autonomously adapt and adjust its operations, enhancing performance in terms of accuracy and precision, process stability and reliability, all without requiring operator intervention [12].

2.4.2 Performance Index Calculation

Machine tool performance is quantified through the definition of a performance index, which is based on selected metrics. This index should be sensitive to deviations from the established baseline. This baseline represents the expected or optimal values for the chosen performance metrics under normal operating conditions. It also involves establishing and monitoring a baseline for machine tool performance using dynamic programming and provides a systematic approach to ensuring operational integrity. Surface roughness, Ra, selected for this study is the primary parameter used to characterize average surface roughness, representing the total of absolute values in the roughness profile, *x*, measured over a specific evaluation length, *l* as indicated in (10).

$$R_a = \left(\frac{1}{l}\right)\int_0^l |z(x)|dx \tag{10}$$

2.5 Systematic Framework to Enhance Machine Tool Operational Integrity

2.5.1 System Architecture

Figure 6 presents the architecture of the proposed systematic framework.

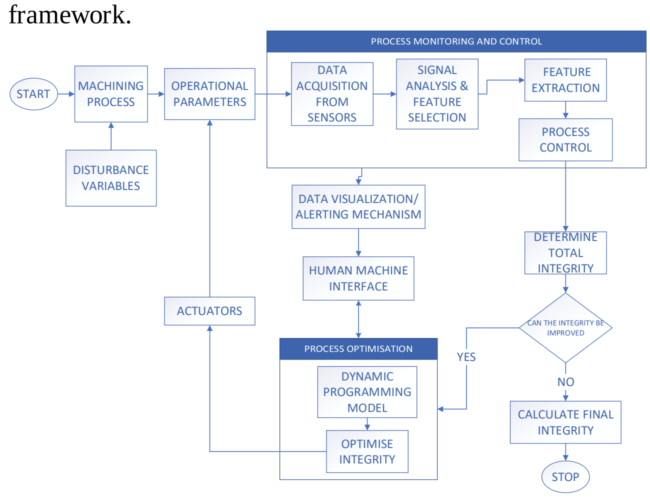


FIGURE 6
THE ARCHITECTURE OF THE PROPOSED SYSTEMATIC FRAMEWORK

2.5.2 Systematic Framework

This implies a structured methodology or set of principles designed to address a particular problem or set of problems in the domain of machine tools. This systematic framework as indicated in Figure 6 utilizes dynamic programming algorithms to optimize decision-making processes in real-time, enhancing integrity in machine tool operations. Utilizing a systematic approach, the analysis of requirements guides the development of subsystems, encompassing optimization, actuators, and sensors, with the overarching goal of optimizing the entire process. This analytical process identifies models that translate signals from the process into variables, establishing crucial connections between these variables and the resulting product quality. Through this framework, quality criteria such as surface roughness, and the major influencing process variable are measurable.

The current machine tool states are obtained from collected data to indicate the current condition of machine parameters and their dependent variables. The data from the sensors is processed to calculate the integrity and if necessary fed into the dynamic programming model in real time for optimisation. In the initial stage, the focus is on collecting data for performance analysis of various input variables within the machining process. To enable real-time data collection, a network of sensors is strategically deployed throughout the machine tool. These sensors are chosen based on their ability to accurately measure the identified key parameters. In manufacturing systems, conventional sensors typically indicate the activity status of a process but lack information about the specific operating point concerning quality [4]. Enhancing machining characteristics, minimizing human effort and errors, optimizing production time, and meeting the demands of Industry 4.0 is imperative, necessitating the adoption of advanced sensor systems. [24].

Based on the incoming data, the dynamic programming model is updated to reflect the current state of the machine tool. Recommended remedial actions or adjustments that can be made to improve operational integrity, such as tool replacement, speed adjustments, and cooling system modifications amongst others are optimised for the best decision. These remedial actions are also coded at the back end of the Python Program. The dynamic programming model generates a policy that dictates the optimal actions to be taken under specific conditions. The recommended actions are implemented in real time to optimize operational integrity. Other remedial actions may involve making adjustments to machining parameters, activating maintenance protocols, or even triggering automated self-correction mechanisms. The model continuously adapts to changing conditions, allowing it to make informed decisions based on the most recent data. A reward function that quantifies the benefit or cost associated with each state-action pair would be established whose goal is to maximize operational integrity while minimizing negative impacts. The system monitors the outcomes of the implemented actions and compares them to the expected results. If discrepancies are observed, the system learns from the feedback and updates the dynamic programming model accordingly. This ensures continuous improvement and adaptation to evolving machine tool conditions.

During the decision-making algorithm stage, the problem is broken down into two key steps, among which are modelling and multi-objective optimization. In the modelling phase, the focus is on depicting the correlation between process design factors, with a special emphasis on assessing integrity. Models are crafted to evaluate real-time product quality using sensor data, and when direct online assessment isn't feasible, surrogate process variables closely tied to product quality are employed. This surrogate approach guides the design of subsystems, encompassing sensors, actuators, and optimization challenges [4]. At a higher level, the model-based optimization system integrates self-optimization by determining optimal operating points and strategies. This system's inputs are the overarching goals of production facilities, with outputs including internal objectives and the refinement of control parameters from sensors and actuators which establishes a dynamic control loop utilizing data analysis [25].

Indicator models developed for each metric are first used to create an integrated model specific to that metric. These models are then utilized in multi-objective techniques to determine the best set of process parameters. This study adopts a strategy that recommends corrective actions, derived from optimal process values, to achieve specific objectives and adhere to constraints. Machine tool accuracy is affected by a range of error sources, including conditions of cutting tools, environmental factors, and operational components of the machine tool [26]. Monitoring devices enable adjustments to control parameters, like cutting speed and feed rate, to impact the ongoing process [12]**.** The three major functions of the activities implemented while working with the framework are:

1. **Deviation Detection:** Regularly compare the real-time performance data with the established baseline by utilising the performance index and dynamic programming algorithm to detect any deviations from the baseline.
2. **Alerts and Notifications:** Implement an alert system that triggers notifications when significant deviations are detected. This allows operators and maintenance personnel to address issues promptly, reducing downtime and preventing defects in manufactured parts. Implementing an effective alarming and alerting system is crucial for proactive intervention.
3. **Continuous Improvement:** Periodically update the baseline by re-analysing performance data and adjusting the dynamic programming algorithm to indicate the current machine tool integrity. This ensures that the baseline remains accurate and relevant as the machine tool undergoes wear and tear or operational changes.

By combining real-time monitoring with dynamic programming, the systematic framework allows machine tools to operate with enhanced integrity, as the system can respond promptly to changing conditions and optimize performance in a dynamic manufacturing environment.

SYSTEM IMPLEMENTATION ON A CASE STUDY

A turning operation was carried out on a Boxford Lathe using mild steel workpiece material with an HSS tool as shown in Table 2. Several sensors as identified in Table 1 were connected to the lathe to measure different machine parameters. The cylindrical workpiece is 80 by 150 mm and the workpiece is held on the spindle while the pressure sensors monitor the gripping force.

TABLE 1
APPLIED SENSORS ON CASE STUDY

S/No.	Name of Sensor	Function
1	Sound and Vibration Sensors	Monitor machine tool acoustics and vibrations to detect irregularities or excessive wear in components.
2	Temperature Sensors	Measure the temperature of critical components to prevent overheating and ensure optimal operating conditions
3	Speed and Feed Sensors	Collect spindle speed and feed rate data in real-time for accurate control and optimal performance.
4	Tool Wear Sensors	Monitor cutting tools' wear to facilitate timely replacements and prevent quality issues.
5	Roughness sensor	To monitor tool wear behaviour with roughness
6	Voltage and current sensor	Monitor Cutting Power
7.	Pressure Sensors	To Monitor hydraulic and pneumatic pressures within the coolant and the gripping force of the spindle

The dynamic programming algorithm developed was executed using the Python programming language. It involved creating functions and modules for initializing the problem, managing state transitions, computing rewards, and updating the solution, all tailored to the specifics of the chosen algorithm. Recommended remedial actions or adjustments that can be made to improve operational integrity, such as tool replacement, speed adjustments, or cooling system modifications were also coded at the back end of the Python Program. A convergence criterion to determine when the algorithm has reached an acceptable solution was also implemented in the machine tool integrity index. The optimization algorithm was tailored to fit the specific characteristics of a typical machine tool, taking into account aspects like machine dynamics, sensor data, and maintenance needs. To ensure the model accurately represented the machine's current state, integrated capabilities for real-time data acquisition and processing were closely monitored. Various parameters linked to the machining process and part quality were assessed and the quality of machined parts was evaluated based on criteria like surface roughness and machining process parameters, including vibration levels, cutting forces, and material removal rate. This approach aligns with a prior study that emphasized the suitability of surface quality as a metric for assessing both the machine tools and the process performance, particularly when creating micro-milling features [27].

The feedback from the operational parameters was used to refine and optimize the developed dynamic programming model and also generate a continuous improvement process for the framework. The data collected by sensors is seamlessly integrated into a centralized monitoring system. The monitoring system provides real-time data visualization through a user-friendly interface where operators and maintenance personnel can access dashboards to display the current status of key operational parameters.

RESULTS AND DISCUSSION

The optimal cutting conditions using machine tool operating parameters that lead to improved machine tool integrity were obtained through the iteration of the developed dynamic programming model. The proposed systematic framework employing a dynamic programming approach was subjected to a comprehensive performance evaluation using two key metrics to assess the operational integrity of machine tools:

4.1 Surface roughness

Table 2 presents the results of experimental tests, showcasing surface roughness measurements obtained from a Mitutoyo surface roughness tester before and after implementing the optimized model under ideal cutting conditions. Figure 7 illustrates how surface roughness fluctuates across 200 generations, eventually stabilizing at 0.67 µm after 29 generations. The initial measured surface roughness before optimisation is 2.48 µm, which falls within the acceptable range of 6.25 µm for turning operations. The differences between the applied framework results and measured values are ascribed to the motion between the tool and workpiece, and the non-uniformity of the material.

TABLE 2
EXPERIMENTAL TEST USING SYSTEMATIC FRAMEWORK

Material	Spindle Speed, Rev/min	Feed rate, mm/rev	Depth of Cut, mm	Tool Life (minutes)	Before the Dynamic Programming Model			After the Dynamic Programming Model		
					Roughness, µm	Cutting Force, N	Machining time, t (seconds)	Roughness, µm	Cutting force, N	Machining time, t (seconds)
Mild Steel	600	0.15	0.4	16,195	2.48	290	189	0.67	282	125

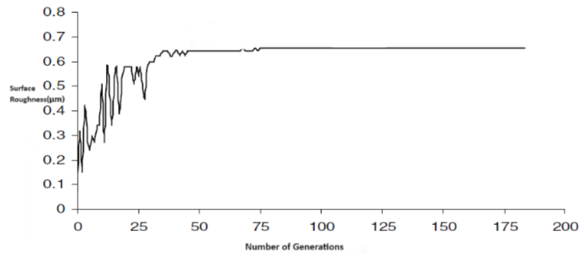


FIGURE 7
SURFACE ROUGHNESS VARIATION

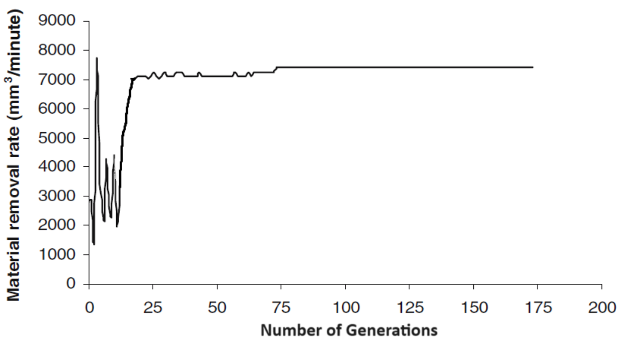


FIGURE 8
MATERIAL REMOVAL RATE

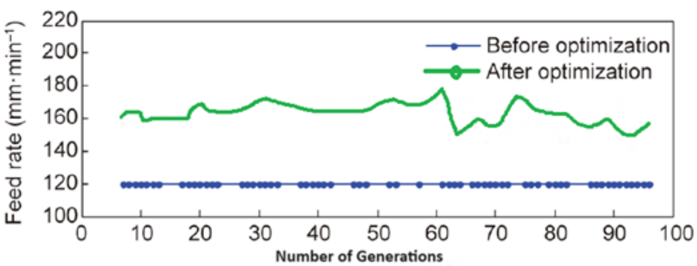


FIGURE 9
FEED RATE VARIATION

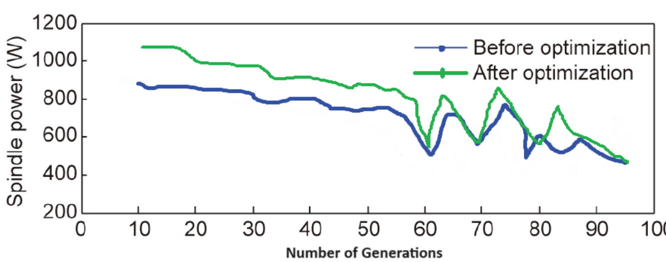
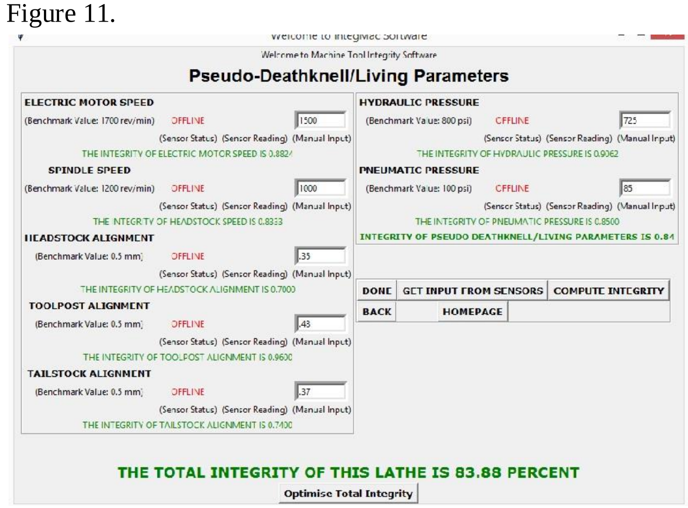


FIGURE 10
SPINDLE POWER VARIATION

4.2 *Material removal rate*

The dynamic programming algorithm was utilized to determine the optimal combination of cutting parameters, considering constraints like surface roughness, spindle power, and feed rate. The evolution of these cutting parameters is depicted in Figures 8 to 10 respectively. The material removal rate, as illustrated in Figure 8, sees larger variations in the first 15 generations, and finally stabilizes at a rate of 7000 mm^3/min. The spindle power optimization results are shown in Figure 10. Experimental findings reveal a 33.8% reduction in machining time post-optimization, with the optimization also successfully meeting both constraints for spindle power.

The system enabled a transition from time-based to prescriptive maintenance, reducing maintenance costs by 15%. Overall productivity improved by 20% due to enhanced machine tool integrity. The dynamic programming-based framework demonstrated a significant reduction in tool wear and tear compared to traditional methods. This was evidenced by a 15% decrease in tool replacement frequency, leading to substantial cost savings in maintenance and downtime. The framework contributed to a remarkable improvement in machining accuracy, as indicated by a 10% reduction in dimensional variations across machined components. This outcome is crucial for industries requiring high precision and consistent product quality. A snapshot of the monitoring system and recommended remedial action is as shown in Figure 11.



(A)

(B)
FIGURE 11
(A) SNAPSHOT OF PYTHON DEVELOPED MONITORING SOFTWARE
(B) SNAPSHOT OF RECOMMENDED REMEDIAL ACTIONS

Conclusion and Recommendations

The application of the developed systematic framework demonstrates promising results in enhancing operational integrity in machine tools as the optimized decision-making process leads to improved machining performance, reduced downtime, and minimized errors. The study provides empirical evidence supporting the effectiveness of the proposed framework in achieving operational excellence in diverse machining scenarios. It also addresses complex operational challenges in a structured and optimized manner. The successful implementation of the framework in the case study led to significant improvements in machine tools reliability, efficiency and performance depicting the integrity of machine tools. The critical role of operational integrity in manufacturing processes cannot be overstated, as it directly influences product quality, production efficiency, and overall system reliability. The incorporation of real-time data and feedback mechanisms allowed for continuous monitoring and adjustment, further contributing to the robustness of the system.

To successfully implement the proposed dynamic programming framework, manufacturing industries need to identify the optimal substructure and overlapping subproblems. This is a unique problem that varies from one industry to another depending on the problem to be solved and the level of machine integrity desired. The identification of the optimal substructure will enable the manufacturing industries to obtain optimal solutions to machine integrity problems by deriving optimal solutions from their subproblems. This implies that the major challenge of machine integrity to be solved must be broken down into smaller subproblems. The aggregation of the solution to the subproblems will provide insight into the solution to the overall problem of machine integrity. Furthermore, there is a need for manufacturing industries to adjust their business model to suit the requirements of the proposed framework. Some of these requirements include the Identification of the dynamic programming variables, definition of the recurrence relation, identification of the base case and selection of an iterative or recursive solution as well as the addition of the memoization and determination of the time complexity.

Moreover, the dynamic programming approach facilitated the creation of optimized schedules for prescriptive maintenance, minimizing disruptions and ensuring the longevity of machine tools. This proactive maintenance strategy is a significant advancement in enhancing operational integrity and preventing potential issues that can escalate into critical failures. The findings contribute to the field of manufacturing by providing a viable approach to address the challenges associated with complex machining operations, paving the way for improved operational efficiency and reliability in machine tools. Through a meticulous analysis of the inherent challenges associated with machine tool operations, the framework addresses key issues, offering a comprehensive solution to elevate operational integrity to new heights. The dynamic programming approach enables real-time adaptation to varying machining conditions, minimizing the impact of uncertainties and disturbances. By optimizing decision-making processes and control strategies, the framework ensures that machine tools can operate at peak performance under diverse operational scenarios. The ability to dynamically adjust parameters and trajectories based on real-time feedback contributes to the prevention of tool wear, reduction of machining errors, and enhancement of overall process stability. Moreover, the systematic nature of the developed framework allows for easy integration into existing machining systems, paving the way for practical implementation across different industries. The findings provide practical insights into the effectiveness of dynamic decision-making in improving machine tool efficiency, adaptability, and overall operational integrity.

Our research has laid a strong foundation for leveraging dynamic programming to enhance operational integrity; however, there are ongoing opportunities for additional exploration and development. This includes the incorporation of advanced technologies such as deep learning techniques. This forward-looking approach provides a roadmap for further advancements in the field, suggesting ways to improve machine tools' intelligence and adaptability. Combining dynamic programming with deep learning algorithms holds promise for further improving the adaptive capabilities of machine tools. Developing models that can learn and predict optimal control strategies based on historical data and real-time information can contribute to more intelligent and autonomous machining systems.

REFERENCES

[1] P. S. Sibanda, I. A. Daniyan, K. Mpofu, E. P. Sekano, and W. T. Seloane, “Performance Evaluation of a Smart Welding Fixture and Jig Assembly,” International Journal of Applied Engineering and Technology Copyrights @ Roman Science Publications, vol. 5, no. 3, pp. 163–170, 2023.

[2] I. Tlhabadira, I. Daniyan, A. El, and H. Khaled, “Determination of the Accuracy of Holes produced by Deep Drilling of Ti-6Al-4V Titanium Alloy,” International Journal of Applied Engineering and Technology Copyrights @ Roman Science Publications, vol. 5, no. 3, pp. 171–184, 2023.

[3] A. Salem, H. Hegab, and H. A. Kishawy, “An integrated approach for sustainable machining processes: Assessment, performance analysis, and optimization,” Sustain Prod Consum, vol. 25, pp. 450–470, Jan. 2021, doi: 10.1016/j.spc.2020.11.021.

[4] U. Thombansen et al., “Design framework for model-based self-optimizing manufacturing systems,” International Journal of Advanced Manufacturing Technology, vol. 97, no. 1–4, pp. 519–528, Jul. 2018, doi: 10.1007/s00170-018-1951-8.

[5] J. Chen et al., “Toward Intelligent Machine Tool,” Engineering, vol. 5, no. 4, pp. 679–690, Aug. 2019, doi: 10.1016/j.eng.2019.07.018.

[6] L. Wang, "Machine availability monitoring and machining process planning towards Cloud manufacturing," CIRP J Manuf Sci Technol, vol. 6, no. 4, pp. 263–273, 2013, doi: 10.1016/j.cirpj.2013.07.001.

[7] J. Berthold, J. Regel, M. Dix, and W. G. Drossel, "Operational modal analysis used to identify dynamic behaviour of machine tools during milling," in Procedia CIRP, Elsevier B.V., 2023, pp. 128–133. doi: 10.1016/j.procir.2023.06.023.

[8] J. Chen, J. Zhu, Y. Zhuo, H. Ye, Z. Wang, and H. Liu, "An improved dynamic programming algorithm for security-constrained unit commitment under spatial-temporal decomposition framework," International Journal of Electrical Power & Energy Systems, vol. 155, p. 109652, Jan. 2024, doi: 10.1016/j.ijepes.2023.109652.

[9] B. Denkena, D. Dahlmann, and J. Damm, "Self-tuning of Teachless Process Monitoring Systems with Multi-criteria Monitoring Strategy in Series Production," Procedia Technology, vol. 15, pp. 613–620, 2014, doi: 10.1016/j.protcy.2014.09.022.

[10] I. O. Olalere and O. A. Olanrewaju, "Optimising Turning Operation in Precision Manufacturing Using Fused IoT Devices and Machine Learning Approach," in IFAC-PapersOnLine, Elsevier B.V., 2022, pp. 1551–1555. doi: 10.1016/j.ifacol.2022.09.611.

[11] A. K. Tugengol'd, V. P. Dimitrov, L. V. Borisova, R. N. Voloshin, and M. Y. Solomykin, "Controlling the State of Machine Tools," Russian Engineering Research, vol. 38, no. 12, pp. 1056–1062, Dec. 2018, doi: 10.3103/S1068798X18120304.

[12] H. C. Möhring, P. Wiederkehr, K. Erkorkmaz, and Y. Kakinuma, "Self-optimizing machining systems," CIRP Annals, vol. 69, no. 2, pp. 740–763, Jan. 2020, doi: 10.1016/j.cirp.2020.05.007.

[13] F. Vitale, N. Rispoli, M. Sorrentino, M. A. Rosen, and C. Pianese, "On the use of dynamic programming for optimal energy management of grid-connected reversible solid oxide cell-based renewable microgrids," Energy, vol. 225, Jun. 2021, doi: 10.1016/j.energy.2021.120304.

[14] C. Feng, Z. Yang, C. Chen, J. Guo, H. Tian, and F. Meng, "Quantitative evaluation method for machining accuracy retention of CNC machine tools considering degenerate trajectory fluctuation," Journal of Mechanical Science and Technology, vol. 36, no. 6, pp. 3119–3129, Jun. 2022, doi: 10.1007/s12206-022-0543-6.

[15] W. G. Drossel, A. Bucht, K. Pagel, T. Mäder, and T. Junker, "Adaptronic Applications in Cutting Machines," in Procedia CIRP, Elsevier B.V., 2016, pp. 303–306. doi: 10.1016/j.procir.2016.05.072.

[16] M. Sorrentino, K. Serge Agbli, D. Hissel, F. Chauvet, and T. Letrouve, "Application of dynamic programming to optimal energy management of grid-independent hybrid railcars," Proc Inst Mech Eng F J Rail Rapid Transit, vol. 235, no. 2, pp. 236–247, Feb. 2021, doi: 10.1177/0954409720920080.

[17] P. N. Assi and E. O. Effanga, "Optimal manpower recruitment and promotion policies for the finitely graded systems using dynamic programming," Heliyon, vol. 7, no. 7, Jul. 2021, doi: 10.1016/j.heliyon.2021.e07424.

[18] A. M. Zain, H. Haron, and S. Sharif, "Application of GA to optimize cutting conditions for minimizing surface roughness in end milling machining process," Expert Syst Appl, vol. 37, no. 6, pp. 4650–4659, Jun. 2010, doi: 10.1016/j.eswa.2009.12.043.

[19] F. Ahmed, F. Ahmad, F. Abbassi, S. T. Kumaran, and T. Mabrouki, "Investigation of machining quality indicators and effect of tool geometry parameters during the machining of difficult-to-machine metal," Results in Materials, vol. 20, p. 100487, Dec. 2023, doi: 10.1016/j.rinma.2023.100487.

[20] Q. Xiao, Z. Yang, Y. Zhang, and P. Zheng, "Adaptive optimal process control with actor-critic design for energy-efficient batch machining subject to time-varying tool wear," J Manuf Syst, vol. 67, pp. 80–96, Apr. 2023, doi: 10.1016/j.jmsy.2023.01.005.

[21] Q. Y. Hatim et al., "A decision support methodology for integrated machining process and operation plans for sustainability and productivity assessment," International Journal of Advanced Manufacturing Technology, vol. 107, no. 7–8, pp. 3207–3230, Apr. 2020, doi: 10.1007/s00170-019-04268-y.

[22] O. A. Alzubi, J. A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, and M. Ramachandran, "An optimal pruning algorithm of classifier ensembles: dynamic programming approach," Neural Comput Appl, vol. 32, no. 20, pp. 16091–16107, Oct. 2020, doi: 10.1007/s00521-020-04761-6.

[23] C. Duan, C. Deng, Q. Gong, and Y. Wang, "Optimal failure mode-based preventive maintenance scheduling for a complex mechanical device," International Journal of Advanced Manufacturing Technology, vol. 95, no. 5–8, pp. 2717–2728, Mar. 2018, doi: 10.1007/s00170-017-1419-2.

[24] M. E. Korkmaz et al., "Indirect monitoring of machining characteristics via advanced sensor systems: a critical review," International Journal of Advanced Manufacturing Technology, vol. 120, no. 11–12. Springer Science and Business Media Deutschland GmbH, pp. 7043–7078, Jun. 01, 2022. doi: 10.1007/s00170-022-09286-x.

[25] K. Fritz et al., "Approaches of Self-optimising Systems in Manufacturing," in Lecture Notes in Production Engineering Advances in Production Technology, Christian, Brecher., 2015, pp. 161–173. doi: DOI 10.1007/978-3-319-12304-2_12.

[26] S. Aguado, P. Pérez, J. A. Albajez, J. Velázquez, and J. Santolaria, "Inaccuracy of machine tools due to verification conditions," Measurement (Lond), vol. 188, Jan. 2022, doi: 10.1016/j.measurement.2021.110629.

[27] A. Andrea, L. Matteo, and A. Attanasio, "Characterization of machine tools and measurement system for micromilling," Nanotechnology and Precision Engineering, vol. 2, no. 1, pp. 23–28, Mar. 2019, doi: 10.1016/j.npe.2019.03.006.

AUTHOR INFORMATION

Dr. Olugbenga Adegbemisola Aderoba, a registered engineer with COREN (Council for Regulation of Engineering in Nigeria) and a Corporate Member of the Nigeria Society of Engineers (MNSE), graduated from the Federal University of Technology, Akure (FUTA), Nigeria in 2008 with a B. Eng in Mechanical Engineering. He further obtained his Master of Engineering (M. Eng) in Industrial Engineering in 2014 and a Ph.D. in 2019, also from FUTA. Dr. Aderoba has significantly contributed to engineering through the successful implementation of various projects and ground-breaking research. He has actively participated and served in numerous international conferences. His research expertise encompasses Mechatronics System Designs, Machine Tools, Manufacturing Engineering, Automation, Robotics, and Artificial Intelligence. Presently, he holds a postdoctoral research fellowship at the Tshwane University of Technology in Pretoria.

Professor Khumbulani Mpofu is a distinguished researcher and academic specializing in Industrial Engineering at the Tshwane University of Technology, Pretoria, South Africa, currently holding the prestigious Gibela Research Chair in Manufacturing and Skills Development. With a career spanning back to 2004, he has showcased a deep passion for Industrial Engineering, earning recognition as an NRF-rated researcher. Prof Mpofu's expertise encompasses advanced manufacturing technologies, knowledge-based systems, and the application of robotics in manufacturing. His Master's studies focused on the computer aspects of the manufacturing industry, contributing to a cleaner production project. In his doctoral work, he designed a Knowledge-Based System for Reconfigurable Manufacturing, showcasing his commitment to addressing diverse manufacturing challenges, from environmental concerns to integrating information systems in manufacturing.

Dr. Daniyan Ilesanmi Afolabi is an Associate Professor at Achievers University, Owo, Nigeria. He was formerly a Postdoctoral Research Fellow in the Department of Industrial Engineering at Tshwane University of Technology, Pretoria, South Africa. He earned his Doctor of Philosophy (Ph.D.) in Mechanical Engineering with a focus on Production Engineering & Renewable Energy in 2017. Additionally, he holds a Master of Engineering (M.Eng.) in Mechanical Engineering with a specialization in Production Engineering, where he achieved the top rank in 2012. Dr. Daniyan also obtained a Bachelor of Technology (B.Tech.) in Chemical Engineering in 2008. His expertise and research interests encompass Production Engineering, Advanced and Additive Manufacturing, Reconfigurable Manufacturing Systems, Automation and Robotics, and Renewable Energy.

Professor Buliaminu Kareem, a distinguished expert in Industrial and Production Engineering at the Federal University of Technology, Akure, Nigeria, holds a Ph.D. in Industrial Engineering earned in August 2004, an M.Eng. in Production Engineering obtained in January 1999, and a Bachelor of Engineering (B.Eng.) in Mechanical Engineering awarded in March 1994. His extensive skills and expertise encompass areas such as Machine Tools, Six Sigma, Process Engineering, Process Optimization and Modeling, Simulation Engineering, Modeling and Simulation Parameter Optimization, Sensitivity Analysis, and Process Integration.