applied sciences

MDPI

Article

A Novel Approach for Mining Machining Process Decision Knowledge Based on Knowledge Constraint Combined with Water Wave Optimization Algorithm

Xinzheng Xu, Zhicheng Huang *, Lihong Qiao, Yongqiang Wan, Chao Chen and Zhujia Li

School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China; secret332@163.com (X.X.); lhqiao@buaa.edu.cn (L.Q.); 18375248@buaa.edu.cn (Y.W.); chao_chen@buaa.edu.cn (C.C.); jackiekingli@163.com (Z.L.)
* Correspondence: zc_huang@buaa.edu.cn

Abstract

Knowledge discovery constitutes a vital component in building intelligent CAPP systems, and the effective discovery of process knowledge has become a prominent research focus within intelligent manufacturing. Process decision knowledge is a type of knowledge that reflects the relations between process data items, represented in the form of production rules. However, PDK discovery faces low accuracy challenges from complex high-dimensional manufacturing data and implicit experience-dependent process decisions. This paper proposed a PDK mining framework that combines knowledge constraint and the water wave optimization algorithm. This approach formulated prior knowledge mathematically using an association discriminant matrix and embedded this representation into the knowledge mining model, thus equipping the algorithmic framework with the ability to discover PDK accurately. The WWO is utilized to search within the sample space for combinations of process data items that constitute valid knowledge. In contrast to traditional association rule mining algorithms that lack accuracy and template-based methods that are inherently rigid, the proposed approach provides a robust solution by achieving over 90% correctness in PDK mining. It also serves as a demonstration and offers insights for mining similar rule-based knowledge in other fields.

Keywords: machining process planning; knowledge discovery; knowledge constraint; association rule mining; water wave optimization

Check for updates

Academic Editors: Nadia Nedjah and Luiza de Macedo Mourelle

Received: 10 February 2026
Revised: 3 March 2026
Accepted: 7 March 2026
Published: 14 March 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

1.1. Background

Computer-Aided Process Planning (CAPP) systems are expert systems that facilitate the process planning of products or components through technologies such as numerical computation, logical judgment, and reasoning [1]. They effectively assist process planners in completing process planning while ensuring product quality. In recent years, traditional CAPP systems (e.g., variant and generative types) have been evolving towards intelligent CAPP systems driven and enabled by artificial intelligence (AI) technologies [2–4]. By integrating technologies such as machine learning, fuzzy analysis and modern optimization algorithms, intelligent CAPP systems possess capabilities for adaptive knowledge acquisition, intelligent knowledge delivery and reasoning, as illustrated in Figure 1. Currently, practical and mature intelligent CAPP systems are still scarce in the manufacturing

field, and related technologies remain largely in the stage of theoretical exploration and validation [3–5]. In intelligent CAPP systems, the knowledge discovery module serves as a critical component. It is responsible for the continuous and automatic discovery and extraction of process knowledge from the production environment, as well as the timely updating of the process knowledge base [6]. Nevertheless, due to factors such as the heterogeneity of data sources, and the broad scope of knowledge content, the implementation of the knowledge discovery module faces numerous technical challenges, significantly constraining its efficiency and accuracy [7]. Knowledge discovery remains one of the major technical bottlenecks in constructing intelligent CAPP systems [8]. Therefore, in-depth research on process knowledge discovery is of great significance for the realization and development of intelligent CAPP systems.

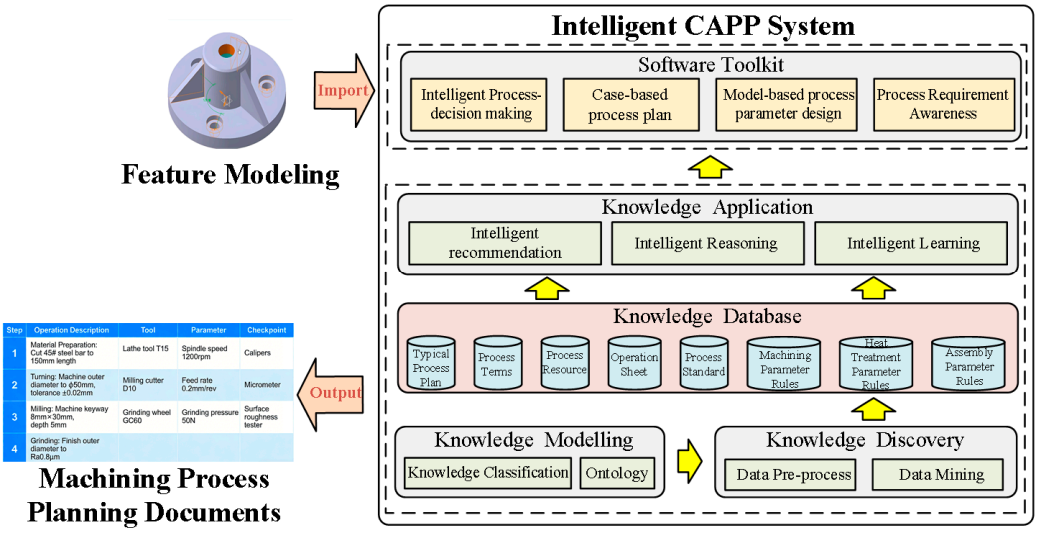


Figure 1. The architecture of intelligent CAPP.

Process decision knowledge (PDK) is a commonly used type of process knowledge. It reflects relations among process information data items and is typically represented in the form of production rules [9]. This category of knowledge is highly reusable and can assist process planners in making decisions such as tooling selection, process parameter selection, and machining method selection, thereby facilitating the process planning [10]. Process planning documents are the carriers of process routes and operational methods for products or components. They are central to production planning, organization, and control, and are crucial for ensuring product quality [11]. In manufacturing activities, countless process planning documents are accumulated, which are then used to assist process planning for new parts. These documents carry data related to component parts, machining methods, equipment, cutters and other process details. Within these data, rich PDK applied in process planning is contained [12]. PDK discovery from process planning document data is an effective approach for manufacturing enterprises to automatically acquire and accumulate knowledge resources, and it has become one of the research hotspots in knowledge discovery within the field of intelligent manufacturing.

In order to address the issue of low-accuracy knowledge discovery associated with traditional association rule mining, this paper proposes a PDK mining method that integrates knowledge constraint with the water wave optimization (WWO) algorithm for the purpose of achieving effective process knowledge discovery in intelligent CAPP systems. The method incorporates prior knowledge from the process domain by establishing a matrix that describes the pairwise semantic relations among process data items. Based on this matrix, a fitness function is designed to identify whether a rule-based combination

of data items can constitute correct PDK. Leveraging the distribution characteristics of PDK within the discrete process data spaces, water wave operators and a corresponding algorithm for combinatorial search are established. With the propagation of water waves driving the algorithm, PDK is searched and extracted from the sample space. During this process, combinations of data items whose fitness values exceed a predefined threshold are retained as the final results of PDK mining.

1.2. Related Work

1.2.1. Association Discrimination in Association Rule Mining for Knowledge Discovery

The PDK mining falls within the domain of rule-based knowledge discovery [13]. Knowledge discovery based on association rule mining is the process of extracting meaningful or application-oriented rules from structured data (e.g., two-dimensional tables) or unstructured data (e.g., text, images). This is a combinatorial search problem over discrete data items [14,15]. Common association rule mining methods include recursive search, heuristic search, and instance-based learning [16]. The mining models of these methods primarily consist of two key elements: (1) association discrimination metrics, and (2) search strategies for rule-form data item combinations. Association discrimination metrics are parameters that measure the correctness of knowledge. They are used to determine whether a rule-form data item combination can be served as valid knowledge, and directly influence the quality of mining results.

Support and confidence are typical association discrimination metrics that reflect the statistical level of rules in data samples. Rules with high statistical levels are defined as strong association rules [16]. Classical association rule mining algorithms, such as Apriori, FP-Growth, and Eclat [17–19], utilize support and confidence as the basis for strong rule mining. Kamsu-Foguem et al. [20] developed a rule mining method based on the metrics of support and confidence. It analyzed data from machine tool maintenance processes to mine rule-form decision knowledge for guiding inspectors in equipment fault diagnosis. However, the study did not specify the number or proportion of correct knowledge in the mining results, and the impact of the two metrics on the quality of knowledge mining were not further analyzed. Garcia et al. [21] argued that domain knowledge possesses inherent logic and features for a particular discipline. They maintain that relying solely on support and confidence to judge whether data items constitute domain knowledge can retain statistically significant but irrelevant items, resulting in a large number of invalid domain knowledge. Therefore, association discrimination metrics based on engineers' demand information were established, and a user demand-oriented rule mining method was proposed to reduce the generation of invalid rules. In this approach, Knowledge Cohesion (KC), the association metric, was used to assess rule validity from two perspectives: design purpose and user demands. On petroleum processing data, the method outperformed the Apriori benchmark, demonstrating that explicit demand integration in the association metric can yield more valid rules. Chemchem et al. [22] proposed a rule mining method based on cluster analysis to mine the decision rules of the transportation domain. By analyzing the characteristics of decision rules in this domain, a computational model based on word vectors to quantify the similarity between pairs of rule contents was constructed. Driven by an improved hybrid K-means and K-NN clustering algorithm, the method started with commonly used decision rules in business and recursively derived more useful rules in the decision rule space through backward reasoning. Simulation experiments validated the mined rules, demonstrating their effectiveness in addressing decision-making problems such as route planning, transport mode selection, and vehicle scheduling.

In conclusion, association discrimination metrics based on statistical levels are inadequate for determining whether different data items can constitute the content of domain

knowledge in rule form, and they fail to ensure the correctness of the mined knowledge. Therefore, for PDK mining, it is essential to fully incorporate the characteristics of the mining target when constructing association discrimination metrics.

1.2.2. Combinatorial Search Strategy for Rule-Based Knowledge Mining

Combinatorial search strategies for rule-form data item combinations refer to the methods for generating such combinations during mining, used to explore individuals in discrete data spaces that meet the threshold conditions of association discrimination metrics. These strategies reflect the mechanisms for expanding the state space or problem space by the mining algorithm [23]. The design of search strategies should align with the features of the problem model, as they directly impact the operational efficiency of the mining algorithm and the amount of knowledge in the mining results. He et al. [24] pointed out that traditional association rule mining algorithms suffer from low efficiency and high memory usage when applied to large-scale rule mining. In the proposed method, the search process can be guided toward the mining target by incorporating certain constraints of the problem into the control information of search paths. Studies on heuristic-based rule mining have demonstrated that incorporating heuristic search strategies alleviates the burden in computing association discrimination, and this can effectively decrease algorithm runtime and improve mining efficiency. Chen L et al. [25] proposed a rule mining method based on reinforcement learning, which leverages the adaptive characteristics of reinforcement learning to construct search strategies for rules through interactive trial and error with data samples. This strategy recursively generates rules that meet threshold conditions in the sample space. However, when the data sample size is large, the state space of reinforcement learning expands. This may increase the difficulty of learning rule search strategies and affects the performance of the mining algorithm.

Currently, rule-based knowledge mining has achieved certain research results in the manufacturing field, primarily focused on fault diagnosis and production scheduling [26–28], but rarely in process planning. Schuh et al. [29] proposed a PDK mining method based on decision trees (DT-PDKM). This method employs information gain to construct a decision tree from historical process planning data. This information tree encapsulates rule-form PDK in an intuitive, easily understandable and interpretable manner. In this method, association discrimination metric was established based on the information gain for rules. A process planning feature that exhibits higher information gain is assigned greater weight in the decision-making process, and corresponding data instances are thus preferentially selected as components of decision rules [30]. The method computes information gain based on the entropy of features (the proportion of each feature instance in the data sample), which essentially belongs to probabilistic analysis for knowledge discovery. Consequently, during the mining process, irrelevant features may be retained as nodes in the decision tree, leading to inaccurate process rules and affecting the correct application of the decision tree in subsequent process planning. Moreover, decision tree-based mining algorithms partition data samples in detail according to each feature, leading to a risk of overfitting in the knowledge mining model. A key limitation is the poor generalization ability of decision tree-based models. This is particularly evident when mining process data from different products, which severely hinders its broad applicability [31]. Additionally, the method employed a greedy algorithm to search for feature instances. This search strategy considers only the feature with the optimal gain value at the time of adding tree nodes and ignores potential feature combination effects. As a result, the algorithm tends to converge to local optima during combinatorial search, and the resulting decision tree structure fails to fully exploit other potentially relevant features in the sample space. This thus limits its utility in process planning. In the context of PDK, a wide range of aspects are encompassed, including materials, heat treatment, cutting parameters,

cutters, machine tools, and fixtures. These elements are interconnected and influence each other, possessing inherent structure and logic [32].

Water wave optimization (WWO) is a novel nature-inspired heuristic algorithm based on shallow water wave theory, primarily designed to solve optimization problems in continuous numerical spaces [33,34]. Recent studies have confirmed that WWO effectively addresses discrete optimization problems, outperforming general heuristics like GA and PSO in applications such as workshop scheduling [35] and path planning [36]. This strength of WWO is largely attributed to WWO's distinctive search mechanisms, including wave propagation and refraction. These mechanisms facilitate comprehensive exploration of the search space while mitigating the risk of local optima entrapment, enabling WWO to be well-adapted for complex discrete combinatorial searches. Consequently, it presents a promising new approach for PDK mining.

In conclusion, discrimination metrics and search strategies are both the key to effectively achieve knowledge discovery. Therefore, this paper proposes a novel knowledge mining framework designed for PDK mining, comprising an association discrimination model and a rule search strategy.

1.2.3. Knowledge Constraint to Inform Domain Knowledge Discovery

A common limitation of generic data-driven models is their failure to leverage accumulated prior information or domain knowledge, leading to a lack of domain's operational "common sense". In knowledge discovery, the lack of domain knowledge can cause models to yield unreasonable outcomes. To address this issue, the approaches integrating domain knowledge to guide the knowledge discovery were put forward, which are also known as knowledge-driven paradigms. It transforms prior knowledge into structured representations (e.g., vectors, matrices, equations) and integrates them into the learning framework [37]. This is achieved by converting high-dimensional, discrete knowledge into a low-dimensional vector space, which in turn offers a computable representation that constrains and informs the algorithm, enabling efficient domain knowledge discovery [38].

Currently, knowledge-driven techniques have been widely adopted in neural network-based knowledge discovery. Depending on how the knowledge drives the knowledge discovery process, these approaches are commonly grouped into soft constraints and hard constraints. Physics-Informed Neural Networks (PINNs) represent a typical form of knowledge-driven modeling under soft constraints [39]. A defining feature of PINNs is the composite loss function, where the prediction error is augmented with a physics-constrained regularization term derived from governing equations, in contrast to conventional neural networks [40]. This explicitly embeds domain knowledge as prior information into the learning process. During the learning process, PINNs simultaneously optimize both data error and physical consistency error, ensuring that the network outputs not only fit the training data but also adhere to physical laws in an average sense. This improves model accuracy and reliability, and better supports expert decision-making [41]. By integrating physics, PINNs retains high predictive accuracy with sparse data, and can effectively reveal the underlying physical mechanisms or novel patterns in the data sample.

However, due to the limitations of penalty-based methods, the inclusion of regularization terms into the loss function does not ensure that the model's predictions strictly satisfy physical constraints at every sample data point [42]. To further improve the generalization of the model, Mohan et al. [43] proposed a hard-constrained embedding approach using convolutional neural networks (CNNs) in 2020. In this method, a physics-embedded decoder was developed as the core mechanism. It enables the network to predict unlabeled intermediate variables and maps them to the final physically consistent output through CNN kernels that encode differential operators. Similarly, Chen et al. [44] developed a

deep learning model based on hard-constrained knowledge embedding. In the approach, domain knowledge (e.g., governing equations) is discretized into projection matrices, and a physics-consistent value space for predictions is constructed. This allows unlabeled coordinate data and domain knowledge to jointly inform model learning. Experimental results indicate that the proposed physics-informed HCP model outperforms fully connected networks and soft-constrained models in terms of learning efficiency, predictive accuracy and robustness to observation noise.

Collectively, these studies demonstrate that integrating domain knowledge into machine learning models improves both robustness and prediction accuracy. Over the decades, a substantial degree of a priori knowledge has been accumulated in the domain of process planning. However, such knowledge remains underutilized for PDK mining. Therefore, integrating such prior information to construct a priori knowledge-informed machine learning models can effectively enable the discovery of PDK and support the development of intelligent CAPP systems.

2. PDK Mining Problem and Method Framework

2.1. Process Decision Knowledge Mining Problem

2.1.1. Fundamental Concepts

Semantics refers to the specific meaning or interpretation assigned to data within a domain. Knowledge incorporates the semantic information of data. Semantic relations within data reflect the intrinsic connections in domain knowledge. To define the semantic elements involved in the PDK mining problem, the concept of process features is given. A process feature is an attribute used to describe and distinguish different process plans. It comprises a set of instances that share the same functional behavior. For example, machining method is one process feature, and its instance set includes rough turning, semi-finish turning, finish turning and so on. A process plan can be described by multiple process features, and each process feature is instantiated by multiple values. Common process features and their instance values in process planning are listed in Table 1. Process features embody the semantics of process data items, while process data items are instance values of process features.

Table 1. Common process features and their instances.

Process Property	Instances
Manufacturing Feature	Through Holes, Grooves, Planar Surfaces, End Faces, etc.
Machining Method	Rough Turning, Semi-Finish Turning, Finish Turning, Rough Boring, Finish Boring, etc.
Machine Tool Equipment	C336-1, C368L, C385L, C512A, etc.
Measuring Tool	Internal Micrometers, Cylindrical Plug Gauges, Vernier Calipers, etc.
Fixture	Self-Centering Chucks, Mandrels, Centers, etc.
.....	.....

PDK represents the summarization and refinement of practical production experience by process planners. It is composed of process data items, their semantics, and the relations among these data items, typically represented in the form of production rules. It can be defined as follows:

$$if\left\{y_{x_i}:x_i\ldots y_{x_o}:x_o\ldots y_{x_p}:x_p\right\}then\left\{y_{x_j}:x_j\ldots y_{x_m}:x_m\ldots y_{x_n}:x_n\right\},i,o,p,j,m,n\in N^+ \quad (1)$$

where $if\left\{y_{x_i}:x_i\ldots y_{x_o}:x_o\ldots y_{x_p}:x_p\right\}$ and $then\left\{y_{x_j}:x_j\ldots y_{x_m}:x_m\ldots y_{x_n}:x_n\right\}$ represent the condition part (rule antecedents) and the result part (rule consequents) of PDK, respectively. Both are composed of multiple indivisible, distinct data units with independent semantics. In Equation (1), x_i denotes a process data item within the condition part, and y_{x_i} is the semantic label of x_i, describing the specific meaning of this data item in the domain of process planning. For example, a piece of PDK—"*YT15* carbide cutter can be used for rough turning of *45-steel* component parts"—can be expressed according to Equation (1) as follows: If {Machining Method: Rough Turning, Material: 45 Steel} Then {Tool Type: Carbide Tool, Tool Grade: YT15}.

In traditional association rule mining, the focus is on probability-based association rules that capture dependencies among data items. Specifically, these rules express the likelihood of antecedents occurring when their consequents are present. However, such probabilistic associations do not sufficiently indicate whether an underlying relation exists among the data items, and further validation through practical application for rules is required. As time and space change, the statistical significance of such rules is subject to variation, and correspondingly the associative relations among data items may also shift.

Different from the association rules in these studies, as a kind of rule-based knowledge, PDK has been validated through extensive practical application. It embodies intrinsic semantic logic and reflects inherent relations among data items. Such relations are pre-existing and do not change with variations in the objective environment. If the semantics of multiple data items are not related, these data items cannot constitute a correct piece of PDK, which cannot be applied in process planning. For example, in the context of PDK for machining method selection, the selection of machining method is primarily related to machining features, economic accuracy, workpiece material, and surface roughness. Therefore, PDK for machining method selection is mainly composed of process data items from these process features.

2.1.2. The Model of the Process Decision Mining Problem

A process plan is the product of a planning process guided by PDK, subsequently recorded in process planning documents. A process plan consists of multiple operations, and each operation comprises several steps. The data samples for PDK mining include data from multiple process planning documents and can be defined as follows:

$$\mathrm{D} = \{p_1, p_2 \ldots\ p_i \ldots p_n\}, n \in N^+ \tag{2}$$

where D denotes the data sample, and p_n denotes the data from the n-th process document, which can be defined as follows:

$$p_i = \{o_1, o_2 \ldots o_j \ldots o_m\}, m \in N^+ \tag{3}$$

where o_j denotes the data of the j-th operation and can be defined as follows:

$$\mathrm{o}_j = \{s_1, s_2 \ldots s_h \ldots s_k\}, k \in N^+ \tag{4}$$

where s_h denotes the data of the h-th step.

In this paper, only table data (textual and numerical process data) in process planning documents is considered, excluding graphical representations such as 2D operation drawings. The information model of process plan is illustrated in Figure 2. The objective of PDK mining is to extract a series of logically related process data items from data samples, organize them into ordered combinations, and form PDK as defined in Equation (1). The problem model is presented in Figure 3. In this paper, key metrics to evaluate the method

include the quantity of correct rules extracted, the accuracy rate of the results, and the speed of knowledge accumulation during the mining process. This study aims to develop an automated method for PDK discovery, with the purpose of improving both the accuracy and the quantity of correct knowledge extracted, and thereby accelerating knowledge accumulation. This provides CAPP systems with the key capability to directly capture correct PDK from historical process planning documents.

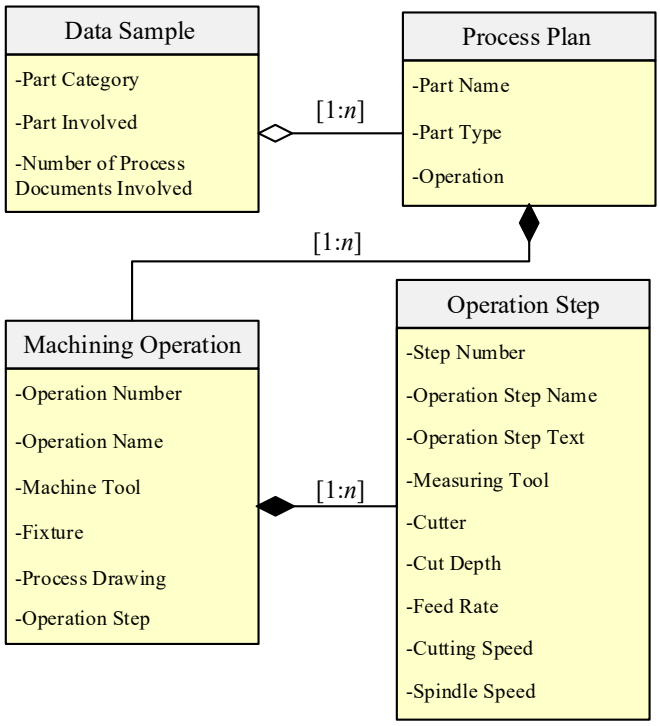


Figure 2. Information Model of Process Plan.

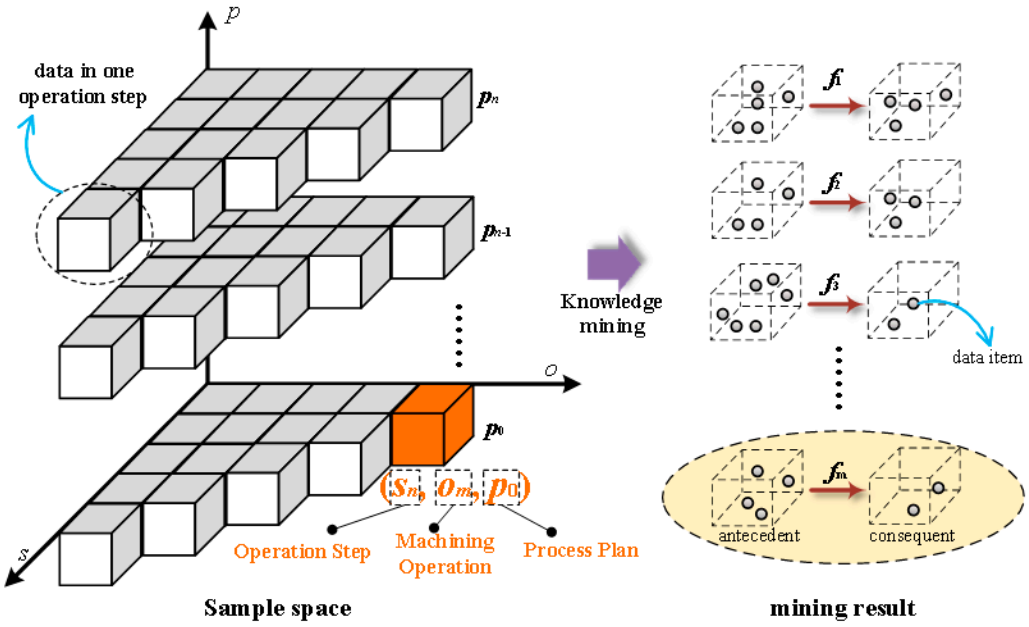


Figure 3. Illustration of the Problem Modeling.

2.2. The Framework of the Proposed Approach

In this paper, PDK mining framework comprises two main components: data sample construction and knowledge mining, as illustrated in Figure 4. The data sample construction

involves technologies such as OCR and named entity recognition (NER) to transform diverse process planning document data into structured data samples in tabular format for subsequent knowledge mining. The knowledge mining primarily includes prior knowledge constraint and the modeling of a mining algorithm based on the WWO, aiming to discover rule-based knowledge from data sample.

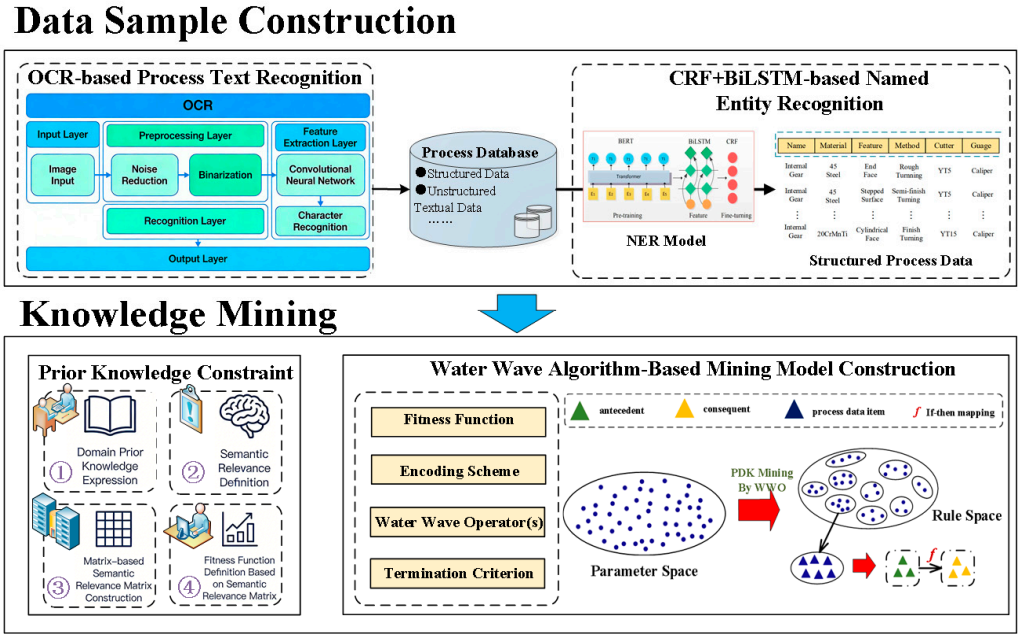


Figure 4. Framework of the method.

(1) Data sample construction

Knowledge mining focuses on process planning document data stored in databases. As the primary carriers of process data, process planning documents are commonly stored in various formats within manufacturing enterprise databases, including electronic files and data stored in two-dimensional table. This diversity in data formats presents challenges for data processing in PDK mining. To address data heterogeneity, conversion into a structured format is required, which improves data readability and usability, thereby facilitating subsequent analysis using the mining algorithm. The process for structuring raw documents is illustrated in Figure 4.

Process planning documents consist of multiple operation cards, which record detailed information about each operation in tabular form, as illustrated in Figure 5. To facilitate computer processing, Optical Character Recognition (OCR) technology is employed to identify and extract information from these operation cards into a database. The data processed by OCR still contains unstructured process information, including natural language descriptions of process steps. To address this, named entity recognition (NER) technology is applied to perform word segmentation and entity type labeling on the natural language text. Following this step, a structured data sample is fully constructed. This structured data is stored in two-dimensional data tables and serves as the data sample for PDK mining, as is illustrated in Figure 6.

Through the structuring process, process planning document data can be transformed into samples composed of data items and their corresponding labels, which are stored in two-dimensional tables. The fully structured data samples can be further represented as a transaction database, defined as follows:

$$T = \{t_1, t_2, \ldots, t_m\}, m \in N^+ \quad (5)$$

where *t* denotes a transaction in the data sample stored in the two-dimensional table. Each transaction corresponds to a record that contains the data content of a process step along with its associated operation. Each record consists of fundamental data units—process data items—where each data item has been entity-labeled and assigned domain-specific semantic meaning, as illustrated in Figure 5.

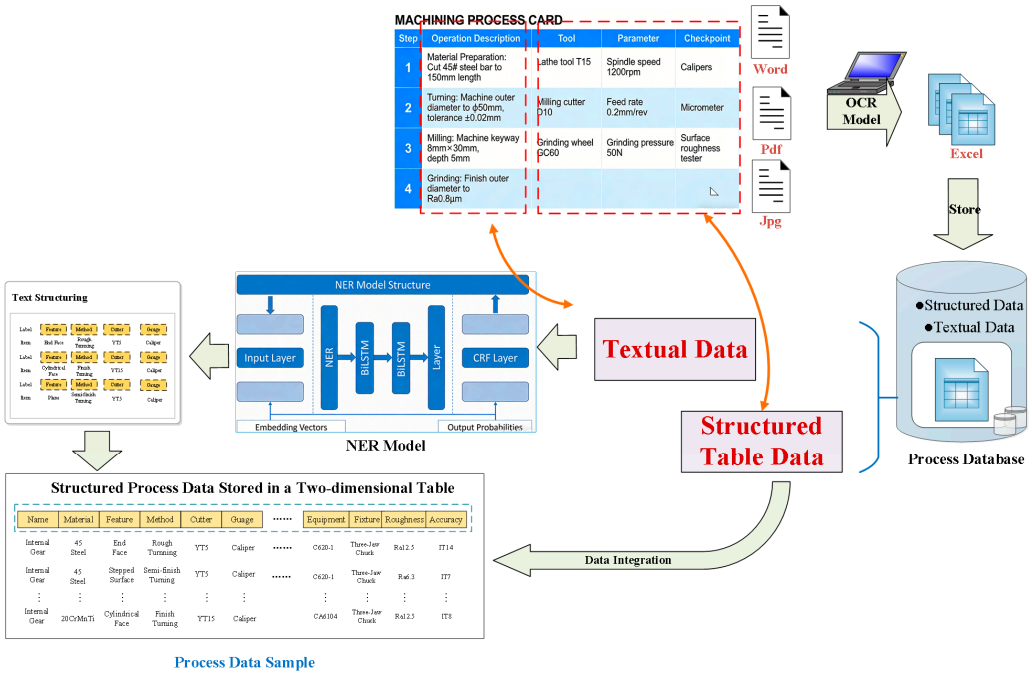


Figure 5. Process Data Sample Construction.

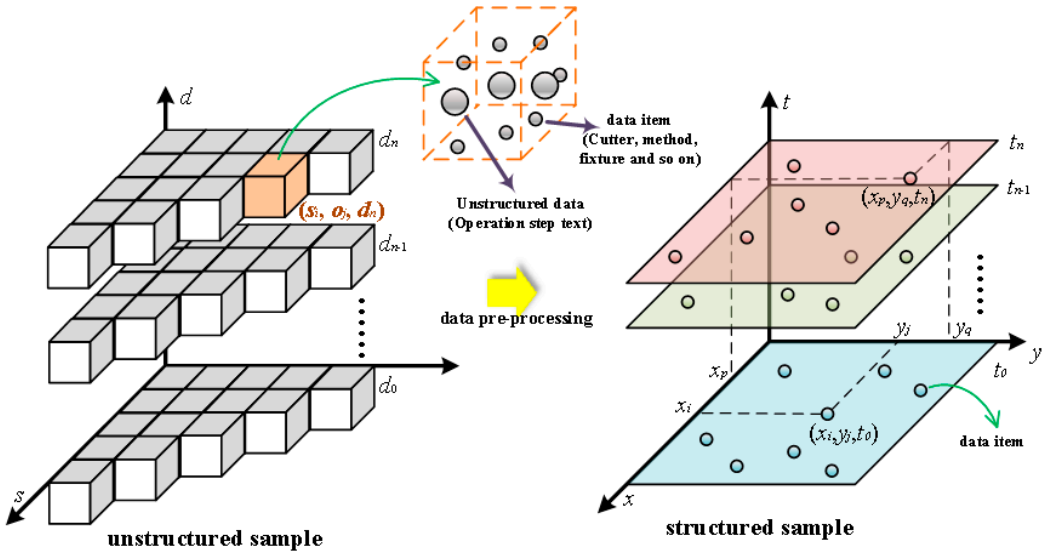


Figure 6. Illustration of Process Planning Documents Data Structuring.

(2) Knowledge mining

In knowledge mining, the prior knowledge of process planners is formalized as a matrix. According to the characteristics of PDK, a fitness function for data association discrimination is defined based on this matrix. Following this, a PDK mining model is developed based on WWO, including the design of propagation, refraction, and breaking operators, the encoding scheme for the algorithm population, and the termination criteria, as depicted in Figure 4. The knowledge mining algorithm is illustrated in Figure 7.

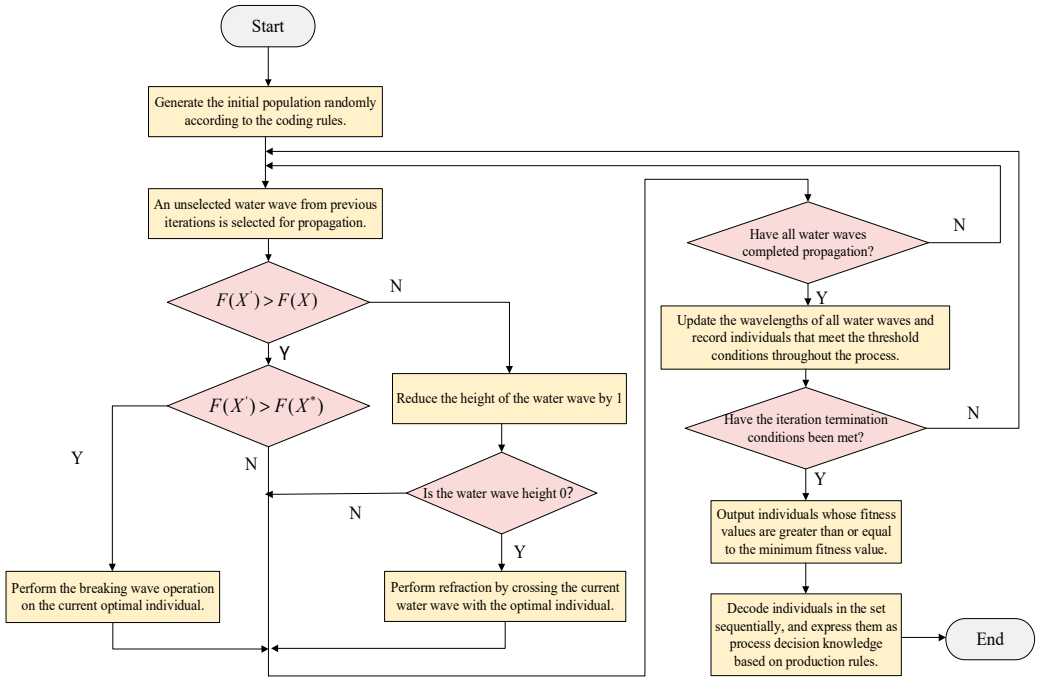


Figure 7. Algorithm flow of PDK mining.

3. The Design and Implementation of Knowledge Constraint with WWO for PDK Mining

The data sample defined by Equation (5) consists of process data items and their corresponding semantics. This study aims to discover semantically related data items and further construct PDK from data samples. Knowledge constraint models the semantic relations among knowledge elements, providing informative representations for subsequent knowledge discovery. In this paper, to fully leverage the semantic information in the data sample, the prior knowledge of process planners is embedded into the knowledge mining algorithm. It should be noted that prior knowledge refers to existing knowledge available before tackling a task or problem, which can be utilized to support or accelerate the learning and reasoning processes of algorithms.

3.1. Knowledge Constraint-Based Association Discrimination Model

In the context of PDK, semantic association captures the interconnections or relations among process data items at the semantic level [21]. It is a critical factor to consider for PDK mining. In this paper, information describing the semantic relations between process data items is used as prior knowledge for building a function to determine whether the rule-form combinations represent valid PDK. To effectively embed prior process knowledge into WWO, it is necessary to quantify the semantic associations among process data items and establish a mathematical model of the knowledge.

3.1.1. Representation of Prior Knowledge

The semantic association between pairwise process data items is defined as follows: during process planning, if the process element to which data item x_1 belongs serves as one of the rule antecedents for determining the instance value of the process element to which data item x_2 belongs, a semantic association is considered to exist between data item x_1 and data item x_2. This relation can be defined as follows:

$$y_{x_1} \rightarrow y_{x_2}, x_1, x_2 \in T \tag{6}$$

where y_{x_1} and y_{x_2} denote the semantic labels of process data items x_1 and x_2, respectively, and T denotes the process data sample. When a semantic association exists, x_1 must belong to the rule antecedents of PDK, while x_2 belongs to the rule consequents. For example, when process planners select machining methods, machining features serve as one of the determining factors. Therefore, "*machining method*" and "*machining feature*" are correlated according to Equation (6). In this case, data related to "*machining method*" belong to the consequents of the machining method selection rule, while data related to "*machining feature*" belong to the rule antecedents.

Based on Equation (6), the semantic association degree between pairwise process data items is defined as follows:

$$\mathrm{SR}_{x_1x_2} = \begin{cases} 1 & , y_{x_1} \rightarrow y_{x_2} \\ 0 & , otherwise \end{cases} \tag{7}$$

where $SR_{x_1x_2}$ denotes the degree of semantic association between process data items x_1 and x_2. During process planning, if a semantic association exists between process data items x_1 and x_2, their semantic association degree is defined as 1; otherwise, it is defined as 0. Under certain conditions, the semantic association degree between process data items of the same type is set as 0.

Based on Equation (6), the prior knowledge to be embedded can be represented as a semantic association matrix. The construction steps are as follows:

Step 1: Enumerate the semantic labels contained in the prior knowledge.
Step 2: Initialize a matrix where the rows and columns are respectively headed by these semantic labels.
Step 3: Commencing from the first row, traverse each element sequentially from left to right. If the semantic corresponding to the element's row is associated with the semantic corresponding to its column according to Equation (6), assign the element a value of 1; otherwise, assign it 0.
Step 4: Repeat this process until all elements of the matrix have been traversed.

Through the above construction process, the prior knowledge to embed is represented as a semantic association matrix as follows:

$$\mathrm{M} = \begin{pmatrix} \mathrm{R}_{11} & \mathrm{R}_{12} & \cdots & \mathrm{R}_{1,\mathrm{n}-1} & \mathrm{R}_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \mathrm{R}_{ij} & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathrm{R}_{\mathrm{n}1} & \mathrm{R}_{\mathrm{n}2} & \cdots & \cdots & \mathrm{R}_{nn} \end{pmatrix}_{\mathrm{n}\times\mathrm{n}} \tag{8}$$

where M represents the semantic association matrix to represent the prior knowledge, n denotes the number of semantic labels involved in the prior knowledge, and R_{ij} denotes the association degree between the semantic corresponding to the i-th row and the semantic corresponding to the j-th column. For example, given prior knowledge describing the associative relations among "*part material*", "*machining feature*", "*machining method*" and "*cutter grade*", it can be constructed into semantic association matrix based on Equation (8), as illustrated in Figure 8.

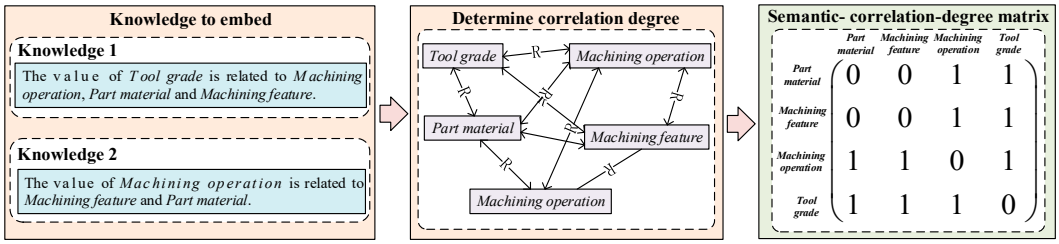


Figure 8. Matrix-based representation of prior knowledge.

3.1.2. Construction of Fitness Function

Based on the semantic association matrix, a fitness function is designed to evaluate the likelihood that a rule-form combination of process data items serves as valid PDK. According to the characteristics of PDK, the following properties can be derived:

① PDK resides within process instances. In the transaction database defined by Equation (5), for a set of data items to constitute valid PDK, the frequency of their joint occurrence in process instances must be non-zero.

② For each data item in the rule consequents, there must exist at least one semantically associated data item within the rule antecedents.

③ For a combination of process data items, the likelihood of it constituting valid PDK increases with the number of semantically associated data items within the rule consequents.

Based on the above properties, the fitness function F is developed. According to Property ①, support metric is adopted for determining whether a data item combination can serve as PDK. An indicator s is introduced, which is the support value of the data item combination:

$$s = Sup(Rule) \tag{9}$$

where *Rule* denotes a rule-form combination of process data items, and *Sup*(*Rule*) denotes the support value of *Rule* in the data sample defined by Equation (5). When the support value of *Rule* is zero, *Rule* cannot be retained as PDK, and its fitness value F is zero.

According to Property ②, an indicator r is introduced, which is a 0–1 variable designed for the association discrimination of *Rule*. If a data item in the consequent of *Rule* lacks a semantic association with any other items within *Rule*, r is defined as 0. In this case, the fitness value F of *Rule* is 0.

According to Property ③, only when both s and r are not zero, the fitness function for the rule-form combination is calculated as follows:

$$F(Rule) = Min(\ln A, \frac{1}{\ln A}) \frac{\sum_{0}^{Num(y_{x_i} \to y_{x_j})} 1}{A \times C + Num(y_{x_m} \to y_{x_n})} \tag{10}$$

where *Rule* denotes a rule-form combination of process data items; A denotes the number of data items in the rule antecedents; C denotes the number of data items in the rule consequents; x_i and x_j denote data items in *Rule*; x_m and x_n denote data items in the consequent of *Rule*; $Num\left(y_{x_i} \to y_{x_j}\right)$ denotes the number of semantic association relations in *Rule*; and $Num(y_{x_m} \to y_{x_n})$ denotes the number of semantic association relations in the consequent of *Rule*, both determined based on the semantic association matrix.

In summary, provided a rule-form data item combination *Rule*, the calculation process of its fitness value is as follows:

Step 1: Calculate the value of r for *Rule* based on the semantic association matrix.

Step 2: If $r = 0$, assign $F(Rule) = r$, and proceed to Step 4. Otherwise, calculate the value of s for *Rule* on the data sample.
Step 3: If $s = 0$, set $F(Rule) = s$. Otherwise, calculate $F(Rule)$ according to Equation (10).
Step 4: The value of $F(Rule)$ is obtained, and the calculation ends. The workflow is illustrated in Figure 9.

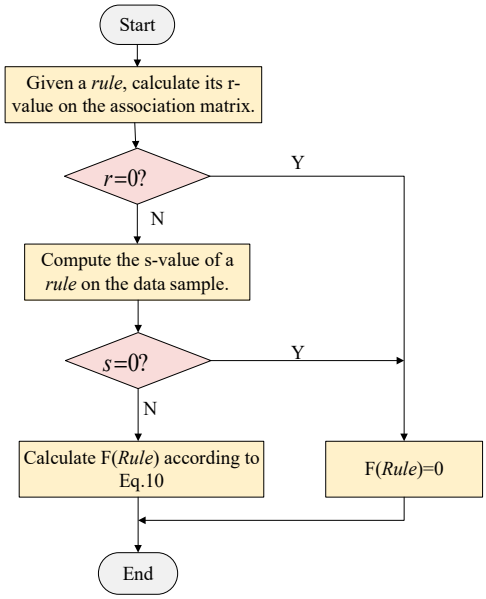


Figure 9. Flow process of fitness calculation for individuals of *Rule*.

In conclusion, the development of the fitness function was primarily motivated by two considerations as follows:

- Alignment with machining process planning requirements: The fitness function was designed to directly measure the adaptability and validity of PDK in machining processes. It reflects the domain logic consistency of PDK. This ensures that the PDK mining process remains tightly aligned with the research objectives, enabling the discovery of more accurate PDK to support subsequent machining process planning.
- Guidance for discrete search: The fitness function is specifically designed to score candidate rules based on two key aspects: data-driven support (frequency) and knowledge-constraint semantic validity. This dual-objective scoring provides a smooth and informative fitness landscape, which effectively guides the algorithm to explore neighborhoods containing high-quality PDK while avoiding regions dominated by invalid rules.

3.2. Design of Operators and Realization of the Knowledge Mining Approach

Considering the characteristics of the PDK mining problem, an encoding scheme for discrete data, algorithmic operators, and a workflow for PDK mining were developed in this paper. To facilitate the definition and description of the algorithm model, the symbols used in this section are listed in Table 2.

(1) Coding Scheme

The individuals processed by WWO are rule-form combinations of process data items. According to the PDK model, the engineering constraint is that an instance of a specific process element can appear only once in an individual during algorithm execution. A real-number encoding scheme is adopted. The steps of encoding individuals during population initialization are as follows:

Step 1: Scan the process data sample $T = \{t_1,t_2,t_3\ldots\ldots,t_n\}$ to establish the set of semantic labels $Y = \{y_1,y_2\ldots y_m\}$.

Step 2: Establish an empty set S. Sequentially add elements from set T with semantic labels $y_1,y_2\ldots y_m$. to S, and renumber the elements in S in the order of addition. Eventually, $S = \{s_1,s_2,\ldots s_n\}$ is obtained.

Step 3: Randomly generate k one-dimensional vectors of length l ($l \leq n$) from set S. In this vector, the first element denotes the index value that partitions the rule antecedents from the consequents, while each remaining element denotes the index (within set T) of a corresponding process data item. For example, the vector $P = \{4,2,4,5,3,6,7,8\}$ represents the rule "*If* $\{s_2,s_4,s_5,s_3\}$ *Then* $\{s_6,s_7,s_8\}$", as shown in Figure 10.

Step 4: To satisfy the engineering constraint, a data elimination operation is performed on each individual: the population is processed to eliminate semantic duplicates within each individual by randomly retaining one instance per unique semantic label.

Table 2. Notation and basic symbols.

Symbols	Notation
P	Rule Encoding
x	Process Data Items in a Rule
X	Original Water Wave Individual
X'	Updated Water Wave Individual
X^*	Optimal Water Wave Individual from the Previous Generation Population
Y	Set of Semantic Labels Included in the Data Sample
λ	Original Wavelength of the Water Wave
λ'	Updated Wavelength of the Water Wave
λmax	Maximum Wavelength of the Water Wave
i,j	Index
k	Number of Individuals in the Initial Population
ε	A Minimal Positive Number
e	Step Size for Neighborhood Search
$round()$	Rounding Function
$Length()$	Function that Returns the Bit Length of an Encoded Individual
N	Number of Iterations
$minF$	Threshold—Minimum Fitness Value
$F_{N\text{-}1_max}$	Global Optimum at Iteration N-1
F_{N_max}	Global Optimum at Iteration N

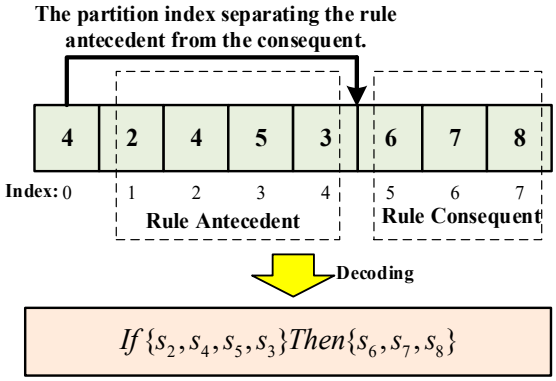


Figure 10. Illustration of training dataset for TransD.

Based on this encoding scheme, the process data sample T is further mapped into a rule space. This space consists of multiple rule-form combinations and serves as the search space for the algorithm. The objective of the algorithm is to search for individuals with higher fitness values from this rule space and extract them as the results of PDK mining.

(2) Propagation Operator

In this paper, the propagation process of water waves over a seabed is simulated by performing neighborhood searches on encoded individuals. A key parameter of the propagation operator, the neighborhood step size e, is directly determined by the water wavelength λ. This step size critically controls the search scope of the propagation operator. In the method, a constraint is that neighborhood search step size e must be an integer, i.e., $e = round(\lambda)$. Let the wavelength of each rule individual be defined as

$$\lambda = \alpha\lambda_{\max}^{(f(x)-f(x)_{\min}+\varepsilon)/(f(x)_{\max}-f(x)_{\min}+\varepsilon)} \tag{11}$$

where $\lambda_{max} = \sum_{l}^{k} length(P)/3$. If an individual's fitness is the minimum in the current generation, set its wavelength λ to a small positive constant $\varepsilon(\varepsilon = 0.00001)$ to avoid division by zero. After each iteration, update the wavelength λ for each individual in the population; α is the attenuation coefficient of the water wave wavelength.

In this paper, neighborhood search aims to identify rules that are semantically similar to a given rule. Semantically similar rules are characterized by similarities in the semantic labels of their data items. Neighborhood-search-based propagation involves two types of operations as follows:

① Neighborhood search within the same semantic scope: This operation modifies a specific data item while ensuring that the semantic labels remain unchanged in the rule individual.

② Cross-semantic neighborhood search: This operation adds a new data item to the rule individual, whose semantic label differs from those of the existing data items.

Neighborhood search within the same semantic scope. A code position A is selected randomly in individual X, where the digit at this position is i, corresponding to data item s_i. The semantic label of this data item is y_i. In set S, the set of instance values with semantic label y_i is $S_{y_i} = \{s_{i-c}, s_{i-c+1}, s_{i-c+2}, \ldots s_i \ldots, s_{i+b-1}, s_{i+b},\}$. Based on the step size e, starting from s_i, sequentially traverse e elements in the positive direction to obtain process data item s_j. During this process, if the end of set S_{y_i} is reached, the traversal wraps around to the first element of the set and continues in positive order until e elements have been traversed. Replace s_i with s_j, as illustrated in Figure 11.

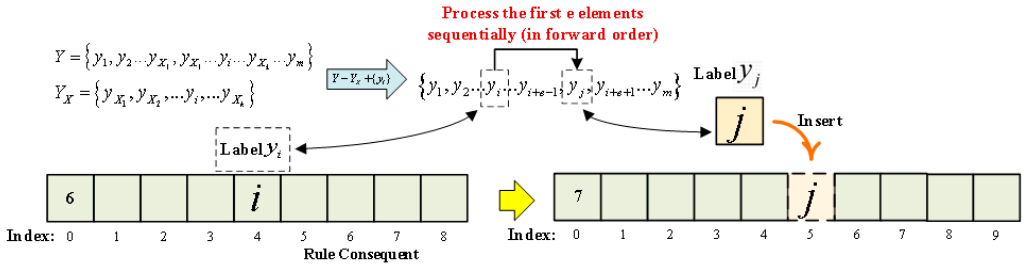


Figure 11. Illustration of the neighborhood search within the same semantic scope.

Cross-semantic neighborhood search. Let the set of semantic labels included in the data sample be Y, and the set of semantic labels contained in X be $Y_X(Y_X \subseteq Y)$. A code position B is selected randomly in individual X, where the semantic label corresponding to this position is y_i. Through the set operation $Y - Y_X + \{y_i\}$, set Y' is obtained. In Y', label y_j is obtained by traversing e elements based on the step size e forward from y_i. An instance value with label y_j is randomly selected and inserted into code position B, as presented in Figure 12.

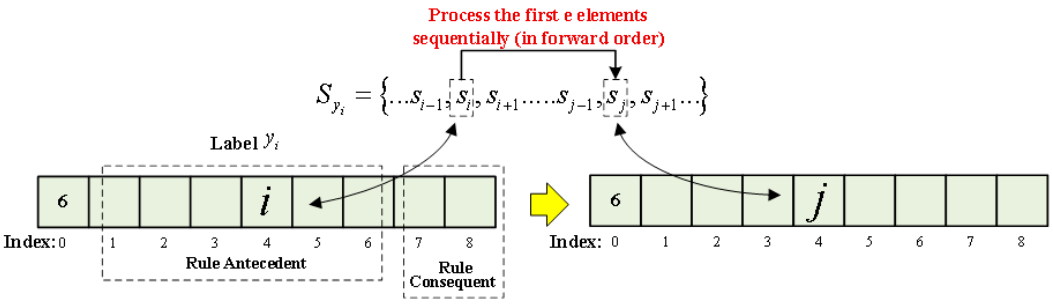


Figure 12. Illustration of the cross-semantic neighborhood search.

The complete propagation operation is as follows: A neighborhood search is performed within the same semantic scope on water wave X to obtain a new water wave individual X'. If $F(X') > F(X)$, replace the original water wave with the new one. If $F(X') \leq F(X)$, a cross-semantic neighborhood search is performed on the water wave to obtain another new water wave X''. If $F(X'') > F(X)$, replace the original water wave X with X''. Otherwise, retain the original water wave X.

(3) Refraction Operator

The refraction operation is performed on an individual when its water wave height is 0. It is achieved by performing a crossover operation between the current water wave individual and the best individual in the current generation. Following this, the characteristics of high-quality individuals can be inherited. In this paper, a two-point crossover method was used for the refraction operation. Provided that two individuals F_1 and F_2 are crossed, a crossover point C_{11} is randomly selected in the antecedents of F_1, and another crossover point C_{12} is randomly selected in the consequents of F_1. Subsequently, crossover points C_{21} and C_{22} are randomly selected in the antecedents and consequents of F_2, respectively, and the antecedents and consequents of F_1 and F_2 are divided into two segments, each by C_{21} and C_{22}. A new individual S_1 is generated by recombining rules F_1 and F_2: the first part of F_1's antecedents are concatenated with the second part of F_2's antecedents, and the first part of F_1's consequents are concatenated with the second part of F_2's consequents. Similarly, the remaining segments are spliced to generate another new individual S_2. The same operation is performed on F_2. The crossover process is illustrated in Figure 13.

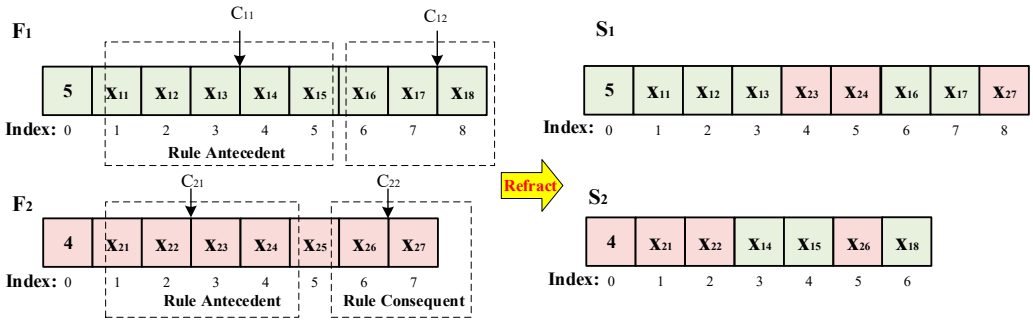


Figure 13. Illustration of the crossover-induced refraction. Color legend: Light green squares represent data items originally belonging to regular individual F_1, while light red squares represent data items originally belonging to regular individual F_2.

After the crossover operation, the new individuals S_1 and S_2 may contain process data items with duplicate semantics. To ensure that the new individuals satisfy the constraint, a data elimination operation must be performed. Assume that in S_1, the duplicated data items involve semantic labels y_1 and y_2, where the instance values of y_1 in S_1 are a_{11} and a_{12}, and the instance values of y_2 in S_1 are a_{21} and a_{22}. According to the constraint that only

one instance value per semantic label is retained, the data elimination for S_1 can yield four possible individual candidates (as shown in Figure 14). The individual with the highest fitness value among these individual candidates is retained as the final result of the data elimination for S_1.

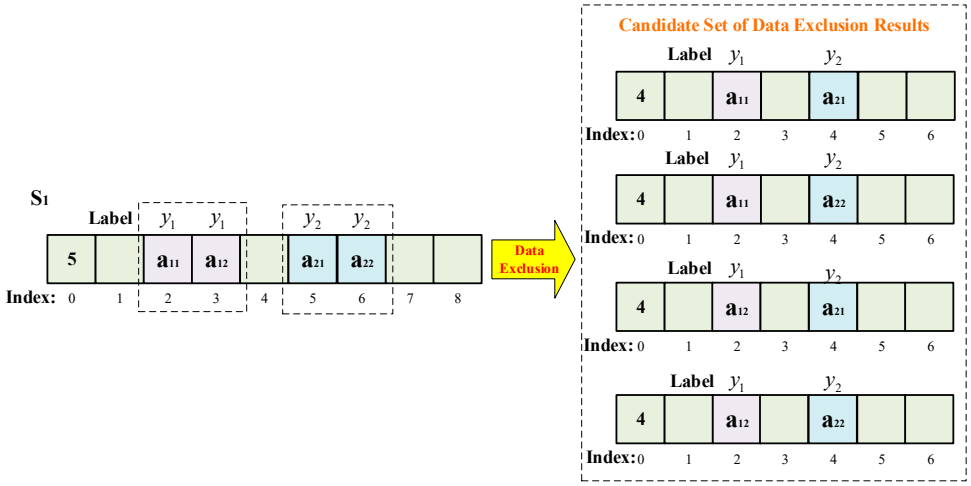


Figure 14. Illustration of the data exclusion for rule individual.

Following refraction, the wave height of the new water wave individual is reset to the maximum wave height h_{max}, and its wavelength λ is updated according to Equation (12).

$$\lambda' = \lambda \frac{F(X')}{F(X)} \tag{12}$$

(4) Breaking Wave Operator

Following propagation operation, a breaking operation is performed when a water wave individual reaches a position superior to the current optimal water wave individual. This aims to conduct a local search around this individual to find a better candidate within its neighborhood and propagate the water wave to a more favorable position.

In the breaking operation, m code positions are randomly selected and modified to generate a new individual for mutation in a semantically consistent manner. If the fitness values of all new water wave individuals are lower than that of the original water wave, the original individual is retained; otherwise, the new individual is preserved for the next generation. The modification of code positions by the breaking operator is consistent with the neighborhood search operation in propagation operations.

(5) Termination Condition

During the initialization of the WWO algorithm for PDK mining, a minimum fitness value $minF$ must be set. A rule-form combination (water wave individual) is retained as valid PDK only if its fitness value is greater than or equal to $minF$. This screening is applied in each iteration. The iteration terminates when the fitness value of the optimal individual stabilizes after the N-th iteration, as defined in Equation (13), or when the algorithm reaches the predefined maximum number of iterations N_{max}.

$$|F_{N_\max} - F_{N-1_\max}| \leq 0.1 \tag{13}$$

(6) Algorithm Flow

By constructing the water wave operators, each water wave individual is characterized by three attributes: wavelength λ, wave height h, and position x in the sample space. The

process of mining PDK is achieved through dynamic changes in these three attribute values. The workflow is as follows:

Step 1: Initialization of WWO algorithm. An empty set R is established to store encoded individuals that meet the threshold condition.
Step 2: An initial population is randomly generated according to the encoding scheme, and the wavelength of each individual is recorded in the population.
Step 3: Propagation operation is performed on the population: a water wave x is randomly selected from the initial population for propagation to obtain a new water wave x'. Then, $F(x')$ is compared with $F(x)$.
Step 4: If $F(x') \leq F(x)$, the wave height h of water wave x is decreased by 1. If the wave height h of x equals to zero, a refraction operation is performed: x undergoes crossover with the optimal individual; otherwise, skip to **Step 6**.
Step 5: If $F(x') > F(x)$, $F(x')$ is compared with $F(x^*)$. If $F(x') > F(x^*)$, a breaking operation is performed on the current optimal individual.
Step 6: Upon completion of propagation by all water waves, the wavelengths of all water waves are updated according to Equation (12); otherwise, return to **Step 3**. Water wave individuals that meet the threshold condition are recorded and added to set R during propagation.
Step 7: When the algorithm meets the termination condition, the water wave individuals in set R are decoded.
Step 8: The decoded rules are represented as PDK based on Equation (1). The algorithm ends. The pseudocode of workflow is presented in Table 3.

Table 3. Pseudocode of algorithm for mining knowledge.

Checkpoint	Pseudocode
1	Initial parameters: The population size M, the maximum wavelength λ_{max}, the maximum wave height h_{max}, the attenuation coefficient α, and the minimum fitness *minF*.
2	Initial an empty set R, and population $D = \{x_1, x_2, x_3 \ldots \ldots x_m\}$ based on encoding scheme.
3	Evaluate all the individuals in D, and keep the individuals of rule meeting the threshold condition in set R.
4	Knowledge(x):
5	Calculate $F(x)$ based on Equation (10).
6	If $F(x) \geq minF$, keep x in R.
7	While termination condition is not met:
8	Perform propagation on x, then x' is generated.
9	Knowledge(x) and Knowledge(x')
10	IF $F(x) \leq F(x^*)$:
11	IF $F(x) \leq F(x')$:
12	$h \leftarrow h - 1$
13	IF $h = 0$:
14	Perform refraction on x^*
15	Knowledge(x^*)
16	ELSE:
17	Perform breaker on x'
18	Knowledge(x')
19	If all the elements in D have been operated by propagation,
20	Update λ based on Equation (12)
21	End While
22	Return set R

3.3. Quantitative Performance Measures

The effectiveness of the algorithm is evaluated from three aspects: the number of correct rules, the accuracy rate, and the speed of knowledge accumulation during the mining process. The definitions of the three evaluation metrics are defined as follows:

Metric 1: Number of Correct Rules—K_{total} (pcs).

This metric measures the quantity of correct PDK discovered by the algorithm. An effective knowledge mining method should be capable of discovering a substantial amount of usable PDK. This directly supports accurate process planning and reduces the risk of erroneous process decisions for stable process quality. The K_{total} is defined as the number of correct rules in the mining results.

Metric 2: Accuracy Rate in Mining Results—A_{cc} (%).

This metric refers to the proportion of correct PDK in the mining results. It is a key indicator for assessing the quality of PDK mining. Higher accuracy indicates more reliable and usable rules and reflects stronger performance of the algorithm. This reliability in turn minimizes the need for subsequent manual intervention to filter and rectify erroneous rules. The accuracy rate A_{cc} is defined as follows:

$$A_{cc} = \frac{K_{total}}{N_{total}} \times 100\% \tag{14}$$

where N_{total} denotes the total number of rule-form combinations in the mining results.

Metric 3: Knowledge Mining Speed—(V_k pcs/s).

This metric quantifies the rate at which correct rules are discovered and extracted per unit of time during the mining process. It serves as an indicator of the knowledge mining module's core capability, determining how fast an intelligent CAPP system can accumulate process knowledge in the production environment. The PDK mining speed V_k is defined as follows:

$$V_k = \frac{K_{total}}{T_{total}} \tag{15}$$

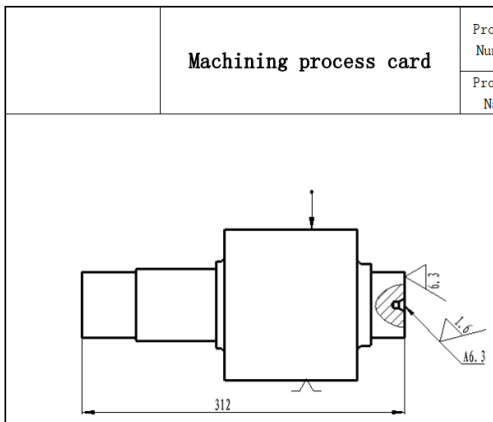
where T_{total} denotes the total runtime of the knowledge mining algorithm.

The three aforementioned metrics can be obtained by performing knowledge mining on process data samples.

4. Case Study

The proposed method is validated using machining process data from housing, plate, shaft and gear parts as case studies. The process planning documents required for the experiments were sourced from a general parts manufacturing plant as shown in Figure 15. A total of 52 machining process planning documents for four types of parts were collected and organized. Using OCR technology and natural language processing methods, these electronic process planning documents were processed and converted into machining process data samples according to Equation (5), which are stored in two-dimensional tables, as illustrated in Figure 5. The experimental data are organized into four categories:

- **Shaft parts (527 records):** Gear shafts, tappets, pin shafts, centrifugal shafts.
- **Gear parts (513 records):** Internal gear rings, planetary gears, couplings.
- **Housing parts (535 records):** Valve bodies, turbine housings, transmission cases.
- **Plate parts (568 records)**: Positioning plates, spacer plates, plate molds.



Machining process card	Product Number		Part Drawing Number		page 12 of 15
	Product Name		Part Name	gear shaft	

Workshop	Operation Number	Operation Name	Material Mark
machining workshop	6	lathe finish	40Cr
Blank	Blank dimensions	Number of pieces /blank	PCS
forging	Φ150×318	1	1
Device Name	Unit Type	Unit Number	
horizontal lathe	C6140		

Fixture Number	Fixture Name	Cutting Fluid

Working Position Apparatus	Apparatus Name	Work Hour (min)	
		Endpoint	Piece
		4	4

Step Num	Step work content	Craft Equipment	Spindle Speed	Cutting Speed	Feed Rate	Cutting Depth	Feed Number	Step Hour	
			r/min	m/min	mm/r	mm		Motor	Aux
1	Clamp the top circle, turn the left cover end and make total length 312	Three-jaw chuck, A6.3 center drill	400	76.15	0.3	2	1	2	2
2	Finish centre hole (A6.3)	Three-jaw chuck, A6.3 center drill	400		0.1	13	1	2	2

Figure 15. Instance of machining process planning documents.

These process data were utilized for five analytical objectives: (1) illustrating the effectiveness of data pre-processing by the NER model; (2) validating the correctness of the proposed approach; (2) optimizing the parameters of the PDK mining method; (3) benchmarking the performance of the novel association model against existing alternatives; (4) evaluating the comparative efficacy of the WWO search strategy compared to other heuristic algorithms; and (5) demonstrating its superiority over existing mining approaches.

4.1. Illustration of Data Pre-Processing

The quality of data pre-processing is crucial to subsequent PDK mining. In this paper, a CRF-BiLSTM (Conditional Random Field-Bidirectional Long Short-Term Memory) model is used for NER (named entity recognition) tasks, which performs word segmentation and semantic annotation on machining process texts described in natural language. Due to space limitations in this paper, the training process and parameter optimization of the CRF-BiLSTM model are not detailed. The optimized model parameters are presented as follows: the number of hidden units in the BiLSTM layer is set to 256, and the learning rate is 0.003. The accuracy, recall, and F_1-score of the model on the datasets of four parts are provided in Table 4.

Table 4. Evaluation metrics for NER task on four types of part datasets.

Part Type	Accuracy	Recall	F_1-Score
Shaft part	0.902	0.939	0.921
Gear part	0.901	0.932	0.913
Housing part	0.907	0.926	0.910
Plate part	0.915	0.928	0.919

As illustrated in Table 4, the employed CRF-BiLSTM model achieves outstanding results, with all three evaluation metrics exceeding 0.9. This robust performance in NER tasks ensures reliable pre-processing of text data, facilitating the construction of high-quality structured data samples essential for accurate PDK mining.

4.2. Validation of Method Correctness

Prior to conducting knowledge mining by the proposed method, the parameter settings were as follows: initial wave height $h_0 = 6$, initial wavelength $\lambda_0 = 14$, initial population size of 15, maximum iteration number of 100, and minimum fitness $minF = 0.4$.

In this paper, the correctness of PDK was assessed through a manual review. In this process, experienced process planners or experts evaluated each piece of PDK, selecting usable knowledge from the PDK candidates obtained using the algorithm. Following manual screening, the validated PDK is archived in the process knowledge base for subsequent process planning. Partial mining results are presented in Table 5. Taking the rule "*IF* {*Planetary Gear*, *Rough Turning*, *40Cr*, *Cylindrical Surface*} *THEN* {*YT5*, *C620-1*}" as an example, this rule-form knowledge enables process planners to select appropriate tools and machining methods when planning the machining process for planetary gears. Specifically, for a planetary gear made of a *40Cr* material, rough turning of the cylindrical surface can be performed using carbide tool *YT5* on a *C620-1* horizontal lathe.

Table 5. Partial mining results of the algorithm.

ID	Rule Antecedent	Rule Consequent	Correctness
1	{45 Steel, IT13, Ra12.5}	{Rough Turning}	√
2	{20CrMnTi, End Face, Ra6.3, IT9}	{Semi-Finish Turning, YT30 Cutter}	√
3	{Internal Gear Ring, End Face, Rough Turning}	{Self-Centering Chuck}	√
4	{Pin Shaft, 45 Steel, End Face, Rough Turning}	{YT5 Cutter}	√
5	{Planetary Gear, Rough Turning, 40Cr, Cylindrical Surface}	{YT5 Cutter, Machine Tool C620-1}	√
6	{Tappet, Finish Turning, AL6061-T6}	{<*Parameter Set_12*>, CDW010 Cutter}	√
7	{Plate Hole, IT8, TC4, Curved Face, YG820 Cutter }	{<*Parameter Set_7*>,Machine Tool i5M8.5,Finish Turning }	√
8	{Housing part, 304L, Curved Surface, Ra0.4}	{<*Parameter Set_3*>}	√

Note: <*Parameter Set_3*>, <*Parameter Set_7*> and <*Parameter Set_12*> are process parameter sets. For example, <*Parameter Set_12*> consists of cutting speed (v) = 60 m/min, feed rate (f) = 0.2 mm/r, and depth of cut (ap) = 2 mm.

The relation curves for maximum fitness F_{N_max} values and metric K_{total} with respect to iterations are presented in Figure 16. As illustrated in Figure 16a, with increasing iterations, the fitness value of the optimal individual monotonically increases and eventually converges toward a stable maximum. This demonstrates that the proposed method can effectively search for near-optimal rule-form knowledge. As illustrated in Figure 16b, the relation between the number of PDK obtained by the algorithm and the iterations reveals the following trends: before a certain iteration, the fitness value of the globally optimal individual searched by the algorithm was lower than the predefined *minF*, resulting in a K_{total} value of zero. The appearance of individuals that meets the minimum fitness threshold directs the neighborhood search toward promising regions in the sample space, enabling the sequential discovery of further qualified individuals. This leads to a gradual rise in the total count of discovered PDK as the algorithm iterates. At this stage, due to the high diversity within the population and significant differences among individuals, the algorithm generated diverse candidate rules through water wave operators, resulting in a rapid increase in K_{total} value. Owing to the stochastic search of WWO, the algorithm cannot evaluate all individuals in the sample space. Progressively, the unexplored search space gradually contracts, and the amount of remaining undiscovered PDK decreases, reducing the new PDK discovered per iteration. This results in a slow increase in the K_{total}

metric, which eventually stabilizes. These results demonstrate that the proposed method can effectively achieve PDK mining, verifying the correctness of the algorithm.

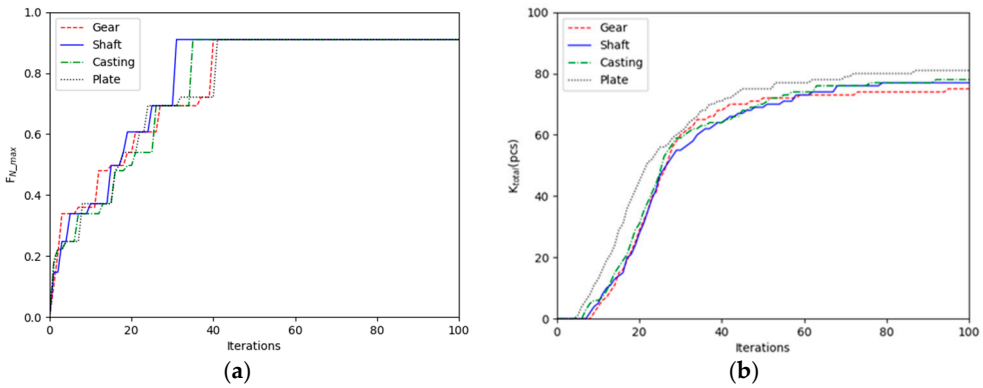


Figure 16. Curves of algorithm performance indicators: (**a**) best individual fitness value; (**b**) number of extracted process knowledge.

Furthermore, a PDK mining module has been developed based on the proposed method to facilitate automated knowledge accumulation. As presented in Figure 17a, this module supports the construction and management of structured process data samples. Figure 17b illustrates the function of PDK mining and the processing of knowledge mining results. In this system, the refinement processing for mining results is performed in a human-in-the-loop manner, where process planners manually verify each candidate rule, filter out erroneous ones, and store the correct rules into a process knowledge base, as presented in Figure 17b.

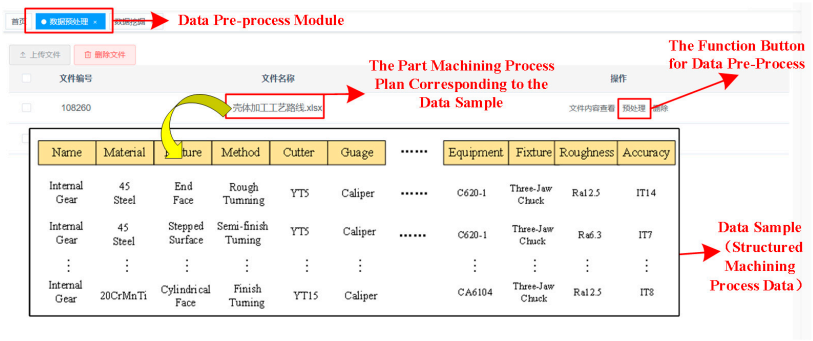


Figure 17. Development of the PDK mining module: (**a**) machining data sample construction interface; (**b**) PDK mining interface.

4.3. Analysis Based on Different Algorithm Parameters

The minimum fitness value *minF*, wavelength attenuation coefficient α, maximum wavelength λ_{max}, and maximum wave height h_{max} are the key parameters of the proposed method. Proper configuration of these parameters directly influences the algorithm performance. In this paper, a parameter sweep approach was conducted to optimize *minF*, while grid search was employed for the combinatorial optimization of α, λ_{max}, and h_{max}.

The fitness function and the minimum fitness value *minF* are core components of the association discrimination model. They determine whether a rule-form data item combination can be retained as PDK in the process of knowledge mining. The *minF* value critically determines both the amount (K_{total}) and the proportion (A_{cc}) of correct rules in the mining result. Line charts depicting K_{total} and A_{cc} of the algorithm under different thresholds (minimum fitness *minF*) on various part data samples are provided in Figure 18. As presented in Figure 18a, when *minF* ranges between 0.1 and 0.4, a larger number of individuals in the sample space meet the threshold condition. Consequently, the algorithm discovers a greater quantity of rule individuals and is more likely to mine more correct PDK. However, due to the high N_{total} value in the mining results, the A_{cc} value of the algorithm is relatively low at this stage, as presented in Figure 18b. As the *minF* value gradually increases, the number of individuals satisfying the threshold condition in the sample space decreases, leading to a reduction in the total number of rule individuals discovered by the algorithm. Correspondingly, the A_{cc} value gradually increases. As *minF* increases to 0.4~0.7, it begins to surpass the fitness values of certain correct PDK. These correct PDK that do not meet the threshold condition are excluded by the algorithm, leading to a decline in the K_{total}. However, because the fitness values of the retained rule individuals are no lower than those of the excluded correct PDK, each retained rule is more likely to be correct. This results in a higher proportion of correct PDK in mining results, and a continued upward trend in the A_{cc} value. When *minF* rises to 0.7~0.9, all rule individuals in the mining results are correct PDK, and consequently, the A_{cc} reaches 100%. Since fewer individuals in the sample space meet the threshold condition, the K_{total} value remains relatively low. In conclusion, to balance both K_{total} and A_{cc} of the algorithm, it is appropriate to set *minF* within the range of 0.3~0.5. When *minF* is set to around 0.4, the A_{cc} stabilizes at approximately 90%, and the K_{total} value reaches 70~80.

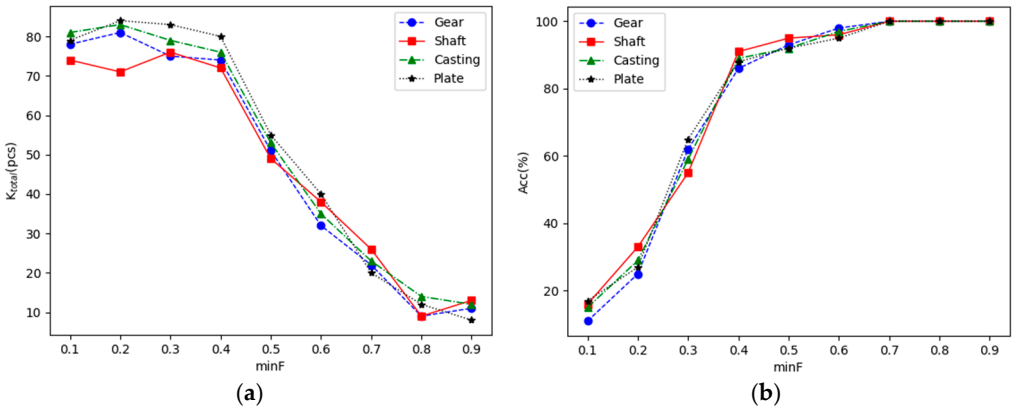


Figure 18. Evaluation metrics under varying thresholds: (**a**) variation in K_{total}; (**b**) variation in A_{cc}.

The wavelength attenuation coefficient α, maximum wavelength λ_{max}, and maximum wave height h_{max} collectively determine the propagation of water waves, such as wave speed, propagation range, frequency, and amplitude. These parameters are key factors that collectively influence the performance of WWO. For WWO, the algorithm may overlook dense clusters of promising rules in the sample space based on a global search strategy,

reducing the amount of correct PDK in the mining results. Conversely, when the algorithm performs mainly based on a local search strategy, it can precisely discover individuals meeting the threshold condition within a limited scope but tends to be trapped in local optimum. This leads to insufficient diversity of PDK in the mining results.

Therefore, to balance the exploration capabilities of the algorithm in both global and local spaces, the three parameters must be appropriately configured. In this paper, grid search was employed to optimize the parameters, and the K_{total} values of the algorithm under different parameter combinations are presented in Figure 19. Four values of λ_{max} were selected: P/5, P/4, P/3, and P/2. The attenuation coefficient α was tested in the range of 0.1~0.6, and the maximum wave height h_{max} was varied within the range of 4~12. As is illustrated in Figure 19b, the optimal configuration is that $\alpha = 0.2$, λ_{max} = P/3, and h_{max} = 7. Under this configuration, the algorithm achieves the highest K_{total} value of 81, demonstrating its ability to discover a substantial amount of PDK in the sample space.

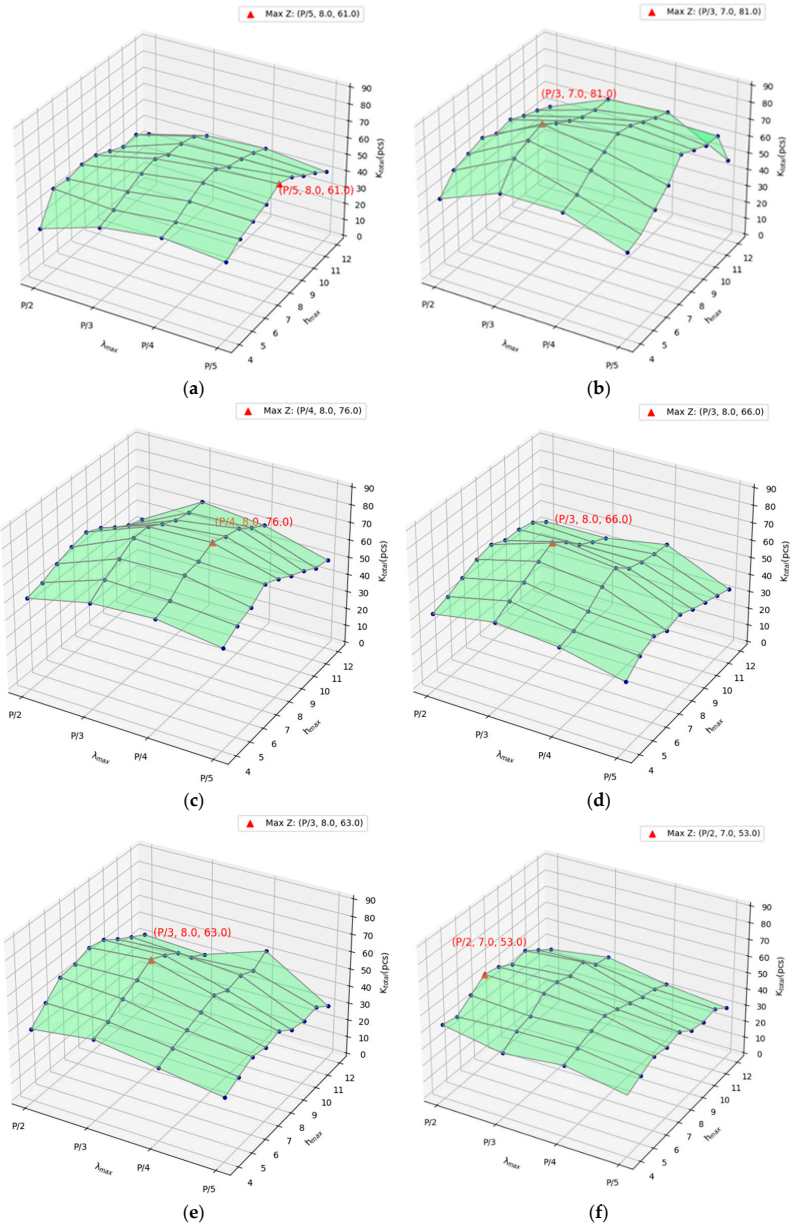


Figure 19. Values of K_{total} across different algorithm parameter combinations. (**a**) K_{total} as a function of h_{max},λ_{max} (α = 0.1); (**b**) K_{total} as a function of h_{max},λ_{max} (α = 0.2); (**c**) K_{total} as a function of h_{max},λ_{max} (α = 0.3); (**d**) K_{total} as a function of h_{max},λ_{max} (α = 0.4); (**e**) K_{total} as a function of h_{max},λ_{max} (α = 0.5); (**f**) K_{total} as a function of h_{max},λ_{max} (α = 0.6).

4.4. Ablation Study of Association Discrimination Models Under a Fixed Search Strategy

The association discrimination model constitutes a critical component of the proposed approach, as it directly influences the A_{cc} and K_{total} values of the algorithm. Support/Confidence [14], Statistical Correlation (*SC*) [16], and *Lift* [19] are commonly used association discrimination metrics in association rule mining problems. They are established based on the probability levels of data item combinations in the sample. In this paper, comparative experiments were conducted in line with the principle of the control variable method.

An ablation study is presented in this section to evaluate the effectiveness and superiority of the proposed association discrimination model. The analysis is performed by comparing different association models under a fixed search strategy, thereby isolating the contribution of the association model itself. Based on the search strategy of WWO, PDK mining was performed on the machining process data samples of gear parts using the following association discrimination models: the proposed knowledge-constraint model, Support/Confidence, Statistical Correlation (*SC*), and *Lift*. The thresholds for these algorithms were set to 0.4, 0.2, 0.2, and 0.25, respectively. The results are provided in Table 6.

Table 6. K_{total} and A_{cc} under different association discrimination metrics.

Group	The Proposed Knowledge-embedded Metric		Support/Confidence [14]		Statistical Correlation [16]		Lift [19]	
	K_{total} (pcs)	A_{cc} (%)	K_{total} (pcs)	A_{cc} (%)	K_{total} (pcs)	A_{cc} (%)	K_{total} (pcs)	A_{cc} (%)
1	71	88.75%	26	19.26%	31	21.38%	25	24.27%
2	75	91.46%	22	18.64%	29	20.71%	19	17.43%
3	81	91.01%	25	20.33%	27	24.11%	23	20.72%
4	79	91.86%	19	18.81%	31	25.83%	22	21.57%
5	77	89.53%	29	23.57%	20	22.22%	28	28.57%
6	82	92.13%	25	22.52%	26	22.03%	31	27.43%
7	78	91.76%	23	21.70%	27	19.85%	18	18.18%
8	79	88.76%	27	21.09%	25	23.81%	27	26.21%
9	83	89.25%	26	24.53%	30	27.27%	33	30.28%
10	80	90.91%	22	23.66%	28	25.93%	32	29.63%
Avg	78.5	90.54%	24.4	21.41%	27.4	23.31%	25.8	24.43%

As presented in Table 5, employing the same search strategy of WWO, the proposed knowledge-constraint association metric outperforms conventional probability-based metrics in PDK discovery, serving as the key factor that enables WWO to achieve an A_{cc} value of approximately 90%.

4.5. Ablation Study of Search Strategies Under a Fixed Association Model

As a key component of the proposed approach, the search strategy directly impacts the K_{total} and K_v metrics of the algorithm. An effective search strategy ensures that the mining algorithm can discover more PDK within a limited time. In this section, an ablation study is conducted to demonstrate the effectiveness and superiority of the proposed WWO strategy by comparing different search strategies under the same association discrimination model.

(1) Comparative analysis on different heuristic search strategies

The WWO-based strategy was compared with four heuristic algorithms through systematic analysis. Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Bee Colony Optimization (BSO) are all heuristic algorithms that have been widely applied to combinatorial search problems due to their unique mechanisms. A controlled comparative study was conducted. Four search strategies—WWO, PSO, GA and BSO—were separately

applied to PDK mining on the datasets of four-type parts. The common parameters for the four heuristic algorithms were set as follows: initial population size = 15, maximum number of iterations = 60, and minimum fitness value *minF* =0.4. Meanwhile, the remaining parameters of PSO, GA, and BSO have been optimized. A detailed account of the optimization is omitted for conciseness, and the optimized parameters are listed in Table 7.

Table 7. Parameter settings.

Algorithm	Parameter Name	Parameter Symbol	Value
PSO	Velocity Inertia Weight	w	0.6
PSO	Acceleration Coefficient	c_1	2
PSO	Acceleration Coefficient	c_2	2
GA	Crossover Probability	p_1	0.8
GA	Mutation Probability	p_2	0.06
BSO	Inertia Weight	w	0.5
BSO	Individual Learning Factor	c_1	1.5
BSO	Social Learning Factor	c_2	2.0

To reduce performance variance arising from algorithmic components and ensure fairness in the comparative experiments, the four algorithms were implemented within a unified framework where population initialization, individual encoding, constraint-based filtering, and termination proposed in this paper are adopted. The convergence curves of the K_{total} values for the four heuristic algorithms are provided in Figure 20, and the K_{total} and K_v metrics values from ten PDK mining experiments are presented in Table 8.

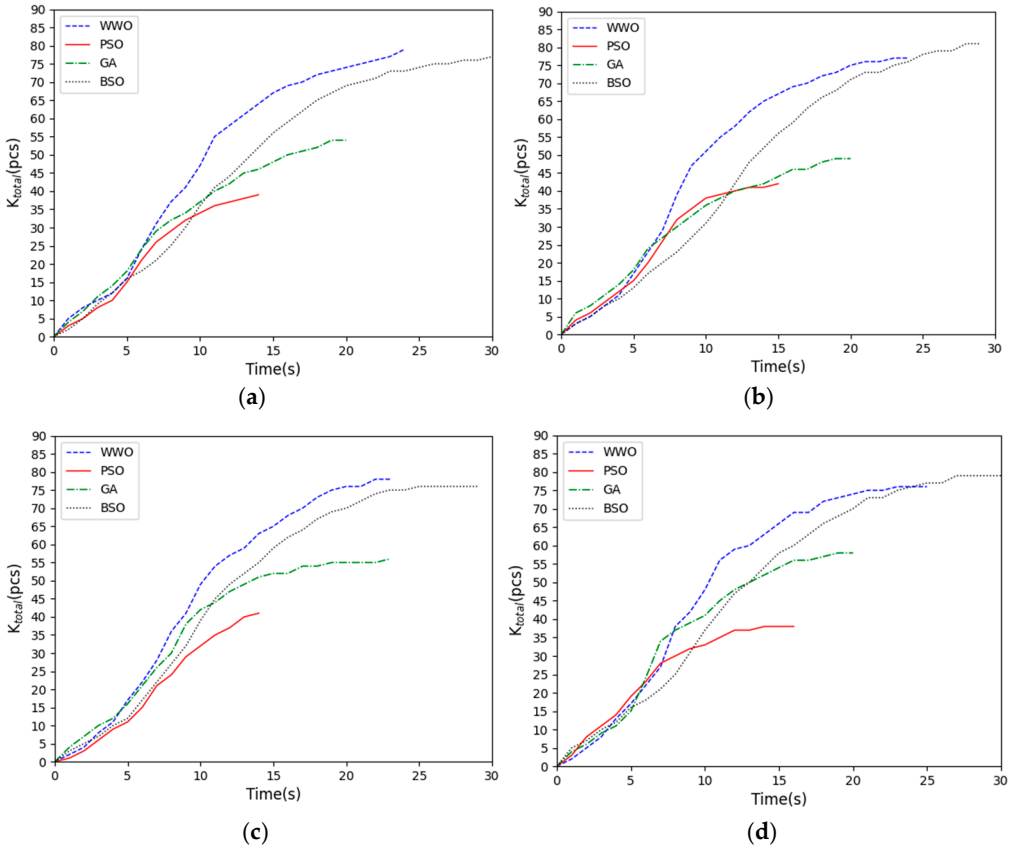


Figure 20. Convergence curves of K_{total} for four heuristic algorithms: (**a**) curves on gear part process data; (**b**) curves on shaft part process data; (**c**) curves on housing part process data; (**d**) curves on plate part process data.

Table 8. Comparison of different heuristic algorithm search strategies.

Group	WWO		PSO		GA		BSO	
	K_{total} (pcs)	K_v (pcs/s)	K_{total} (pcs)	K_v (pcs/s)	K_{total} (pcs)	K_v (pcs/s)	K_{total} (pcs)	K_v (pcs/s)
1	75	3.18	42	2.93	45	2.33	75	2.65
2	76	3.11	39	2.53	48	2.58	82	2.77
3	82	3.25	46	3.13	41	2.02	72	2.53
4	78	3.07	37	2.37	48	2.24	79	2.61
5	80	3.14	41	2.83	44	2.08	76	2.79
6	79	3.25	43	2.85	53	2.75	75	2.47
7	84	3.28	38	2.55	46	2.13	81	2.66
8	82	3.32	37	2.61	55	2.91	71	2.47
9	73	3.09	42	2.69	61	2.95	76	2.73
10	75	3.10	39	2.64	56	2.63	81	2.90
Avg	78.4	3.179	40.4	2.713	49.7	2.462	76.8	2.658

The experimental results demonstrated that under the same association discrimination metric, the algorithms based on the four search strategies—WWO, PSO, GA and BSO—can all achieve effective PDK mining. Due to the non-deterministic and stochastic search mechanisms of heuristic algorithms, the mining results are not repeatable with each run of the algorithm. Therefore, the mean values of the K_{total} and K_v metrics in ten experiments are used to evaluate the performance of the four algorithms in this case study.

As is presented in Table 7 and Figure 20, the observations can be drawn as follows:

- The algorithm based on the PSO search strategy achieved a mean K_{total} value of 40.4, which is lower than the other three algorithms. However, it required the shortest run time for PDK mining, resulting in a high K_v value, second only to WWO.
- The algorithm based on the GA search strategy had a mean K_{total} value higher only than that of PSO, and its mean K_v value was the lowest at 2.462.
- The algorithm based on the BSO search strategy achieved a mean K_{total} value of 76.8, which is close to that of WWO, but it exhibited the slowest convergence speed and required the longest run time for PDK mining. Its mean K_v value was lower than those of PSO and WWO.
- The proposed method achieved the highest mean values for both K_{total} and K_v, at 78.4 and 3.179, respectively.

The results can be attributed to the characteristics of the PDK mining problem and the search mechanisms of the four algorithms. PDK mining aims to search for rule individuals that satisfy the minimum fitness condition within the rule space. The distribution of PDK in this space exhibits the following characteristics: high-quality individuals (whose fitness values are high) are concentrated around other high-fitness rules, while low-fitness rules are predominantly clustered within regions of invalid rules. This characteristic is commonly referred to as the "smoothness" of fitness landscapes in discrete spaces. It implies that once a high-fitness rule is discovered, its immediate vicinity compassing semantically similar and potentially optimal rule variants merits intensive exploitation. Consequently, effective search algorithms tailored for such spaces must incorporate robust local exploitation mechanisms.

- **PSO: Local Exploitation with Premature Convergence Risk**

PSO possesses sufficient local search capability, as search direction is guided by personal best and global best positions, exhibiting a strong tendency toward high-quality individuals. This allows the algorithm to thoroughly explore the neighborhoods of excellent individuals. However, driven by the attraction of locally optimal individuals, PSO is prone to becoming trapped in local optima, hindering its ability to conduct comprehensive

global search. As a result, the PSO-based mining algorithm achieves a relatively low K_{total}, though its fast convergence leads to a high K_v.

- **GA: Global Exploration with Inadequate Local Refinement**

While the selection, crossover, and mutation operators of GA promote global search and maintain population diversity, its performance in discrete rule mining is constrained by an inadequate local search capability. Specifically, standard GA mutation operates stochastically with a fixed probability. In discrete representations, a single random mutation can significantly alter the semantic structure of a rule. This causes the algorithm to exit the local neighborhood prematurely. Although this promotes global exploration, it lacks the fine-grained local exploitation capability essential for refining PDK. Additionally, the crossover operator combines building blocks from two parent individuals, which tends to disrupt coherent rule structures in discrete spaces. Briefly, GA lacks an effective mechanism for detailed neighborhood search around high-fitness individuals, resulting in the oversight of promising candidates in the immediate vicinity of high-quality rules. Consequently, despite its exploratory strength, GA achieves a low K_{total}.

- **BSO: Balanced Search with High Computational Overhead**

The search mechanism of BSO exhibits both strong local and global search capabilities, which results in a high K_{total}. However, the information transfer within the colony operator for local search involves extensive data exchange and computation. Moreover, the switching between global and local search states incurs additional computational overhead, prolonging the knowledge mining process and resulting in a lower K_v.

- **WWO: Controlled Local Exploitation and Complementary Global Search**

In contrast, WWO employs a breaking operator to search for good individuals within the neighborhoods of high-fitness individuals. This operator performs controlled local exploitation, enabling thorough exploration in regions rich with PDK. Meanwhile, the refraction operator performs crossover on low-fitness individuals, allowing the algorithm to escape current local regions and reduce search efforts in areas densely populated with invalid rules. This division of labor—breaking for local exploitation around promising individuals and refraction for controlled global exploration—aligns with the distribution of valid rules in discrete spaces. Furthermore, compared to BSO, WWO does not involve complex local search processes, resulting in lower computational resource consumption.

In conclusion, this comprehensive search strategy allows WWO to achieve higher K_{total} and K_v values. The comparative results demonstrate that the search strategy of WWO is more suitable for PDK mining problems compared to GA, PSO, and BSO, as its operator design explicitly accounts for the neighborhood concentration property of high-quality rules in discrete spaces.

(2) Comparative analysis between the WWO-based strategy and Apriori

Apriori is a traditional association rule mining algorithm that generates all possible candidate rule sets based on data enumeration. In this section, the search strategy of this algorithm is compared with the proposed WWO-based search strategy under the fixed association discrimination model in terms of K_{total} and K_v. The mining results of the two approaches on four categories of the part machining process data are presented in Table 9.

The results indicate that Apriori-based search strategy is capable of mining more process rules compared to the WWO-based approach. However, this comes at the cost of significantly lower mining efficiency, as reflected by its lower K_v. A complexity analysis is subsequently performed to explain this observation as follows:

- **Complexity Analysis of Apriori Algorithm**

The Apriori algorithm is based on a level-wise enumeration approach. Its time complexity is primarily determined by three factors: the number of transactions in the dataset N, the number of candidate item sets M, and the average transaction width w. A brute-force enumeration of all possible item sets results in a time complexity of $O(N \times M \times w)$, where $M = 2^k$ for k different items. This causes exponential growth as the number of items increases. This explains its relatively low K_v despite its ability to achieve higher K_{total}.

- **Complexity Analysis of WWO-Based Search Strategy**

In contrast, the WWO-based search strategy adopts a population-based random heuristic search approach that does not require exhaustive candidate enumeration. Its time complexity is determined by three factors: population size P, maximum number of iterations N and operator execution. The propagation operator, which generates new rules by perturbing existing waves, has a time complexity of $O(P \times d)$ per iteration, where d is the number of process data items. The refraction operator, activated exclusively when rule individuals stagnate, adds $O(P_r \times d)$ overhead where $P_r (P_r \ll P)$ denotes the number of waves to which the refraction operator is applied. The breaking operator, which performs local search around high-fitness rules, has a complexity of $O(P_b \times d \times L)$ where P_b is the number of high-quality rules and L is the local search depth. Therefore, the overall time complexity of the WWO-based approach is $O(T \times P \times d \times L)$, which is linearly dependent on the population size, iteration, and PDK dimension. Unlike Apriori, it does not suffer from exponential growth as the number of items increases. This linear scalability explains its superior mining efficiency, as evidenced by the higher K_v in the experimental results.

The above complexity analysis aligns with the theoretical explanation that while Apriori may discover more PDK due to its exhaustive nature, the WWO-based approach achieves a better balance between rule quantity and mining efficiency, making it more suitable for large-scale PDK mining tasks.

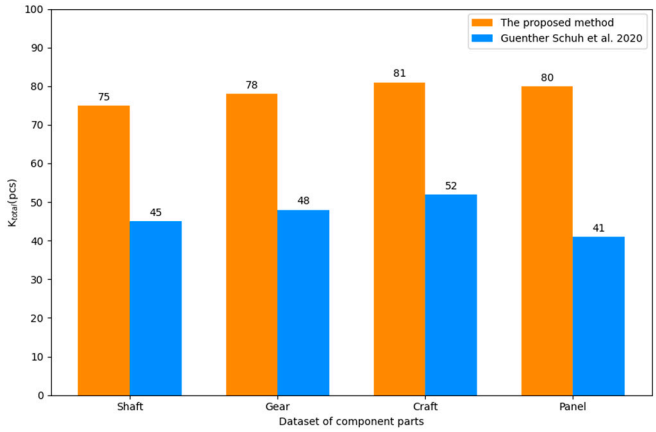
Table 9. K_{total} and K_v of WWO and Apriori on the dataset of four parts.

Part Type	WWO-Based Strategy		Apriori-Based Strategy	
	K_{total} (pcs)	K_v (pcs/s)	K_{total} (pcs)	K_v (pcs/s)
Shaft part	79	3.30	87	1.02
Gear part	77	3.19	85	0.94
Housing part	76	3.15	89	0.89
Plate part	81	3.25	91	1.22

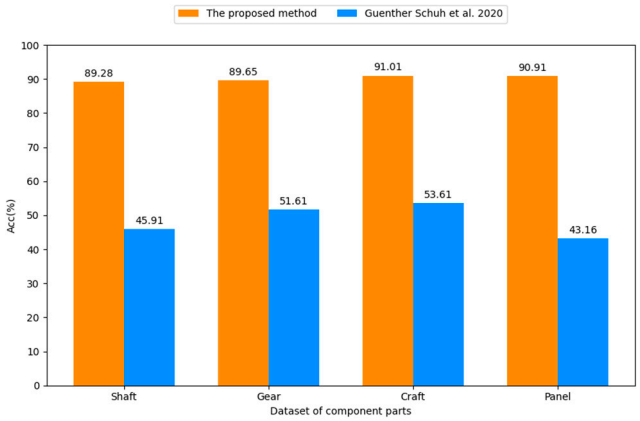
4.6. Comparison with Baseline Methods

DT-PDKM (Decision-Tree-Based Process Decision Knowledge Mining) is a PDK mining method proposed for intelligent CAPP systems [29]. It employs decision trees to perform process knowledge mining. Widely recognized and regarded as a stable, standard approach in the process planning domain, it can serve as a reliable reference benchmark for evaluating the performance of new methods, and it is adopted in this paper to validate the advantages of the proposed approach through comparative analysis.

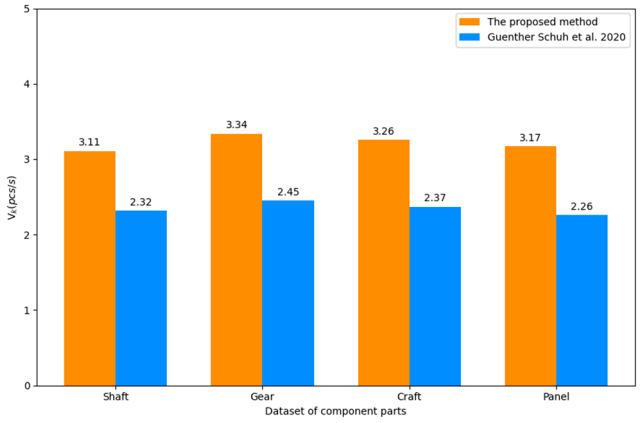
A comparative analysis was conducted between the proposed method and DT-PDKM. Comparative experiments were performed to assess k_{total}, A_{cc} and V_k of the two algorithms. As presented in Figure 21, the k_{total} of DT-PDKM falls within the range of 41~52, its A_{cc} within the range of 43~54, and its V_k within the range of 2.2~2.5. All three metrics are lower than those achieved by the proposed method. This demonstrates that the proposed method can more rapidly discover a greater amount of correct PDK compared to DT-PDKM.



(**a**)



(**b**)



(**c**)

Figure 21. Computational results of two methods on process data from four parts. (**a**) bar chart comparing K_{total} values; (**b**) bar chart comparing A_{cc} values; (**c**) bar chart comparing V_k values [29].

Furthermore, in the study [29], the input of knowledge discovery module based on DT-PDKM is a two-dimensional process decision table, which limits its processing capability to fully structured data stored in process databases. In contrast, the proposed method integrates OCR, named entity recognition (NER), and text correction in the data pre-processing stage. This enables it to handle unstructured data carried in process planning documents, demonstrating strong data adaptability. Consequently, the proposed method is more suitable for PDK mining in environments with multi-source heterogeneous data, thereby effectively supporting rule-based knowledge discovery for intelligent CAPP systems.

5. Discussion

5.1. Contributions

Artificial intelligence has become a transformative force in the evolution of CAPP systems. At present, knowledge discovery is one of the key enabling technologies for building intelligent CAPP systems. In this context, a novel approach was proposed to address the discovery of rule-based decision knowledge from process planning document data in this paper. The main contributions are summarized as follows:

(1) An innovative knowledge mining framework for rule-based domain knowledge was proposed. A knowledge mining framework for PDK mining was proposed in this paper, consisting of an association discrimination model and a search strategy. The association discrimination model is a classification model composed of a fitness function and a threshold condition, which determines whether combinations of process data items can serve as usable decision knowledge. Building on this, a search strategy was developed based on WWO, which efficiently searches for PDK in the sample space. In addition, a process knowledge discovery module for PDK mining was developed in the CAPP system, verifying the correctness and executability of the proposed method.

(2) A knowledge constraint approach for rule-based knowledge mining was introduced. In this paper, a method for integrating process prior knowledge was proposed to address the low accuracy of domain rule-based knowledge by data-driven models. By constructing a matrix that describes the semantic relations between pairs of process data items, the prior knowledge of process planner is mathematically represented. A fitness function was designed based on this matrix to identify the rule-form data item combinations serving as PDK, thereby achieving the integration of prior knowledge into the algorithm model. The experimental results demonstrated that the association discrimination model with prior knowledge integrated can enable the algorithm to effectively discover PDK, achieving an accuracy rate of approximately 90%. Furthermore, this knowledge constraint approach provides methodological guidance for effective rule-based knowledge mining in other domains.

(3) The adaptability of the search strategy based on the WWO for PDK mining was verified. Using the association discrimination metric proposed in this paper, different search strategies were employed for comparative analysis. The experimental results demonstrate that the search strategy based on the WWO balances local and global rule search, rapidly discovering correct PDK from the sample space. In terms of performance evaluation, this search strategy enabled the algorithm to achieve K_{total} and K_v values of 78.4 and 3.179, respectively, outperforming PSO, GA and BSO algorithms. This confirms the adaptability of WWO for the PDK mining problem.

5.2. Limitations and Future Work

In the domain of process planning, the semantic labels involved are relatively limited in scope and number, rendering the manual construction of the semantic association matrix a manageable task. However, in more complex domains where such a matrix becomes difficult to maintain, the proposed method exhibits certain limitations. Under such circumstances, it is not necessary to include all semantic labels in the matrix. Instead, a partial semantic association matrix can be established, which focuses only on the most critical or user-specified entity categories. For the remaining data and labels, where explicit semantic associations are unavailable, a data-driven method can be employed to discover implicit relations by statistical analysis. This integrated approach constitutes a hybrid data–knowledge-driven framework for rule-based knowledge discovery, enhancing both scalability and applicability across domains. Nevertheless, when transferring the proposed method to a new domain, the semantic association matrix must be reconstructed based

on domain-specific expert knowledge. This reliance on manual reconfiguration limits the transferability and generalizability of the method and restricts its adaptive capacity when applied to diverse and dynamic application scenarios.

Future work could explore automated learning mechanisms for semantic association across multiple domains to improve the generalization and reduce manual effort in cross-domain applications. In parallel, investigation is needed to optimize the trade-off between manual domain expertise and automated data-driven approaches, ensuring an effective balance between knowledge constraint and scalability. Furthermore, with the evolution of advanced manufacturing technologies, new knowledge continuously emerges and gradually replaces existing or legacy knowledge. Driven by iterative updates in machining processes, and the continuous introduction of new materials or equipment, the knowledge stored in process knowledge bases face challenges of timeliness decay and obsolescence. Therefore, another future work could focus on developing adaptive updating mechanisms for process knowledge bases to achieve autonomous evolution of process knowledge. This mechanism, operating in synergy with the knowledge discovery module, will collectively establish a technical foundation for the realization of intelligent CAPP systems.

Author Contributions: X.X.: conceptualization, methodology, investigation, visualization, writing—original draft. Z.H.: conceptualization, methodology, writing—review and editing, supervision. L.Q.: methodology, writing—review and editing, supervision. Y.W.: writing—review and editing, investigation. C.C.: writing—review and editing, visualization. Z.L.: writing—review and editing, investigation. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PDK	Process Decision Knowledge
WWO	Water Wave Optimization
DT-PDKM	Decision Tree-Based Process Decision Knowledge Mining

References

1. Zhang, E.; Zhang, S.W.; Jia, J.L. A dynamically updatable knowledge graph construction method for computer-aided process planning and design. *J. Eng. Des.* **2025**, *36*, 1551–1584. [CrossRef]
2. Liu, Q.; Li, X.; Gao, L. A novel MILP model based on the topology of a network graph for process planning in an intelligent manufacturing system. *Engineering* **2021**, *7*, 807–817. [CrossRef]
3. Xu, Q.; Zhou, G.; Zhang, C.; Chang, F.; Cao, Y.; Zhao, D. Generative AI and DT integrated intelligent process planning: A conceptual framework. *Int. J. Adv. Manuf. Technol.* **2024**, *133*, 2461–2485. [CrossRef]
4. Cheng, K.; Zhao, G.; Wang, W.; Liu, Y.; Hu, D. A STEP-NC complaint and feature-based solution for intelligent process planning. *Int. J. Adv. Manuf. Technol.* **2024**, *133*, 6023–6039. [CrossRef]
5. Kumar, S.P.L. State of the art-intense review on artificial intelligence systems application in process planning and manufacturing. *Eng. Appl. Artif. Intell.* **2017**, *65*, 294–329. [CrossRef]
6. Deja, M.; Siemiatkowski, M.S. Machining process sequencing and machine assignment in generative feature-based CAPP for mill-turn parts. *J. Manuf. Syst.* **2018**, *48*, 49–62. [CrossRef]
7. Xu, T.; Chen, Z.; Li, J.; Yan, X. Automatic tool path generation from structuralized machining process integrated with CAD/CAPP/CAM system. *Int. J. Adv. Manuf. Technol.* **2015**, *80*, 1097–1111. [CrossRef]

8. Imsetif, J.; Nezamoddini, N.; Aqlan, F. Job shop planning and scheduling of reconfigurable manufacturing systems. *Oper. Manag. Res.* **2025**, *18*, 891–912. [CrossRef]
9. Kaiyao, Z.; Xiangming, F.; Wenlei, X.; Gang, Z. A novel solution of automated process planning techniques for STEP-NC manufacturing. *Int. J. Adv. Manuf. Technol.* **2025**, *138*, 287–314. [CrossRef]
10. Stanković, A.; Petrović, G.; Turudija, R.; Marković, D.; Ćojbašić, Ž. Hybrid machine learning approach for parallel machine scheduling under uncertainty. *Expert Syst. Appl.* **2025**, *279*, 127427. [CrossRef]
11. Huang, Z.; Guo, X.; Jiang, C.; Yang, M.; Xue, H.; Zhao, W.; Wang, J. mKGMPP: A multi-layer knowledge graph integration framework and its inference method for manufacturing process planning. *Adv. Eng. Inform.* **2025**, *65*, 103266. [CrossRef]
12. Chaube, A.; Benyoucef, L.; Tiwari, M.K. An adapted NSGA-2 algorithm based dynamic process plan generation for a reconfigurable manufacturing system. *J. Intell. Manuf.* **2012**, *23*, 1141–1155. [CrossRef]
13. Nowak-Brzezińska, A.; Wakulicz-Deja, A. Exploration of rule-based knowledge bases: A knowledge engineer's support. *Inf. Sci.* **2019**, *485*, 301–318. [CrossRef]
14. Zhang, S.; Wu, X. Fundamentals of association rules in data mining and knowledge discovery. *WIREs Data Min. Knowl. Discov.* **2011**, *1*, 97–116. [CrossRef]
15. Khan, S.; Parkinson, S. Eliciting and utilising knowledge for security event log analysis: An association rule mining and automated planning approach. *Expert Syst. Appl.* **2018**, *113*, 116–127. [CrossRef]
16. Ghafari, S.M.; Tjortjis, C. A survey on association rules mining using heuristics. *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1307. [CrossRef]
17. Zhuang, C.; Liu, Z.; Liu, J.; Ma, H.; Zhai, S.; Wu, Y. Digital twin-based quality management method for the assembly process of aerospace products with the grey-markov model and apriori algorithm. *Chin. J. Mech. Eng.* **2022**, *35*, 105. [CrossRef]
18. Chunduri, R.K.; Cherukuri, A.K. Scalable algorithm for generation of attribute implication base using FP-growth and spark. *Soft Comput.* **2021**, *25*, 9219–9240. [CrossRef]
19. Wang, L.; Guo, Y.; Guo, Y.; Xia, X.; Zhang, Z.; Cao, J. An improved eclat algorithm based association rules mining method for failure status information and remanufacturing machining schemes of retired products. *Procedia CIRP* **2023**, *118*, 572–577. [CrossRef]
20. Kamsu-Foguem, B.; Rigal, F.; Mauget, F. Mining association rules for the quality improvement of the process design. *Expert Syst. Appl.* **2008**, *40*, 1034–1045. [CrossRef]
21. Garcia, A.C.B.; Ferraz, I.; Vivacqua, A.S. From data to knowledge mining. *Ai Edam* **2009**, *23*, 427–441. [CrossRef]
22. Chemchem, A.; Drias, H. From data mining to knowledge mining: Application to intelligent agents. *Expert Syst. Appl.* **2015**, *42*, 1436–1445. [CrossRef]
23. Sahu, A.; Sinha, S.; Banka, H. Fuzzy inference system using genetic algorithm and pattern search for predicting roof fall rate in underground coal mines. *Int. J. Coal Sci. Technol.* **2024**, *11*, 1. [CrossRef]
24. He, P. Multi-dimensional Information Multimedia Big Data Mining Analysis Relying on Association Rule Mapping Model. *Arab. J. Sci. Eng.* **2024**, *50*, 7361–7373. [CrossRef]
25. Chen, L.; Jiang, S.; Liu, J.; Wang, C.; Zhang, S.; Xie, C.; Liang, J.; Xiao, Y.; Song, R. Rule mining over knowledge graphs via reinforcement learning. *Knowl.-Based Syst.* **2022**, *242*, 108371. [CrossRef]
26. Zhou, J.T.; Li, X.Q.; Wang, M.W.; Niu, R.; Xu, Q. Thinking process rules extraction for manufacturing process design. *Adv. Manuf.* **2017**, *5*, 321–334. [CrossRef]
27. Jiao, J.; Zhang, L.; Zhang, Y.; Pokharel, S. Association rule mining for product and process variety mapping. *Int. J. Comput. Integr. Manuf.* **2013**, *21*, 111–124. [CrossRef]
28. Feng, Y.; Wang, H.; Xia, G.; Cao, W.; Li, T.; Wang, X.; Liu, Z. A machine learning-based data-driven method for risk analysis of marine accidents. *J. Mar. Eng. Technol.* **2024**, *24*, 147–158. [CrossRef]
29. Schuh, G.; Prote, J.-P.; Hünnekes, P. Data mining methods for macro level process planning. *Procedia CIRP* **2020**, *88*, 48–53. [CrossRef]
30. Saleem, K.; Kumar, A.; Prasad, K.D.V.; Alkhayyat, A.; Ramachandran, T.; Dey, P.; Kaur, N.; Sivaranjani, R.; Sapaev, I.B.; Mottaghi, M. Robust development of data-driven models for methane and hydrogen mixture solubility in brine. *Geomech. Geophys. Geo-Energ. Geo-Resour.* **2025**, *11*, 37. [CrossRef]
31. Fatnassi, K.; Zahaf, S.; Gargouri, F. Artificial intelligence integration for extension of big data for decision-making. *Future Gener Comput. Syst.* **2025**, *166*, 107635. [CrossRef]
32. Xia, Q.; Etienne, A.; Dantan, J.-Y.; Siadat, A. Reconfigurable machining process planning for part variety in new manufacturing paradigms: Definitions, models and framework. *Comput. Ind. Eng.* **2018**, *115*, 206–219. [CrossRef]
33. Zhao, F.; Liu, H.; Zhang, Y.; Ma, W.; Zhang, C. A discrete water wave optimization algorithm for no-wait flow shop scheduling problem. *Expert Syst. Appl.* **2018**, *91*, 347–363. [CrossRef]
34. Yan, Z.; Zhang, J.; Zeng, J.; Tang, J. Water wave optimization algorithm for autonomous underwater vehicle path planning problem. *J. Intell. Fuzzy Syst.* **2021**, *40*, 9127–9141. [CrossRef]

35. Zhao, F.; Zhang, L.; Zhang, Y.; Ma, W.; Zhang, C.; Song, H. An improved water wave optimisation algorithm enhanced by CMA-ES and opposition-based learning. *Connect. Sci.* **2020**, *32*, 132–161. [CrossRef]
36. Zheng, Y.J. Water wave optimization: A new nature-inspired metaheuristic. *Comput. Oper. Res.* **2015**, *55*, 1–11. [CrossRef]
37. Gu, J.; Du, M.; Chen, Y.; Chen, S. Discover Green's function based on symbolic regression with physical hard constraints. *Phys. Fluids.* **2025**, *37*, 037131. [CrossRef]
38. Zhang, L.; Du, M.; Bai, X.; Chen, Y.; Zhang, D. Complex-valued physics-informed machine learning for efficient solving of quintic nonlinear Schrödinger equations. *Phys. Rev. Res.* **2025**, *7*, 013164. [CrossRef]
39. Zhang, H.; Shao, X.; Zhang, Z.; He, M. E-PINN: Extended physics informed neural network for the forward and inverse problems of high-order nonlinear integro-differential equations. *Int. J. Comput. Math.* **2024**, *101*, 732–749. [CrossRef]
40. Arzani, A.; Cassel, K.W.; D'SOuza, R.M. Theory-guided physics-informed neural networks for boundary layer problems with singular perturbation. *J. Comput. Phys.* **2023**, *473*, 111768. [CrossRef]
41. Leung, W.T.; Lin, G.; Zhang, Z. NH-PINN: Neural homogenization-based physics-informed neural network for multiscale problems. *J. Comput. Phys.* **2022**, *470*, 111539. [CrossRef]
42. Mattey, R.; Ghosh, S. A novel sequential method to train physics informed neural networks for Allen Cahn and Cahn Hilliard equations. *Comput. Methods Appl. Mech. Eng.* **2022**, *390*, 114474. [CrossRef]
43. Mohan, S.; Kasthuri, N. Automatic Segmentation of Psoriasis Skin Images Using Adaptive Chimp Optimization Algorithm–Based CNN. *J. Digit. Imaging* **2023**, *36*, 1123–1136. [CrossRef]
44. Chen, Y.; Huang, D.; Zhang, D.; Zeng, J.; Wang, N.; Zhang, H.; Yan, J. Theory-guided hard constraint projection (HCP): A knowledge-based data-driven scientific machine learning method. *J. Comput. Phys.* **2021**, *445*, 110624. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.