A Graph Theory-Based Optimization Design for Complex Manufacturing Processes

ZHONG HAN1, LIN CHENG2, (Senior Member, IEEE), LIUDONG XING3, (Senior Member, IEEE), AND LITING TIAN2

1School of Information Technology and Engineering, Qiongtai Normal University, Haikou 571127, China
2Department of Electrical Engineering, Tsinghua University, Beijing 100084, China
3Department of Electrical and Computer Engineering, University of Massachusetts Dartmouth, Dartmouth, MA 02747, USA

Corresponding author: Zhong Han (hanyaozhong@sina.com)

This work was supported in part by the National Natural Science Foundation of China (NNSFC) under Award 51807107, in part by the Special Scientific Research of Shaanxi Provincial Department of Education (SSR-SPDE) under Award 18JK0713, in part by the Energy Internet Research Institute, Tsinghua University, China, and in part by the Joint project of Ministry of Education and Guangdong Province in the production and research.

ABSTRACT The manufacturing process of modern equipment becomes very complex due to features such as mass units, multiple machining, and complicated coupling-relationships, posing a big challenge for determining the manufacturing scheme. This paper addresses the challenge by proposing a graph theory-based optimization design for the complex manufacturing process. A detailed analysis of a serial of graph models built according to the manufacturing process features reveals that the Hamilton graph is suitable for modeling the manufacturing process system. Some model weight assignment functions are extracted for the quantitative study. Further the optimal scheme for an optimization design of the complex manufacturing process is solved using the full link graph feature algorithm – a search optimization algorithm. A manufacturing model matrix is constructed, and penalty number and divisor are formulated to simplify the matrix and improve the algorithm efficiency in the process of algorithm design. An example is provided to demonstrate feasibility and effectiveness of the proposed method.

INDEX TERMS Graph model, graph theory, manufacturing process, model weight, optimization design.

I. INTRODUCTION Due to advances in science and technology, the manufacturing capacity of modern equipment becomes more and more powerful so that many complex manufacturing processes can be completed automatically and intelligently [1]–[2]. According to studies in [3]–[5], most of the manufacturing processes of modern equipment belong to the discrete manufacturing (any two machining actions can be combined probably arbitrarily and processed in an adjacent order). Thus, for a task involving many machining actions, a large number of manufacturing schemes need to be arranged [6]–[8]. Different schemes incur different costs. Therefore, an intelligent design that can lead to an optimal solution becomes very significant for the complex manufacturing process [9], [10].

The manufacturing cost computation is quite complex, which involves a variety of factors such as index requirements, processing resources, machining ways [11], and work efficiency [5], [12]. The research contents contain two parts. The first part is to effectively organize these factors to form a processing scheme. An arbitrary combination sequence of processing actions is generated as a difference manufacturing scheme. As the quantity of the action unit increases, the number of possible schemes increases quickly following a geometric progression (like a “combination explosion”). The second part of the research is to determine an optimal processing scheme. Due to the explosion problem, it is often impossible to manually select an optimal scheme; an intelligent design is necessary for solving the optimization problem.

Considerable research efforts have been expended on the intelligence for complex manufacturing processes. Existing researches are mainly concerned with aspects such as modeling of complex manufacturing processes, quality control, intelligent methods [12], [13], energy management, and the big data analysis. Specifically, system modeling is often divided into physical modeling [14], logical modeling [15], [16], and hybrid modeling. Manufacturing modeling involves both the modeling method research [17]–[19], and
the modeling applications [20]. The quality control research focuses on the system reliability [21]–[23], vulnerability [24], cost issues [25], and the quality control method [26]. The intelligent methods research involves 4 aspects: 1) intelligent problems [27] and their review [28]; 2) intelligent ways, e.g., the bionic method [29], and mathematical method [30]; 3) optimization algorithms (e.g., optimization review [31], and optimization means [32]); and 4) optimal control decisions (e.g., decision index [33], [34]); and 4) energy managements [35]. The big data has been widely applied to the manufacturing process (e.g., feature data extraction and data decision [36]).

To the best of our knowledge, none of the above mentioned works use the graph theory to find the optimal scheme for the complex manufacturing processes though it can facilitate an effective and tangible solution to the optimization problem. This work makes contributions by proposing an intelligent method based on the graph theory representation of the complex manufacturing processes. A detailed case study is performed to verify the proposed method and compare the proposed method with the existing bionic intelligent algorithms.

In the current research, works in [37], [38] are relatively close to our study. Among them, literature [37] mainly solves the obstacle of the non-manufacturable or non-machineable results of topology optimization in process manufacturing, while our paper focuses on the research of the fixed topology optimization in process manufacturing. In reference [38], the coefficient of Johnson-Cook model of ultra-fine titanium is determined by an inversion algorithm, while the chip formation model and the search method of Kalman filter iterative gradient are adopted. Different from the study in [38] that is aimed at continuous processing, the proposed study is applicable to the discrete processing.

II. MANUFACTURING PROCESSES MODELING

A. NOTATION

| Symbol | Description |
|--------|-------------|
| \( X \) | Vector of \( n \) variables |
| \( Y \) | Processing efficiency/cost |
| \( f(X_i) \) | Efficiency/cost function of \( X_i \) |
| \( C \) | Computation capacity |
| \( P \) | Tool position |
| \( N \) | Tool number |
| \( V_{ij} \) | Motion speed from node \( i \) to \( j \) |
| \( D_{ij} \) | Distance between two processing unit/node \( i \) and \( j \) |
| \( U_{ij} \) | Transmission speed of material from node \( i \) to \( j \) |
| \( \Phi_{ij} \) | Machining shape from node \( i \) to \( j \) |
| \( \delta_{ij} \) | Machining size from node \( i \) to \( j \) |
| \( \delta_{ij} \) | Material plastic from node \( i \) to \( j \) |
| \( M \) | Matrix of vector \( X \) |
| \( R_i \) | Divisor number of the \( i \)th row. The minimum number of the \( i \)th row |
| \( P_i \) | Penalty number of the \( i \)th row. The difference between the minimum number and the second smallest number of the \( j \)th row |
| \( R'_i \) | Divisor number of the \( i \)th column. The minimum number of the \( i \)th column |

B. MANUFACTURING EQUIPMENT COMPOSITION

Modern manufacturing equipment mostly adopts the design idea of modularization and generalization. Thus, it can be regarded as the combination and integration of multiple manufacturing function modules. Among the numerous manufacturing equipment, the computerized numerical control (CNC) is one of the most commonly used equipment. Though the external forms of different CNCs are diverse, the structure of their component modules is similar or the same. The typical CNC model structure is shown in Fig. 1.

The CNC equipment structure contains two parts: the mechanical system, and the information control system. The mechanical system can be divided into the main body, energy conversion equipment, motion transmission equipment, motion actuator and other parts. Among them, the motion actuator has complex composition and is the core part of the processing, which can produce different shapes through doing work and releasing energy.

For the complex CNC system, its manufacturing process is usually composed of cutting, stamping, drilling, curling, stamping, engraving, typing and other processing. Modern advanced CNC can carry nearly 100 kinds of cutting tools and realize up to 100 kinds of machining. For example, the forming CNC (FCNC) is aimed at the forming and processing of metal plate materials, and widely used in the manufacture of automobile shell, external box of electric appliance, etc. Fig. 2 illustrates the tool library structure of the FCNC.

In this paper, FCNC is used as the research prototype to introduce the proposed optimal control method. The modeling of the complex manufacturing process and the definition of the model quantization function in the follow-up study are all built with reference to the composition of FCNC.

C. MANUFACTURING COMPLEXITY ANALYSIS

The equipment manufacturing is developing towards being intelligent. During the process, many practical problems are encountered and must be solved. Among them, the
automation of machining solutions is a difficult problem to be solved in the process of implementing the intelligent manufacturing. Through research and analysis of the manufacturing process, a complex manufacturing process can be abstracted into a complex network model. Based on this network model, the problem of scheme determination of the manufacturing process is then solved.

At present, modern intelligent manufacturing involves upgrading the traditional manufacturing equipment, and introducing the latest information electronics technology. It still has a lot of work to do in terms of resource utilization and performance. Modern manufacturing is characterized by large-scale, large-volume, and multiple processes. In the face of such a complex situation, it is difficult to work out an efficient and optimized processing plan. The primary task of the manufacturing process is to determine the processing sequence. Due to the discrete nature of each processing step, the previous processing step and the next processing step are often loosely coupled. Therefore, for tasks with \( N \) processing steps, the first step has \( N \) options; the second step has \( N - 1 \) options; and so on. In general, when \( R \) steps have been processed, there are \( N - R \) processing options for the next step; the process ends until all the \( N \) steps are completed. Thus, there are \( N \) combinations or schemes. When \( N \) is large, it is hard or even impossible to find the optimal scheme out of the \( N \) schemes manually.

The choice here is based on the index of processing efficiency or cost. With reference to the CNC or FCNC kinematic structure, the calculation of manufacturing selection cost is determined based on factors such as tool, distance, processing type, material properties, etc (Refer to Section II-E for details).

In addition to iterative calculation and cost calculation, the scheme selection process also involves logical operations. Based on the cost calculated, the logic operation determines the final choice, that is, the next processing step. For computers, the above-mentioned calculations can be implemented using operations such as iteration, nesting, and calling. The amount of calculation is measured using the computational complexity. It is almost impossible to implement the above operations manually. A computer intelligent algorithm is required to implement this complicated process.

### D. MODEL REFINING

Up to now, the manufacturing process of modern equipment is only fixed machined object one-time. So nearly 100 machining types are achieved, and more than ten thousand times of processing actions are involved. However, a manufacturing scheme must be considered and determined when the processing actions happen due to the batch machining caused by the large-scale production. The processing order affects greatly the manufacturing efficiency. Because of the complexity of machining activities, it is very necessary to abstract complex manufacturing processes into an expression model. In other words, the model is a simplification of the complex real system, and can be optimized as a whole. More specifically, the system model is the input, and the optimal node sequence of the system is the output of the optimization procedure. The node sequence generated represents the processing sequence, that is, the processing scheme.

The Graph Theory is investigated to model a manufacturing process. Usually, a machining task is divided into many separate manufacturing actions. Each action is regarded as a model node, and the coupling relationship of two machining actions is considered as an edge in the graph model. The convert cost between nodes is called weight. These nodes, edges and weights constitute a graph model representing the complex manufacturing process.

For example, a manufacturing task illustrated in Fig. 3 contains 4 machining types (represented by different shapes) and 16 processing actions (6 circles, 6 pentagonal, 3 triangles, and 1 rectangle). The processing action performed on the machine of the same type has the same cost. The position and dimension of each machining object are marked on the Fig. 3.

Although the processing sharp is cut and fixed, the manufacturing cost is often different. The processing order can be organized and arranged arbitrarily; different orders lead to different processing costs. Thus, it is important to arrange the processing order between machining actions in an optimal manner.

As a traditional method of deciding the processing order, the artificial method places and clamps the material and then replaces and fixes the tools every time when the processing step changes. Under this method, the processing efficiency and the machining accuracy are both low.

In our suggested method, only one positioning and clamping are needed, and hundreds of processing steps are executed continuously, leading to higher machining efficiency and accuracy. The key problem addressed in the proposed method is how to find the best processing scheme. This research presents an intelligent design method to solve the problem. The method finds an optimal processing scheme according to the discrete manufacturing feature. In particular, the Hamilton algorithm is utilized in this paper to handle the “combinatorial explosion” problem mentioned in the Introduction.
Note that to better explain the proposed method, simulated data is used. Specifically, the whole process includes the extraction of graph models in the manufacturing process, the formulation of parameter quantization functions needed to realize manufacturing, the analysis of complex calculation algorithms, the construction of model matrices, and the optimization to simplify matrices. The data in actual processing often cannot fully reflect the above-mentioned multiple requirements. To this end, simulated data are used.

E. MODEL QUANTIFICATION

In order to get the optimal ordering scheme that minimizes the manufacturing cost while meeting quality constraints or requirements, some quantitative methods are desired.

Because the physical unit of the system is fixed, and its performance is also fixed, the weight factors that depend on the model for quantification are thus valid and significant. Factors, such as the unit’s motion speed, the calculation frequency of the system, the hardness of the material are determined. Therefore, the quantitative indicators of the study can be determined.

Based on the manufacturing model, the manufacturing cost can be divided into two parts: the machining cost and the controlling cost. The machining cost is usually decided by the equipment performance; the higher the manufacturing efficiency, the lower the cost. For actions performed on the same type of machining, the cost is relatively fixed. The controlling cost involves multiple aspects, and its computation is more complex than the computation of the machining cost. Specifically, the controlling cost includes cost involved in fetching tool (shifting tool from tool lib and fixing tool in the designated position), taking material (moving the material to the designated position), and machining delay (material hard, thickness and sharp). As the same activity may be performed in different ways, thus taking different amounts of time. As a result, the controlling cost can be different.

Let $Y$ represent the total processing efficiency/cost of a task and $X=(X_1, X_2, \ldots X_n)$ be a vector of $n$ variables whose values affect the value of $Y$ based on (1).

$$Y = \sum_{i=1}^{n} f(X_i) \quad (1)$$

Assume each variable $X_i$ can be elaborated using $m$ sub variables denoted by $X_{ij}, j=1 \ldots m$. Thus, the formula (1) can be further written as (2).

$$Y = \sum_{i=1}^{n} \sum_{j=1}^{m} f(X_{ij}) \quad (2)$$

Similarly, if variable $X_{ij}$ is further segmented into $l$ sub variables, then equation (2) can be detailed as (3).

$$Y = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{l} f(X_{ijk}) \quad (3)$$

In this work $i$ and $j$ denote node indexes on a certain path $p$. The processing step from node $v_i$ to node $v_j$ involves fetching tool ($k=1$), taking materials ($k=2$) and machining ($k=3$). For these different activities (different $k$) the $f$ function in (3) can be different, which are detailed as follows.

Specifically, the efficiency/cost of fetching tool, denoted by $f_1(X_{ij1})$ is affected by factors including the system computation capacity $C$, the tool position $P_{ij1}$, processing number or quantity $Q$, and motion speeds $V_{ij1}$. Equation (4) shows its calculation, involving an iterative and nested calculation.

$$f_1(X_{ij1}) = \frac{\ln Q \cdot P_{ij1}}{C \cdot V_{ij1}} \quad (4)$$

$C$ in (4) is typically constant for an equipment; its value is mainly used for corrections. The position of the tool $P_{ij1}$ is the relative position. The quantity to be processed $Q$ refers to the total quantity of the next processing, which is usually obtained through an iterative operation, so its logarithm is taken here. The motion speed $V_{ij1}$ refers to the speed of fetching the tool.

The efficiency of taking material, denoted by $f_m(X_{ij2})$ is mainly affected by the distance $D_{ij2}$ between two processing units/nodes and the transmission speed $U_{ij2}$ of materials. The calculation method is given in expression (5).

$$f_m(X_{ij2}) = \frac{D_{ij2}}{U_{ij2}} \quad (5)$$

More specifically, the moving distance $D_{ij2}$ in (5) refers to the distance that the processed material is moved to the processing point. The moving speed $U_{ij2}$ refers to the speed at which the material to be processed is moved to the processing point.

The efficiency of the machining delay, denoted by $f_d(X_{ij3})$ is affected by the machining shape $\Phi_{ij3}$, the machining size
$S_{ij3}$, and the material plasticity $\delta_{ij3}$. The calculation of the efficiency associated with the machining delay is given by (6).

$$f_d(X_{ij3}) = \frac{\delta_{ij3}}{\Phi_{ij3} \ast S_{ij3}} \quad (6)$$

The machining shape $\Phi_{ij3}$ in (6) is usually inversely proportional to the machining cost. It represents the difficulty of the machining, quantified as the number of protruding corners. For example, circular machining is relatively simple compared to the machining of Polygons. For example, rectangular machining is relatively difficult. The processing size $S_{ij3}$ is inversely proportional to the processing difficulty, quantified as the relative area size. The plasticity $\delta_{ij3}$ of the processed material represents the hardness of the material, which is directly proportional to the processing cost. The value of $\delta_{ij3}$ can be determined based on the material properties.

The total cost/efficiency involved in the processing step from node $v_i$ to node $v_j$ is thus $f_i(X_{ij1}) + f_m(X_{ij2}) + f_d(X_{ij3})$.

Based on the above discussions, the intelligent design problem can be transformed into a mathematical problem, which finds an optimal task arrangement of $N$ tasks in terms of an optimal processing path, minimizing the processing efficiency $Y$ as formulated in (7) and (8).

$$Y(p) = \sum_{i=1}^{n} \sum_{j=1}^{m} \{f_i(X_{ij1}) + f_m(X_{ij2}) + f_d(X_{ij3})\} \quad (7)$$

$$Y_{\text{min}} = Y(p) \quad (8)$$

### III. OPTIMIZATION ALGORITHM

#### A. ALGORITHM

The manufacturing process is modeled as a graph model $G = (V, E, W)$, which is composed of $V$ the node set, $E$ the edge set, and $W$ the weight set. Each node in $V$ represents a different manufacturing activity. An edge in $E$ represents the relationship between two process activities. Each edge is associated with a weight representing the processing cost.

Consider a model with $n$ nodes. There are $C_n^2 = n! / 2! \times (n - 2)! = (n - 1) \times n / 2$ edges because there is an edge between any two nodes $v_i$ and $v_j$ according to the discrete manufacturing feature. In other words, the model for the manufacturing process is a fully linked network graph, also called the Hamilton graph. In this work, the Hamilton graph-based algorithms are studied to implement the intellectualization of the complex manufacturing process. The optimal solution of a manufacturing process is converted into finding a Hamilton circle where each node is passed exactly once on a route with the smallest size, i.e., the shortest path.

For a model with $n$ nodes, there are $n!$ manufacturing schemes according to the permutation and combination theory. As an illustration, there are 16! manufacturing schemes for a model with 16 nodes, as shown in Fig. 4.

The problem is to find the scheme with the shortest path in the model graph. According to the Hamilton circle principle, a computation expression (9) can be defined to find the shortest route with all nodes appearing only once.

$$(y_{ij} + y_{i(i+1)(j+1)}) < (y_{i(i+1)} + y_{j(j+1)}) \quad (9)$$

where $y_{ij}$ represents the weight of node $v_i$ to node $v_j$ ($1 < i+1 < j$), and others have similar meaning. Inequality (9) involves four edges, which form an original circle. According to inequality (9), either the left-hand side or the right-hand side is taken, which means two edges are kept, while the other two edges are canceled. This operation is repeated continuously until all nodes are searched. Ultimately, the Hamilton circle can be obtained.

In the algorithm implementation, a Hamilton graph model is represented using a data structure, corresponding to an $n \times n$ matrix. If the data (weight) at different rows and different columns in the matrix can be identified, and their sum is minimum among all the possible combinations, then the nodes corresponding to these data compose a Hamilton circle (an optimal scheme). The computation of the minimum value of node $v_i$ to node $v_j$ is performed using formula (8).

#### B. IMPROVED ALGORITHM

The algorithm described in Section 3.1 used to find the optimal path can be computationally expensive (with Exponential complexity). In this section, a simplified computation method is investigated to improve the computational efficiency. This method is to find divisor numbers and penalty numbers of the matrix, and then generate a new simplified matrix.

The divisor number calculation for the matrix contains computations of row divisor and column divisor. The row divisor of the $i^{th}$ row, denoted as $R(i)$, is obtained by taking the minimum number of the $i^{th}$ row from the numbers of this row in the matrix. The minimum value in the $i^{th}$ row is denoted as $R_i$. All the row vectors $R(i)$ form a new simplified matrix denoted as $M'$. The minimum value in the $j^{th}$ column of new matrix $M'$ is denoted as $R'_j$.
The column divisor of the \( j^{th} \) column denoted as \( R'(j) \), is obtained by taking the minimum number of the \( j^{th} \) column from the numbers of this column. The minimum value in \( R'(j) \) is denoted as \( R'_j \). The sum of these divisor numbers of all rows and columns is called the matrix divisor number, denoted as \( R \). The simplified matrix of matrix \( M \) (through traditional row and column simplifications) is denoted by \( M' \).

We also define the penalty number for each row and each column of the matrix. In matrix \( M \), the difference between the minimum number and the second smallest number of the \( i^{th} \) row is referred to as the penalty number of the \( i^{th} \) row, denoted as \( P(i) \). Similarly, the difference between the minimum number and the second smallest number of the \( j^{th} \) column is referred to as the penalty number of the \( j^{th} \) column, denoted as \( P'_j \). The penalty number of a row or a column represents a minimum value change in weight, which is used to choose a row or a column when building a Hamilton circle.

The matrix simplification process may be applied multiple times during the optimization. For a matrix after one simplification, it is referred as one-level simplified matrix; if two rounds of simplifications are applied, then a two-level simplified matrix is obtained, and so on.

### C. OPTIMIZATION BASED ON INEQUALITY

Traditional optimization methods often adopt iterative algorithms involving numerous iterations to find a desired result. Such optimization schemes are resource consuming, sometimes are hard to implement for practical engineering problems that have resource constraints. In this work we propose an optimization mechanism based on inequality (9). Specifically, the inequality is used as a controller to regulate and select a node variable each time from a node set to another node set. The sequence of these choices is recorded as a path set. The operation is continuously repeated, until this path set contains all nodes of the graph model of the manufacturing system. Nodes involved in the path are all different. Thus, the weight sum of the route set is minimum. As compared to the traditional iterative methods, the proposed optimization method has high computational efficiency.

Consider the actual manufacturing process. A variable space \( X \) is used to save the nodes (i.e., \( X \) is also a node set). \( X_1, X_2, \ldots, X_n \) denote sub-variable spaces of \( X \). \( x_i \) is a variable in \( X_i, i = 1 \ldots n \). Variable \( Y_{ij} \) in (10) is the cost between variable spaces \( X_i \) and \( X_j \).

\[
Y_{ij} = f(X_{ij}) = f(X_i, X_j), \quad i, j = 1 \ldots n \tag{10}
\]

Expression (10) is used to search the transfer point/node in the case of many transfer points existing. These transfer nodes and their ordering form a route set. There are many different distances between node variable spaces. The connected node sequence with the minimum distance (11) is acquired as the path node set, forming an optimal path. Thus, these nodes in the path set are the transfer points.

\[
y_{ij} = \min f(X_{ij}) \tag{11}
\]

The whole path length is obtained according to the following expression (12). The solution route is namely the shortest path and the optimal path, and the path weight means the smallest cost of the manufacturing process.

\[
Y = \sum_{i=1,j=1}^{n} y_{ij} \tag{12}
\]

### IV. CASE STUDIES

The example shown in Fig. 3 is used to illustrate the intelligent design process based on the Hamilton algorithm. The process modeling, weight calculation, matrix construction, matrix simplification, optimization solution are explained in the following subsections.

#### A. PROCESS MODELING

As illustrated in Fig. 3, the model contains 16 processing activities, modeled by 16 nodes in the graph model of the manufacturing process. These nodes are numbered from 1 to 16 and classified into 4 types according to the processing action classification (mentioned in Section II. B), as shown in Table 1.

| num | nodes | types | num | nodes | types |
|-----|-------|-------|-----|-------|-------|
| 1   | a1    | circle| 9   | b3    | Triangle|
| 2   | a2    | circle| 10  | c1    | pentagon|
| 3   | a3    | circle| 11  | c2    | pentagon|
| 4   | a4    | circle| 12  | c3    | pentagon|
| 5   | a5    | circle| 13  | c4    | pentagon|
| 6   | a6    | circle| 14  | c5    | pentagon|
| 7   | b1    | Triangle| 15  | c6    | pentagon|
| 8   | b2    | Triangle| 16  | d1    | rectangle|

#### B. WEIGHT CALCULATION

To implement the quantitative calculation, a \( 16 \times 16 \) matrix \( M \) is constructed to save nodes, edges, and weights of the graph model. The weight value stored in position \( (i, j) \) of the matrix is the total cost from node \( v_i \) to node \( v_j \), which is the summation of three costs determined based on formula (4) - (6), respectively.

For example, consider the evaluation of weight/cost from node \( v_1 \) to node \( v_2 \). We first consider the cost of fetching tool \( C \). Assume the machining servo motor speed \( V \) is 1600 mm/s. The computation capacity \( C \) is 1 in research case. The value of \( P \) is 30mm, and \( N \) is 16. According to (4), we have

\[
f_t(x_{121}) = \frac{\ln N^2 \ast P_{121}}{C \ast V_{121}} = \frac{\ln 16^2 \ast 60}{1 \ast 1600} = 0.00208
\]

The efficiency/cost \( f_{S}(x_{122}) \) of material transmission is calculated below according to formula (5). Here, the motion
speed $U$ is 1800 mm/s, distance $D$ is 18mm.

$$f_m(x_{122}) = \frac{D_{122}}{U_{122}} = \frac{18}{1800} = 0.00100$$

The machining efficiency $f_d(x_{123})$ is evaluated according to formula (6). The plastic deformation of the material is depended on its own fluidity, and fetches $\delta = 100\%$. The steel sharp $\Phi$ takes 1, and size $S$ is 10mm$ \times $10mm.

$$f_d(x_{123}) = \frac{\delta_{123}}{\Phi_{123} \cdot S_{123}} = \frac{1}{1 \cdot 100} = 0.01000$$

Finally, the total cost is

$$f_t(x_{121}) + f_m(x_{122}) + f_d(x_{123}) = 0.01308$$

The actual weight value stored in position (1, 2) of the matrix is 13, after a normalization treatment of $Y_{12}$.

In a similar way, costs of all motion transfers are calculated and stored as the weight value in each position of the matrix. The weight is simply defined as infinity ($\infty$) if no transfer can happen between two nodes.

### TABLE 2. The first order simplified matrix $M'$.

|   | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | $R_i$ | $P_i$ |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|-----|-----|
| 1 | $\infty$ | 0 | 0 | 2 | 7 | 12 | 12 | 13 | 14 | 15 | 16 | 16 | 17 | 18 | 18 | 14 | 13 | 0   |
| 2 | 0 | $\infty$ | 2 | 0 | 7 | 10 | 7 | 8 | 9 | 13 | 14 | 15 | 15 | 16 | 17 | 12 | 13 | 0   |
| 3 | 0 | 2 | $\infty$ | 0 | 3 | 10 | 14 | 13 | 12 | 12 | 13 | 14 | 13 | 15 | 14 | 15 | 14 | 13 | 0   |
| 4 | 2 | 0 | $\infty$ | 3 | 8 | 9 | 8 | 7 | 8 | 9 | 10 | 9 | 10 | 12 | 12 | 13 | 13 | 0   |
| 5 | 4 | 4 | 0 | 0 | $\infty$ | 0 | 12 | 10 | 8 | 2 | 3 | 4 | 1 | 2 | 3 | 7 | 20 | 0   |
| 6 | 9 | 7 | 7 | 5 | 0 | $\infty$ | 11 | 9 | 7 | 3 | 2 | 1 | 2 | 1 | 0 | 4 | 20 | 0   |
| 7 | 15 | 10 | 17 | 12 | 18 | 7 | $\infty$ | 0 | 4 | 17 | 16 | 16 | 18 | 17 | 17 | 7 | 10 | 4   |
| 8 | 16 | 11 | 16 | 11 | 16 | 15 | 0 | $\infty$ | 0 | 14 | 13 | 13 | 17 | 16 | 16 | 7 | 10 | 0   |
| 9 | 17 | 12 | 15 | 10 | 14 | 13 | 14 | 0 | $\infty$ | 13 | 12 | 12 | 14 | 13 | 13 | 8 | 10 | 8   |
| 10 | 18 | 16 | 15 | 11 | 8 | 9 | 17 | 14 | 13 | $\infty$ | 0 | 3 | 0 | 1 | 4 | 12 | 10 | 0   |
| 11 | 19 | 17 | 16 | 12 | 9 | 8 | 16 | 13 | 12 | 0 | $\infty$ | 0 | 1 | 0 | 1 | 11 | 10 | 0   |
| 12 | 20 | 18 | 17 | 13 | 10 | 7 | 16 | 13 | 12 | 3 | 0 | $\infty$ | 4 | 1 | 0 | 0 | 0 | 10 | 0   |
| 13 | 19 | 18 | 16 | 12 | 7 | 8 | 18 | 17 | 14 | 0 | 1 | 4 | $\infty$ | 0 | 3 | 3 | 10 | 0   |
| 14 | 20 | 19 | 17 | 13 | 8 | 7 | 17 | 16 | 1 | 0 | 1 | 0 | 3 | $\infty$ | 0 | 12 | 10 | 0   |
| 15 | 21 | 20 | 18 | 15 | 9 | 6 | 17 | 16 | 13 | 4 | 1 | 0 | 3 | 0 | $\infty$ | 11 | 10 | 0   |
| 16 | 10 | 8 | 10 | 8 | 6 | 3 | 0 | 0 | 1 | 5 | 4 | 3 | 6 | 5 | 4 | $\infty$ | 21 | 0   |
| $R_j$ | 0 | 0 | 0 | 0 | 4 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0   |
| $P_j$ | 0 | 0 | 0 | 0 | 3 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0   |

### C. MATRIX CONSTRUCTION

The relationship between any two nodes in the graph model $G$ is examined, and the weight associated with edge representing each relationship is calculated using the method illustrated in Section IV. B.

The final matrix $M$ for the example process is given in (13), as shown at the bottom of the next page. Next, the algorithm is illustrated for obtaining an optimal path in terms of a Hamilton circle.

### D. OPTIMIZATION SOLUTION

The optimization process involves the following four steps.

**Step 1** Simplifying matrix $M$. First, the penalty number and divisor number of the matrix $M$ is computed, as summarized in Table 2. The simplified matrix is called the first order simplified matrix, and denoted as $M'$. The nodes in any row of Table 2 forms a variable space, and any two spaces have many routes according to formula (10).

**Step 2** The penalty numbers corresponding to zero values are obtained from matrix $M'$, i.e., Table 2. Their positions in the matrix are recorded according to the decreasing order of the corresponding row or column’s penalty number, as summarized in Table 3.
Based on formula (10) and Table 3, the edges forming feasible routes obtained as follows:

\((3, 1), (1, 2), (2, 4); (5, 6), (6, 15), (8, 7), (9, 8); (10, 11), (10, 13), (11, 12), (12, 16), (13, 14), (14, 15), (16, 7)\).

These edges compose into two paths without any cross: 4-2-1-3, and 5-6-15-14-13-10-11-12-16-7-8-9.

1. **Step 3** Because the two paths do not cross, the first order matrix and its corresponding submatrix are restructured.
2. **Step 4** The divisor number and the penalty number of the second order simplified matrix are calculated according to step 1 to step 3, and feasible edges \((3, 5)\) and \((4, 9)\) are obtained according to formula (10). Now a feasible route with edges generated in this step and step 2: 4-2-1-3, 3-5-6-15-14-13-10-11-12-16-7-8-9. Based on expression (11), this feasible route forms a Hamilton loop with total weight/cost 241: 4-2-1-3-5-6-15-14-13-10-11-12-16-7-8-9-4.

Fig. 5 illustrates the optimal path based on Hamilton loop generated, leading to the optimal manufacturing scheme: \(a_1-a_2-a_3\) …

**V. ALGORITHM COMPLEXITY ANALYSIS AND VERIFICATION**

In this section, the advantages of the FCNC manufacturing process optimization method based on the Hamilton algorithm are investigated using three indexes of computational complexity, computational accuracy and applicability.

To show the superiority of the proposed algorithm, a comparative study is performed between the proposed Hamilton (full connection graph) algorithm and two representative methods—the Hopfield neural network (HNN) algorithm, and the Evolutionary Ant Colony System (EACS) algorithm.

The HNN algorithm has a feedback characteristic similar to the Hamilton algorithm. It has a strong computing capability. Fig. 6 demonstrates its connection mode/structure. The HNN network model is mainly used to solve the problem of energy system stability. Through self-learning and associative memory function, the system will develop towards a stable state.
The energy function $E$ of the HNN optimization [39] can be expressed as (14).

$$
E = \frac{A}{2} \sum_{x} \sum_{i \neq j} v_{yi}v_{xj} + \frac{B}{2} \sum_{i} \sum_{y \neq x} v_{xi}v_{yi} \\
+ \frac{C}{2} (\sum_{i} v_{xi} - n)^2 \\
+ \frac{D}{2} \sum_{x} \sum_{i \neq x} \sum_{y \neq x} d_{xy} v_{xi} (v_{yi,i+1} + v_{yi,i-1})
$$

(14)

In (14), $v_{xi}, v_{yi}$ represents the connection of nodes $v_{xi}$ and $v_{yi}$. If there is a connection, it is valued 1, otherwise it is valued 0. A, B, C and D are network coefficients. $d_{xy}$ is the distance between points $x$ and $y$.

When $E$ reaches the minimum, the connected network composed of nodes $v_{xi}$ and $v_{yi}$ represents the best optimization path.

For the optimal solution of the HNN algorithm with $n$ nodes, $n^2$ neurons are desired for computations. The complexity is calculated according to equation (14), which is $O(n!)$. The complexity of the algorithm has a great influence on the stability of the algorithm. A simple algorithm typically has good stability; a complex algorithm tends to have worse stability.

The EACS algorithm regards ants as evolutionary individuals, introduces the processes of individual coding, selection, crossover and mutation, and then controls algorithm parameters to achieve the optimal selection. EACS does not need a large number of data experiments to get the optimization results. This algorithm can be used in engineering practice. Next the complexity of the EACS algorithm is discussed with an illustration of the optimization of the FCNC machining process.

Consider $n$ machining positions. The Euclidean distance between the position vector $X$ and the movement of the processing unit can be computed as (15) [40].

$$
f_1 = \min_{i=2}^{n-1} \|X_i - X_{i+1}\| + \|X_1 - X_n\| \tag{15}
$$

The complexity of the EACS algorithm in the case of $n$ nodes is decided by the minimum value of its calculation process. The computational complexity of the EACS algorithm is $O(n^3)$.

The complexity of the proposed Hamilton algorithm can be evaluated based on (8). The complexity of step 1 is $O(n \times n)$, step 2 is $O(n^2)$, step 3 is $O(nm)$, and step 4 is $O(m'')$. Because of $m \ll n$ in each step, the overall complexity of the Hamilton optimization algorithm is $O(n^2)$.

Table 4 summarizes the performance comparison of the three algorithms (the Hamilton Graph, the HNN, and the EACS).

For the optimal solution of the HNN algorithm with $n$ nodes, $n^2$ neurons are desired for computations. The complexity is calculated according to equation (14), which is $O(n!)$. The complexity of the algorithm has a great influence on the stability of the algorithm. A simple algorithm typically has good stability; a complex algorithm tends to have worse stability.

The EACS algorithm regards ants as evolutionary individuals, introduces the processes of individual coding, selection, crossover and mutation, and then controls algorithm parameters to achieve the optimal selection. EACS does not need a large number of data experiments to get the optimization results. This algorithm can be used in engineering practice. Next the complexity of the EACS algorithm is discussed with an illustration of the optimization of the FCNC machining process.

Consider $n$ machining positions. The Euclidean distance between the position vector $X$ and the movement of the processing unit can be computed as (15) [40].

$$
f_1 = \min_{i=2}^{n-1} \|X_i - X_{i+1}\| + \|X_1 - X_n\| \tag{15}
$$

The complexity of the EACS algorithm in the case of $n$ nodes is decided by the minimum value of its calculation process. The computational complexity of the EACS algorithm is $O(n^3)$.

The complexity of the proposed Hamilton algorithm can be evaluated based on (8). The complexity of step 1 is $O(n \times n)$, step 2 is $O(n^2)$, step 3 is $O(nm)$, and step 4 is $O(m'')$. Because of $m \ll n$ in each step, the overall complexity of the Hamilton optimization algorithm is $O(n^2)$.

Table 4 summarizes the performance comparison of the three algorithms (the Hamilton Graph, the HNN, and the EACS).

In terms of the computational complexity, since $n! > n^3 > n^2$ when the number of nodes exceeds 5, the complexity of the proposed Hamilton algorithm is the lowest among the three algorithms compared.

In terms of accuracy, the optimizations using the HNN or EACS algorithm can get an optimal or suboptimal solution by imitating the biological evolution, while the Hamilton algorithm obtains the precise solution by the mathematics method. For the example system, the optimization results using the three optimization methods are the same, verifying the correctness of the proposed Hamilton method.

In terms of applicability, the HNN method has limited applications because of its high computing complexity and difficulty in determining the network coefficients used in the optimization. The EACS has better applicability than HNN, but its application is still limited due to the high computational complexity and coefficient adjustments during the optimization. The proposed Hamilton algorithm has the best applicability among the three methods compared due to its high computational efficiency.

In the example, four kinds of machining tools are used, and the consumption of tool conversion is large. From the optimization sequence obtained using the Hamilton algorithm, it can be seen that the same tool is in the continuous (adjacent) machining process, (4-2-1-3-5-6) - (15-14-13-10-11-12) - (16) - (7-8-9). This result not only meets the algorithm requirements, but also fits the actual processing situation.

**TABLE 4. Comparison of indexes of optimization algorithm.**

| Algorithms      | Complexity | Accuracy | Applicability |
|-----------------|------------|----------|---------------|
| HNN             | $O(n!)$    | optimal, or suboptimal | limited     |
| EACS            | $O(n^3)$   | optimal, or suboptimal | ordinary   |
| Hamilton-Graph  | $O(n^2)$   | precise  | broad         |

**FIGURE 5.** The optimal route.
In summary, it is effective and feasible to optimize the FCNC manufacturing process using the proposed Hamilton algorithm.

VI. CONCLUSION

In this paper, a graph theory-based efficient design method is proposed for the manufacturing process. The quantitative calculation models presented are very helpful for comprehending equipment intelligentization, and solving complex manufacturing scheme design problems in an efficient manner. As demonstrated by the example, the suggested method can improve the equipment utilization, and promote the equipment upgrade. The proposed method is feasible and can be implemented for real-world engineering applications.

The optimization design method based on the Hamilton graph algorithm for the complex manufacturing processes only gives the theoretical calculation and verification based on simulated data. In the future, we will consider the practical engineering of the manufacturing process optimization based on a visual design method, improving the manufacturing capability of the equipment. In addition, we are interested in performing a quantitative performance comparison between the Hamilton graph-based method and HNN and EACS in the future. The optimization method based on enumeration proposed in [41] can improve the computing efficiency of large-scale matrices by 300%. We are interested in exploring this method for the determination of intelligent manufacturing scheme.

ACKNOWLEDGMENT

The authors would like to thank Engineer G. Zhang and his colleague for their suggestions and useful discussions. Zhong Han would like to extend especial thanks to his students for their support in carrying out experiments and performing measurements. They would also like to thank anonymous referees for their constructive comments that help improve the quality of this article.

REFERENCES

[1] M. Khakifirooz, M. Fathi, and K. Wu, “Development of smart semiconductor manufacturing: Operations research and data science perspectives,” IEEE Access, vol. 7, pp. 108419–108430, 2019.
[2] C. Gu, Z. Li, N. Wu, M. Khalgu, T. Qu, and A. Al-Ahmari, “Improved multi-step look-ahead control policies for automated manufacturing systems,” IEEE Access, vol. 6, pp. 68824–68838, 2018.
[3] J. Lee, B. Bagheri, and H.-A. Kao, “A cyber-physical systems architecture for industry 4.0-based manufacturing systems,” Manuf. Lett., vol. 3, pp. 18–23, Jan. 2015.
[4] R. Jardim-Goncalves, D. Romero, and A. Grilo, “Factories of the future: Challenges and leading innovations in intelligent manufacturing,” Int. J. Comput. Integ. Manuf., vol. 30, no. 1, pp. 1–3, 2017. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/0951192X.2016.1258120
[5] R. Jardim-Goncalves, D. Romero, and A. Grilo, “Factories of the future: Challenges and leading innovations in intelligent manufacturing,” Int. J. Comput. Integ. Manuf., vol. 30, no. 1, pp. 4–14, 2017.
[6] M. Khakifirooz, C.-F. Chien, and M. Fathi, “Compensating misalignment using dynamic random-effect control system: A case of high-mixed wafer fabrication,” IEEE Trans. Autom. Sci. Eng., vol. 16, no. 4, pp. 1788–1799, Oct. 2019.
[7] F. Qiao, Y.-M. Ma, L. Li, and H.-X. Yu, “A Petri net and extended genetic algorithm combined scheduling method for wafer fabrication,” IEEE Trans. Autom. Sci. Eng., vol. 10, no. 1, pp. 197–204, Jan. 2013.
[8] C. Bailey, E. Aguilar, D. Espalin, J. Motta, A. Fernandez, M. A. Perez, C. Dibiasio, D. Prynputniewicz, E. Macdonald, and R. B. Wicker, “Augmenting computer-aided design software with multi-functional capabilities to automate multi-process additive manufacturing,” IEEE Access, vol. 6, pp. 1985–1994, 2018.
[9] G. Ming, X. Yangsheng, and D. Ruxu, “An intelligent online monitoring and diagnostic system for manufacturing automation,” IEEE Trans. Autom. Sci. Eng., vol. 5, no. 1, pp. 127–139, Jan. 2008.
[10] E. C. Balta, Y. Lin, K. Barton, D. M. Tilbury, and Z. M. Mao, “Production as a service: A digital manufacturing framework for optimizing utilization,” IEEE Trans. Autom. Sci. Eng., vol. 15, no. 4, pp. 1483–1493, Oct. 2018.
[11] Y. Wu, H.-T. Zhang, T. Huang, G. Ren, and H. Ding, “Robust chatter mitigation control for low radial immersion machining processes,” IEEE Trans. Autom. Sci. Eng., vol. 15, no. 4, pp. 1972–1979, Oct. 2018.
[12] A. M. Farid, “Measures of reconfigurability and its key characteristics in intelligent manufacturing systems,” J. Intell. Manuf., vol. 28, no. 2, pp. 353–369, Feb. 2017.

[13] O. Stocklosa, T. Borangiu, S. Raileanu, O. Morariu, and C. Morariu, “Visual guidance of robots integrated in intelligent manufacturing,” Univ. Politehnica Bucharest Sci. Bulletin, C, Elect. Eng. Comput. Sci., vol. 77, no. 2, pp. 65–80, 2015.

[14] A. Giret, D. Trentesaux, and V. Prabhu, “Sustainability in manufacturing operations scheduling: A state of the art review,” J. Manuf. Syst., vol. 37, pp. 126–140, Oct. 2015.

[15] C. Afteni, R. D. de Jos, and G. Frumusana, “A review on optimization of manufacturing process performance,” Int. J. Model. Optim., vol. 7, no. 3, pp. 139–144, 2017.

[16] R. de Matta, “Scheduling a manufacturing process with restrictions on resource availability,” Int. J. Prod. Res., vol. 56, no. 19, pp. 6412–6429, Oct. 2018.

[17] I. Ion, G. Stamatescu, and C. Dinu, “Intelligent sensing and motion control in robotic manufacturing systems,” Appl. Mech. Mater., vol. 762, pp. 299–304, May 2015.

[18] L. Sudha, R. Dillibabu, S. S. Srinivas, and A. Annamalai, “Optimization of process parameters in feed manufacturing using artificial neural network,” Comput. Electron. Agricul., vol. 120, pp. 1–6, Jan. 2016.

[19] Q. Li, L.-Y. Xie, H.-Y. Li, F.-M. Zhang, and J.-X. Song, “Optimization Priority Analysis of Precision Gear Manufacturing Process Based on AHP Fuzzy Comprehensive Evaluation Method,” Acta Armamentarii, vol. 38, no. 4, pp. 750–757, 2017.

[20] H. G. Lenmu, “Current status and challenges of using geometric tolerance information in intelligent manufacturing systems,” Adv. Manuf., vol. 2, no. 1, pp. 13–21, Mar. 2014.

[21] M. Marques, C. Agostinho, G. Zacharewicz, and R. Jardim-Gonçalves, “Decentralized decision support for intelligent manufacturing in industry 4.0,” J. Ambient Intell. Smart Environ., vol. 9, no. 3, pp. 299–313, Apr. 2017.

[22] A. Giret, E. Garcia, and V. Botti, “An engineering framework for service-oriented intelligent manufacturing systems,” Comput. Ind., vol. 81, pp. 116–127, Sep. 2016.

[23] G. Wang, X. Shang, Y. Yan, J. K. Allen, and F. Mistree, “A tree-based decision method for the configuration design of reconfigurable machine tools,” J. Manuf. Syst., vol. 49, pp. 143–162, Oct. 2018.

[24] H. B. Perez, I. L. Juarez, P. C. G. Alanis, R. R. Cabrera, and A. D. Chavesti, “Reconfiguration distributed objects in a intelligent manufacturing cell,” IEEE Latin Amer. Trans., vol. 14, no. 1, pp. 136–146, Jan. 2016.

[25] A. Sanz-Lobera, E. Gómez, J. Pérez, and L. Sevilla, “A proposal of cost-tolerance models directly collected from the manufacturing process,” Int. J. Prod. Res., vol. 54, no. 15, pp. 4584–4598, Aug. 2016.

[26] N. Gardan, A. Schneider, and J. Gardan, “Material and process characterization for coupling topological optimization to additive manufacturing,” Comput.-Aided Des., vol. 13, no. 1, pp. 39–49, Jan. 2016.

[27] D. Sun, Y. Chen, M. A. El-Meligy, M. A. F. Sharaf, N. Wu, and Z. Li, “On algebraic identification of critical states for deadlock control in automated manufacturing systems modeled with Petri nets,” IEEE Access, vol. 7, pp. 121332–121349, 2019.

[28] A. Jamali, E. Khaleghi, I. Gholaminezhad, N. Nariman-Zadeh, B. Gholaminia, and A. Jamal-Omid, “Multi-objective genetic programming approach for robust modeling of complex manufacturing processes having probabilistic uncertainty in experimental data,” J. Intell. Manuf., vol. 28, no. 1, pp. 149–163, 2014.

[29] A. Casali and C. Ernst, “Discovering correlated parameters in semiconductor manufacturing processes: A data mining approach,” IEEE Trans. Semicond. Manuf., vol. 25, no. 1, pp. 118–127, Feb. 2012.

[30] M. Samadani, S. Behbahani, and C. Nataraj, “A reliability-based manufacturing process planning method for the components of a complex mechatronic system,” Appl. Math. Model., vol. 37, no. 24, pp. 9829–9845, Dec. 2013.

[31] R. Siegbert, N. Vesilad, M. Frings, F. Schmidt, S. Elgeti, H. Sauerland, M. Behr, C. Windeck, C. Hopmann, Y. Queudeville, U. Vroemen, and A. Bührig-Polaczek, “Individualized production in die-based manufacturing processes using numerical optimization,” Int. J. Adv. Manuf. Technol., vol. 80, nos. 5–8, pp. 851–858, Sep. 2015.

[32] T. Henriksen, S. Lo, and U. Knaack, “The impact of a new mould system as part of a novel manufacturing process for complex geometry thin-walled GFRC,” Architectural Eng. Des. Manage., vol. 12, no. 3, pp. 231–249, May 2016.

[33] A. Veira, K. Khan, and I. Farrell, “A strategy for achieving manufacturing statistical process control within a highly complex aerospace environment,” Int. J. Adv. Manuf. Technol., vol. 77, no. 2, pp. 65–80, 2015.

[34] C. Pach, T. Berger, T. Bonte, and D. Trentesaux, “ORCA-FMS: A dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling,” Comput. Ind., vol. 65, no. 4, pp. 706–720, May 2014.

[35] R. A. Valiev, A. K. Khairullin, and V. G. Shibakov, “Automated design systems for manufacturing processes,” Russian Eng. Res., vol. 35, no. 9, pp. 662–665, 2015.

[36] T. Borangiu, S. Raileanu, D. Trentesaux, T. Berger, and I. Jacob, “Distributed manufacturing control with extended CNP interaction of intelligent products,” J. Manuf. Syst., vol. 25, no. 5, pp. 1065–1075, Oct. 2014.

[37] K.-T. Zuo, L.-P. Chen, Y.-Q. Zhang, and J. Yang, “Manufacturing-and machining-based topology optimization,” Int. J. Adv. Manuf. Technol., vol. 27, nos. 5–6, pp. 531–536, Jan. 2006.

[38] J. Ning, V. Nguyen, Y. Huang, K. T. Hartwig, and S. Y. Liang, “Inverse determination of Johnson–Cook model constants of ultra-fine-grained titanium based on chip formation model and iterative gradient search,” Int. J. Adv. Manuf. Technol., vol. 99, nos. 5–8, pp. 1131–1140, Nov. 2018.

[39] J. J. Hopfield and D. W. Tank, “‘Neural’ computation of decision in optimization problem,” Biol. Cybern., vol. 52, pp. 141–152, 1985.

[40] Z. Tao and R. Xiao, “Co-evolutionary ant colony algorithm and its application to multi-objective optimization problems,” (in Chinese), Pattern Recognit. Artif. Intell., vol. 18, no. 5, pp. 588–595, 2005.

[41] M. Gorawski and M. Lorek, “Efficient processing of large data structures on GPUs: Enumeration scheme based optimisation,” Int. J. Parallel Prog., vol. 46, no. 6, pp. 1063–1093, Dec. 2018.
LIUDONG XING (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, in 2002. She is currently a Professor with the Department of Electrical and Computer Engineering, University of Massachusetts (UMass) Dartmouth, USA. She has published two books titled *Binary Decision Diagrams and Extensions for System Reliability Analysis and Dynamic System Reliability: Modeling and Analysis of Dynamic and Dependent Behaviors*. Her current research interests include reliability and resilience modeling, and analysis and optimization of complex systems and networks. Prof. Xing is a Fellow of The International Society of Engineering Asset Management. She was a recipient of the 2014 Leo M. Sullivan Teacher of the Year Award, the 2010 Scholar of the Year Award, and the 2011 Outstanding Women Award of UMass Dartmouth. She was a recipient of the 2018 IEEE Region 1 Outstanding Teaching in an IEEE Area of Interest (University or College) Award, the 2015 ChangJiang Scholar Award by the Ministry of Education of China, and the 2007 IEEE Region 1 Technological Innovation (Academic) Award. She was a co-recipient of the Best (Student) Paper Award at several conferences and journals. She is also an Associate Editor or Editorial Board Member of multiple journals, including *Reliability Engineering & System Safety*, the IEEE *INTERNET OF THINGS JOURNAL*, and the *International Journal of Systems Science*.

LITING TIAN received the B.S. degree in electrical engineering and the M.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2004 and 2007, respectively, where she is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. In 2016, she joined the Energy Internet Institute, Tsinghua University. Her research interests are in multienergy system planning, distribution network planning, and system reliability and safety. Her major is the energy Internet, and energy equipment.

* * *