Chapter 2
Recent Advances of Intelligent Optimization Algorithm in Manufacturing

Due to its good versatility and independence, intelligent optimization algorithm has largely shortened the time of decision-making in large-scale optimization problems of manufacture. However, lower searching time often conflicts with the searching accuracy in most cases. To improve the problem solving capability, research in intelligent optimization algorithm based on different domain characteristics never stopped. From the view of manufacturing, this chapter classified and comprehensively analyzed all kinds of manufacturing optimization problems and their general methods, illustrated the application features and challenges of intelligent optimization algorithm in manufacturing, and summarized the development needs and trends of intelligent optimization algorithm in the field of manufacturing system.

2.1 Introduction

First of all, the application process of intelligent optimization algorithm in manufacturing engineering consists of five main parts, as shown in Fig. 2.1, problem modeling, variable encoding, operator design, simulation and algorithm implementation. Differs from pure algorithm design, the most critical part of algorithm application is problem modeling and variable encoding. Then the design of operators in algorithm depends largely on the specific environment and coding ways.

Problem modeling: The core of modeling is using variables and formulas to concisely and comprehensively express the three main elements of problem—variables, objectives and constraints—according to the environment and requirement. Moreover, the priori knowledge, environmental parameters and the relationship between variables should be given in concise mathematical expression.

F. Tao et al., *Configurable Intelligent Optimization Algorithm*,
Springer Series in Advanced Manufacturing,
DOI: 10.1007/978-3-319-08840-2_2

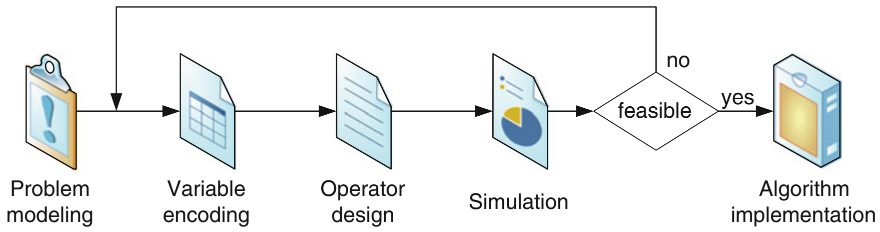

Fig. 2.1 The application process of intelligent optimization algorithm

Variable encoding: Encoding scheme is the link between problem and intelligent optimization algorithm. It is the basis for operators in algorithm to search in the solution space of problem. Different encoding schemes have different levels of randomness and then make the algorithms searching with different capability.

Operator design: With population-based iteration, operators, such as crossover, mutation and so on, need to be selected and designed according to the above encoding scheme. It decides the evolutionary direction of population and the whole searching way of algorithm. Different kinds of operators have different ability of exploration and exploitation and suitable for different sorts of problems. Thus in this step, we should especially focus on the balance of the two ability in the algorithm.

Simulation: Because of the randomness of intelligent optimization algorithm, simulation is the most effective way to verify the algorithm performance with theoretical analysis. Moreover, parameters need to be tuned based on several experiments. If the expected performance is reached, the algorithm can be adopted and applied; if not, we should return and reanalyze the encoding scheme or the operators for adjusting the specific problem.

Algorithm implementation: After the design and simulation, the algorithm can then be developed in practical systems for application.

Based on such a unified process, intelligent optimization algorithm is applied almost in everywhere. It can be seen that there are mainly three types of application objects in manufacturing field: management of manufacturing process, control/simulation for manufacturing system and product/element design and analysis, as shown in Fig. 2.2.

- Management of manufacturing process: It covers the continuous process modeling and discrete workflow management of design, machining and transportation in production line. It is central line for the whole life cycle of manufacturing. Thus it can be called as process optimization.
- Control/Simulation for manufacturing system: It includes the design of manufacturing control system, manufacturing simulation and supervision of production line. Only high efficient control and simulation will guarantee the efficient operation of the whole manufacturing system. It is a kind of system optimization.

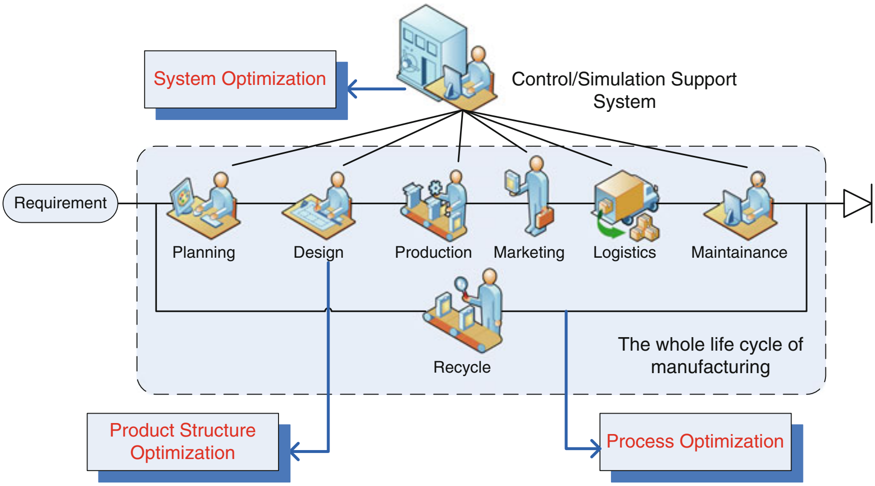

Fig. 2.2 Three main optimization objects in manufacturing

- Product/element design and analysis: It contains the structure design and modeling and finite element analysis of product. It is the core object of manufacturing. Optimization in this category is known as structure optimization as well.

Problems in these three objects have their own particular characteristics and are all along with high complexity. Though a plenty of intelligent optimization algorithms are designed for them, there are still various degrees of difficulties and challenges in the optimization for these three types of objects. On determining how to choose the most beneficial algorithms for different application objects in accordance with problem characteristics, and how to apply and incorporate those algorithms with general optimization methods in practical systems, further discussion is presented next.

2.2 Classification of Optimization Problems in Manufacturing

Problems in manufacturing include single-objective and multi-objective ones. In the light of the attributes of decision variable, problems can be simply divided into continuous numerical optimization and discrete combinatorial optimization as well. From the perspectives of both optimization targets and decision variables, we primarily divide the problems into five categories: numerical function optimization, parameter optimization, detection and classification, combinatorial scheduling, and multi-disciplinary optimization.

2.2.1 Numerical Function Optimization

Numerical function optimization refers to searching for optimal solutions of nonlinear multivariate complex equations. Typical problems in this category include function optimizations in modeling of manufacturing process and complex finite element analysis for product structure design. The variables are usually continuous and the objectives are multi-modal numerical functions.

Generally, numerical function optimization problems often appear in manufacturing process optimization and product structure optimization. The solutions are primarily the suitable values of global characteristics such as cutting or milling speed, and feed rate, process loads and structure length of product. Complex environment always bring about multiple relative parameters and constraints in the problem model. Thus it has the characteristics of large solution space, dispersive and narrow feasible solution regions, complex objective functions and high dimension of variables. Take the structure design of manufacturing part as an example, the decision variables are generally structure sizes, and the objectives are maximizing the key loads and minimizing resource consumption. With nonlinear relationship among stress and strain of the part, material consumptions and the structure sizes, differentiation and integration are both involved in the objective functions, which make the multimodal functions difficult to solve.

In continuous space, when most of the multivariate functions are linear, mathematical programming is commonly used in solving equations. When functions are complex but have small solution space, software such as ANSYS and CAD are often used to simulate. We may find the peak value by means of mathematical modeling and programming. However, with large-scale solution space and nonlinear multivariate functions, when most classical method requires much longer solution time, intelligent optimization algorithm can come in handy. Additionally, because of the uncertainties in the model parameters selection and the machining index tuning, intelligent optimization algorithm with invariance and independence can better adapt to solve these problems. Thus, in recent years, genetic algorithm and particle swarm algorithm, which are suitable for continuous numerical optimization, are applied in many kinds of structure optimization and manufacturing process optimization. And these intelligent optimization algorithms are usually combined with classical deterministic algorithms to optimize the model in two steps or optimize the model under the guidance of classical deterministic algorithms to improve the solutions. Reference related to numerical function optimization in manufacturing can be found in [1–9].

2.2.2 Parameter Optimization

Parameter optimization generally refers to the selection of optimal empirical parameters in complex manufacturing system or process control optimization. In manufacturing system, most parameters such as material and machining properties have big influence on the manufacturing process and systems. With large uncertainty and complexity, it is hard to build theoretical model to calculating the optimum value of these parameters for different situations. Thus they are mostly extracted and solved independently with nondeterministic algorithms.

Parameter optimization often exists in manufacturing process optimization and system optimization. Differs from numerical function optimization, parameter optimization is the optimization of one specific part or local key point, though the global environmental factors of system or process are considered. The parameters involved are highly dynamic and context-relative. For example, in the process of manufacturing such as casting or milling, the variables to be solved are machining force indexes, control time interval, load variance (upper bound and lower bound), etc. And the aim is the lowest loss, the highest manufacturing speed and the highest machining quality. In the process of optimization, objective functions are usually not able to be set out, and the real-time demand of system control is high, which makes the problem more difficult.

To solve this, parameter optimization has two solutions. If the objective functions can be formulated, parameter optimization is often solved by classical function optimization algorithm, or intelligent optimization algorithm when the solution space is huge. Otherwise, we can only simulate the system or related processes, and take the output as the target value. The parameters can be solved by simple feedback when the solution space is small, or by intelligent optimization algorithm when the solution space is huge or highly dynamic. Most of the recent studies of parameter optimization are developed in the two aspects above, and the main research and development solution is dividing and reducing the feasible empirical region of parameters, and then taking the method of integrating the simulation and tools to find suitable values of them. As a new and convenient decision-maker, the intelligent optimization algorithm is widely used in studies. For recent studies, refer to [10–19].

2.2.3 Detection and Classification

Detection generally refers to determine whether the condition and variation of an entity or event are beyond normal by features. Classification refers to define the category of the entity or event by features as well. Both of them contain the process of feature-extracting, pre-training and state-judgment. Therefore, they can be classified into one group and all follow the training process according to quantities of samples.

Detection is usually reflected in the fault diagnosis of manufacturing control/simulation system and supervision of manufacturing process and so on. Classification is often embodied in signal analysis of electrical system and model state of machining part, etc. Those problems have the characteristics of scattered samples distribution and uncertain features influence. They are widely exist in manufacturing process, system and structure optimization, but have less research than parameter optimization in manufacturing field. Take the fault diagnosis of machining process as an example, optimization variables are generally the influence weight of several relative features, and the aim is to identify whether a fault exist in the specific case and which kind of fault it is as accurately as possible. Specifically, it is a determining process in which the influence weights of several relative features are trained with samples, and the status of process or objects are detected according to these weights. In a similar way, the classifications are trying to identify the states of objects with feature weights trained from large-scale samples. Problems in this category are slightly similar to parameter optimization mentioned above. In many cases the target function can not be obtained, and we can only make decision according to the output of system or process simulation.

No matter for training or recognition, detection and classification problems are generally solved by some approximate iterative algorithms when the target function can be obtained, or by intelligent optimization algorithm when the solution space is huge, which is quite similar to the parameter optimization as well. If the target function is difficult to obtain, we may simulate the system or related processes in iterations, and take the output as the evaluation criteria. In recent years, most of the studies in detection and classification focus on solving the problems by support vector machine, decision tree and neural network and so on, among which neural network is the most typical one. While there is increasing number of studies in neural network, the application of them in manufacturing field is quite limited. Currently, because of the empirical limitations and complexity of classifier such as neural network, the research is developing mainly in the integration of classifier and other optimization algorithms and their collaborative application in detection and classification. For related studies, refer to [20–30].

2.2.4 Combinatorial Scheduling

Combinatorial scheduling is the most typical combinatorial optimization problem in manufacturing system. It is a reasonable distribution and management of missions, resources and processes. Combinatorial scheduling here includes process planning, job shop scheduling, task scheduling and resource allocation, which schedules the manufacturing process, assembly line, manufacturing services and machines respectively. It is a kind of discrete management and optimization in manufacturing.

Therefore, the variables of combinatorial scheduling are generally integer. The targets are minimizing the task execution time and energy consumption in global or local workflow, and maximizing the quality (such as maintenance and reliability) and the efficiency of production or calculation. The constraints usually are the limits of resource capability, task size and other QoS indexes. The model is simple, while the solution space is typically huge. In addition, the variances and restrictions of variables are complex, which make the feasible solution space more narrow, so as to make the optimization harder. For example, in job shop scheduling problem, most of the studies aim at shortest completion time of part machining. The steps of machining, the number of machine tools in each step, and their machining capability are known, and the distribution strategy of each machine tool in each step of machining is to be solved. The process is complicated, but the target function is simple. When comprehensively considering multi-QoS and multi-objectives, optimal solutions are always hard to get.

For such problems, when the solution space is not huge, integer programming and dynamic programming are often used in solving. When the solution space gets bigger, the use of deterministic algorithms will always lead to the combinatorial explosion. In most cases, sub-optimal solutions are acceptable in combinatorial scheduling, and short decision time is required. Hence, most researchers pay more efforts on the application of intelligent optimization algorithm in combinatorial scheduling problems. And in manufacturing field, intelligent optimization algorithm has become the most applied method in combinatorial scheduling. Due to its typicality, some of the combinatorial scheduling problems such as job shop scheduling and task scheduling have been used as benchmarks of combinatorial optimization for different researchers to test and analyze the optimization algorithms they designed. The improvements and application of intelligent optimization algorithm in combinatorial scheduling by existing researchers are mostly concentrating on two kinds: algorithm hybridation and encoding scheme design. For related studies, refer to [31–40].

2.2.5 Multi-disciplinary Optimization

Multi-disciplinary optimization refers to the combining modeling and analysis of problems with multi-objectives and constraints in different disciplines such as control/mechanical collaborative design, and realizing multi-disciplinary collaborative decision making. At present, networked and collaborative manufacturing system has being greatly developed. Therefore, the whole life cycle of manufacturing can be connected in network, so as to realize control and mechanical collaborative design, machine and monitoring synchronize execution. Multi-disciplinary optimization then becomes more and more important for collaborative work. With the widely research in integrated manufacturing and service-oriented manufacturing, it gradually develop into one of the typical types of problems in industry.

The variables of multi-disciplinary optimization problems mostly include both discrete and continuous ones. Currently the studies related to this kind of problems are very few, and most of them are based on multi-disciplinary collaborative simulation and solved by multi-step decisions. With more and more complex manufacturing system, it can be embodied in all aspects of process, system and structure optimizations. Because many constraints and objectives come from different fields and the relationships among these factors are complex, transformation and simplification are indispensable. On the other hand, simplification is obtained based on the loss of modeling accuracy. The complexity of such problems is obviously huge. For example, in the control/mechanical collaborative optimization, not only the stability and the efficiency of the control system need to be guaranteed, but also the applicability and the portability of the mechanical model need to be improved. Therefore, the variables to be solved usually include the parameters of control system and the critical sizes of part mechanical models. The objective functions are multiple efficiency indicators of the collaborative works, such as material cost, energy consumption and control efficiency, etc. It can be seen that the multi-disciplinary optimization problems are the combination of the above four kinds.

Multi-disciplinary optimization is the least studied one in the problems above. Solution methods for it are mostly based on empirical adjustment and experimental simulation. Although most of the existing studies focus on the objects in manufacturing process, there are also some multi-disciplinary problems in the design and simulation of product structure, and system management and control. Now the methods of onside decision or multi-step optimization in collaborative manufacturing are inevitably not thorough enough. Therefore multi-disciplinary optimization becomes a big challenge and developing trend with the development of advanced manufacturing system. For more information, refer to [41–44] for references in recent years.

2.2.6 Summary of the Five Types of Optimization Problems in Manufacturing

With large literature review, the above five types of optimization problems can be mapped into the three typical objects in manufacturing as shown in Fig. 2.3. Most common scenarios are contained in the classifications. Among them, manufacturing process optimization covers all five kinds of problems, manufacturing system optimization includes three kinds of problems (parameter optimization, detection and classification, combinatorial scheduling), and product structure optimization contains three kinds, numerical function optimization, detection and classification and multi-disciplinary optimization, as well.

According to random selection of the most related 100 literatures in the last 3 years, it can be found out that combinatorial scheduling is studied the most in manufacturing optimization. It accounts for nearly half of optimization research. There are designs and applications for various kinds of certain and uncertain algorithms targeted to combinatorial scheduling, which covers every steps of manufacturing process and the management of manufacturing system. Then, in the next place, the numerical function optimization and the parameter optimization have a close number of studies. In numerical function optimization, the finite element analysis and structure optimization is the majority, and mainly target to the analysis of various kinds of product designs. In parameter optimization,

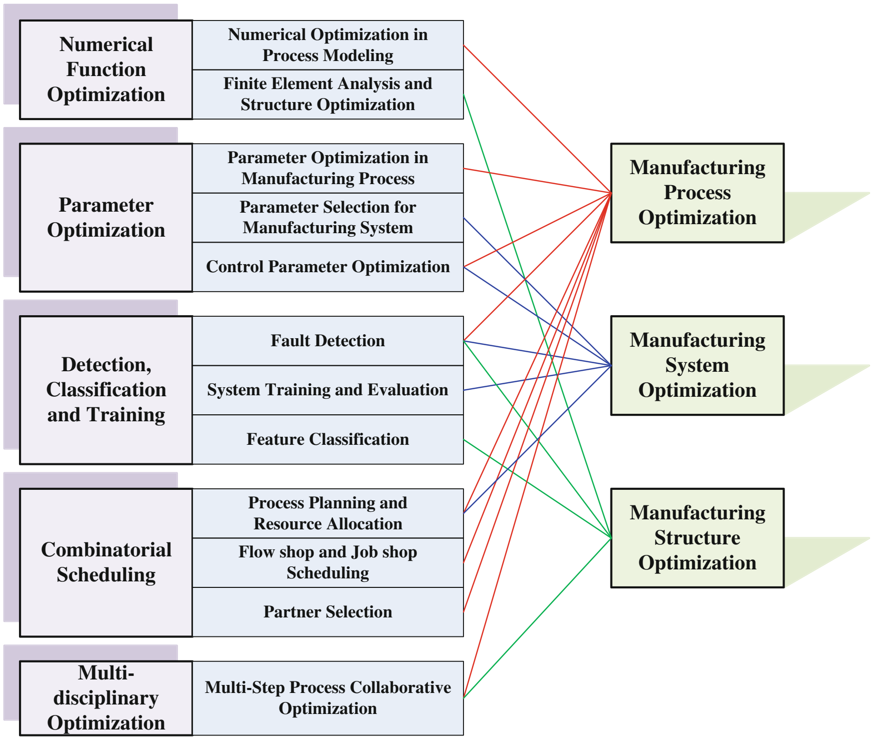

Fig. 2.3 The mapping relationship between typical problems and manufacturing objects

because of the similar adjustment schemes, the studies on parameter tuning in process and parameter alternation in system control are evenly distributed. The three kinds of problems account for nearly 90 % of the studies of manufacturing optimization. On the other hand, there are not many studies on detection and classification in manufacturing field. In this kind of problems, the studies are more on diagnosis and detection, and less on system training and evaluation. The application of feature-based classification in structural optimization is lesser. It is clear that the classification is not highly concerned in manufacturing field yet. Finally, although multi-disciplinary optimization appears more and more in the design of manufacturing process and structure, its complexity makes it the least studies in all these kinds of optimization problems. It is not highly concerned, and the most of the existing studies separate these problems in to several steps and optimize them individually. However, in the above five kinds of optimization problems, the multi-disciplinary optimization problem is one of the most urgent problems. Because of the one-side independent decisions with multi-steps, the decisions are usually inaccurate and the efficiency is not high. A lot of research is required in the modeling of associated features in different disciplines and comprehensive optimization among several disciplines in manufacturing process,

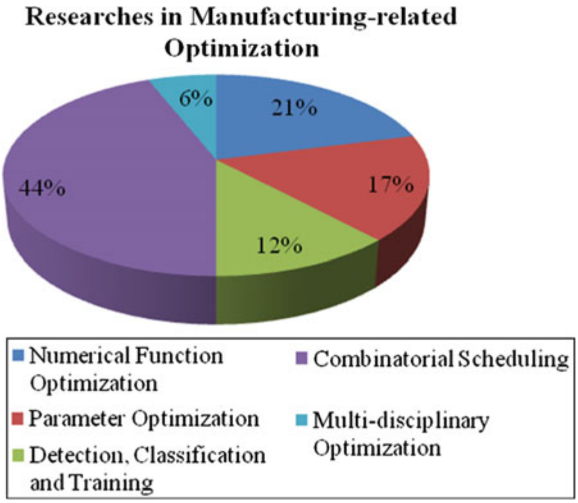

Fig. 2.4 The sample research results on the five types of problems in manufacturing

structure and product design. The sample research results on the problems above are shown in Fig. 2.4.

2.3 Challenges for Addressing Optimization Problems in Manufacturing

After the above analysis, we can see that optimization problems in manufacturing are countless and various. Researchers conducted plenty of works on solving them with different point of view. Nevertheless, those problems with the characteristics of large-scale, multimodal functions and NP-hard are still hard to solve. With widely applied nondeterministic methods, researchers and engineers are taking many attempts to improve the solution quality and reduce optimization time. In this section, we mainly classify the challenges faced in the problems into different categories and analyze them with some existing solution schemes.

The challenges of optimization problems in manufacturing can be divided into seven kinds: Balance of multi-objectives, Handling of multi-constraints, Extraction of priori knowledge, Modeling of uncertainty and dynamics, Transformation of qualitative and quantitative features, Simplification of large-scale solution space, Jumping out of local convergence.

2.3.1 Balance of Multi-objectives

We care the objectives the most in optimization, including the main objectives and the secondary objectives. Most of the problems have more than one objective. It is unavoidable that several objectives are conflicted with each other. In the case that all objectives are unreachable at the same time, balance of multi-objectives generally refers to the average consideration of multi-objectives according to their weights during optimization process.

To normal multi-objective problems, some typical solutions are listed as follows:

(1) Transformation to single-objective problem: It means to combine the objectives into a single function according to their weights, and solve the problem with only one target function. The weights of different objectives in the unique function are set in experience with specific environment. It is a traditional way with low efficiency.
(2) Transformation to constraints: This method considers the main objectives only and tries to transform the secondary objectives to constraints. It mainly takes some main objectives as optimization goals, and takes the minimum requirements of the other objectives as constraints to solve the problems.
(3) Pareto optimization: It considers all objectives at the same time by Pareto non-dominate sorting scheme. The solution is non-dominate only if all of its objective values are better than others. With a Pareto convex set used to collect those non-dominate solutions, both the main and the secondary objectives are evenly considered. Though the optimization of the main objectives is often restricted by the secondary ones, it is the mostly applied method in recent years.

Currently, engineers mostly use the first two methods according to the actual situation, and the researchers are mostly exploring and studying the third method. Among those methods, the first one is the simplest, and it is the earliest method for multi-objective problems. Because the weight of each method is decided by actual condition, the model is usually empirical and has a narrow application scope. It ignores many optimal solutions after weighting. The second method is more flexible than the first one, but it also requires the experience and environmental factors to decide the strength of constraints, which is the weight of secondary objectives, either. It is more adaptive to general decisions with different algorithms, but the transformation of the objectives to constraints brings us a multi-constraint problem, which is more complex. The third method uses the concept of equilibrium in the game theory, improves the other two methods by avoiding the influence of experience. It can give a series of equilibrium optimal solutions taking advantages of non-dominate sorting and the Pareto convex set. However, because of the complexity of its algorithm design, and the uncertainties brought by selecting the solutions in the Pareto convex set according to actual environment, it is rarely applied in engineering, and the studies and comparisons of the algorithms based on this method are not very clear.

It is thus clear that balance of multi-objectives is one of the big challenges in optimization. Now more and more engineers are trying to apply the Pareto thought to practice. How to implement low complexity determination of Pareto optimal solutions and how to select better solutions in the Pareto convex set are both key bottlenecks in manufacturing. For balance of multi-objectives in manufacturing problems, refer to references [45–56].

2.3.2 Handling of Multi-constraints

Besides the variables and objectives in decision-making, the constraints produced by the relationships between variables and parameters are one of the direct reasons to complicate problem. In problem modeling, most of the constraints exist in practice are abandoned for simplification. When applying to real production system or process, that may cause low accuracy or even mistakes in decision making. Thus how to suitably handle multi-constraints in accordance with system environment is one of the most important problems in manufacturing optimization.

Following the previous section, the objectives in the problem can be transformed to constraints. When the number of constraints increases, the constraints can be transformed into the objective functions as well. For handling of multi-constraints, there are several specific ways:

(1) Constraints as penalty function: It means to transform the constraints as a penalty function and multiply or add it with the objective functions. If the constraints are not exceeded, the value of the penalty function is 1 or 0, or it becomes a huge value to make the value of objective functions unacceptable, so as to make the solution abandoned.
(2) Bounds checking: In this method, constraints are independently stored as searching rules. Each solution generated in searching process is checked whether there are out of the restrictions or not, if yes, it will be discarded and replaced by a new one.
(3) Branch-and-bound: It is a classical method which narrows the domains of variables, and divides the solution space into several branches and then reduces the searching range. It is also a preprocessing procedure for searching. The only drawback of it is its high complexity and high dependence in the preprocessing for specific problem.

In addition, there are many other strategies like transforming the constraints into heuristic information or objective functions. Traditional engineering mostly uses the first two methods listed above to deal with the problems. After extensive development of algorithmic search, branch and bound, at present, is gradually used in engineering, which brought many benefits. In general, the first method is simple in design, and has got strong versatility. It can quickly filter out the solutions which do not meet the conditions in the space. But it could easily lead to a loss of feasible solutions and inevitable useless search in large solution space. The second method avoids the influence of deciding by experience, but item-by-item checking in optimization process will lead to greater algorithm time complexity. The third one, which narrows the solution space by branch and bound, is one of the mostly popular methods at present. It can greatly decrease the searching complexity, but brings preprocessing consumption on the other hand. It needs in-depth analysis of actual problems, and with complex problems, it is hard to define the best bounds of solutions with constraints, which leads to a complex design process and low versatility.

Now with the development of intelligent manufacturing system, the handling of multi-constraints in manufacturing problems tends chiefly to design versatile and automatic processing scheme and simplification way for multi-constraints to minimize the searching complexity. For the handling of multi-constraints in manufacturing problems, refer to recent literature [10, 34, 57, 58, 59, 60, 61, 62, 63, 64].

2.3.3 Extraction of Priori Knowledge

For solving a variety of complex problems in manufacturing system and process, research and application also tend to extract the priori knowledge of problem aims at instruct the algorithm to faster searching. Typical examples are the use of prioritization according to priori status of tasks which enables the algorithm find suitable solutions faster, and the selection of nearest neighbor according to priori information in path optimization. It can be down as a kind of greedy strategy. The extraction of priori knowledge has become an important way of solving the problems. When facing various changing problems, the extraction of priori knowledge needs to be conducted in line with the actual environment and features of problems. The versatility of extraction method is low, and improperly designed method will directly cause wrong search direction and then get poor or even wrong solutions.

Currently, on the one hand, the extraction of priori knowledge usually applied in artificial immune systems, artificial neural network systems and intelligent systems based on Agent. By designing the priori knowledge of specific problems, it may perform the rule-based reasoning and prediction to achieve a fast or efficient optimization. On the other hand, it may coordinate with approximation algorithms or intelligent optimization algorithms for complex problems solving, which enhances the searching direction of the algorithms. The extraction of priori knowledge is usually achieved by obtaining the local interactions between variables and objectives. The common factors considered for the extraction can be classified as follows:

(1) Influence of single variable to single objective: Considering only one variable with one critical objective, the interaction between them is calculated as priori knowledge for searching.
(2) Influence of single variable to multi objectives: Considering one key variable with part of objectives, the relationship between the variable and multi objectives are weighed and connect together as priori knowledge.

(3) Influence of multi variables to single objective: Considering multi variable with one key objective, the correlation between multi variables and the objective are weighed and merge together as priori knowledge.
(4) Influence of multi variables to multi objectives: Comprehensively considering part or all of the variables and objectives simultaneously, the priori knowledge is the relationships of the variables and objectives or a reasoning rules for them.

For the calculating of different kinds of influence relations and change of status, we may design an evaluation function as the measurement of priori knowledge, or blur the relations and status and design the mapping between fuzzy priori knowledge and variables. In addition, we can predict the priori knowledge by intelligent training and reasoning according to the existing features and data of simulation systems or models. Now because the lack of research and theoretical analysis in the extraction of priori knowledge especially in nondeterministic optimization, the applications of priori knowledge in manufacturing engineering are much less. An important and difficult point is the way of simplifying and universalizing the priori knowledge extraction and applying them in the widely in actual systems. For the optimization based on priori knowledge, refer to recent literature [65–75].

2.3.4 Modeling of Uncertainty and Dynamics

Problems in manufacturing are all highly uncertain and dynamic. The uncertainty mainly refers to the randomness of characteristics and constraints in the problems, which means that only the range of them can be determined as most time, but the specific values in a period can't be determined. The dynamics refers to the property that the characteristics and constraints of problems are changing with time. The values can be determined only in a period, but they will change gradually. In most manufacturing systems, researchers and engineers always simplify the uncertainties and dynamics of problem to certain values, which will make the design and application of algorithms more convenient. But the simplification will bring inaccuracy and instability. To improve the stability and solving efficiency, the uncertainties and dynamics are accepted as key considerations.

In general, there are several methods to deal with the uncertainty and dynamic nature of the problems in manufacturing:

(1) Replicated simulation: This method is mainly for the modeling of uncertainty. It takes repeated measurements to obtain the mean value and variance of uncertain parameters. Then conduct a number of decisions in a small range around the value to get a set of good solutions. It can be applied in all algorithms but is quite time consumption. Because few tests can not cover all situations, solutions obtained are often inaccurate.

(2) Description with fitting function: This method can be either for the solving of uncertainty or dynamics. From mathematical point of view, it obtains the fitting functions of uncertainty or dynamic by capturing the relation between the actual environment and the variation rules of uncertain or dynamic parameters.
(3) Cyclical forecasting: It is primarily for the modeling of dynamics. It refers to predicting the variation characteristics of the problems at regular intervals. Predicting rules is also conducted according to some test or fuzzy relation among problem features and the environment.
(4) Feedback control: This method can be applied to deal with both uncertainty and dynamics. It does not need to analyze the characteristics of problem and its environment in advance. It refers to design an adaptive feedback control strategy in optimization algorithm to automatically adjust the decision making parameters with variant problem characteristics during the optimization process. It can be seen that this scheme is generally carried out with multi-period problem simulation.

Engineers commonly use the first method according to actual situation, while researchers mostly focus on the design and application with the last three methods, in which the second method and the fourth one are the most typical. In the four methods, the first one is a kind of brute-force methods. It is the earliest processing method of uncertainties and dynamics without mathematical analysis. The second one requires a theoretical basis and practical understanding of the actual problem, and it is more flexible than the first method. However, the design of the second method is harder. The third method conducts regular testing and estimation to the problem by typical predictor and corrector, which solve the design difficulties of the second method. But the prediction time during optimization directly increase the time complexity of algorithm in most cases. The fourth method borrows the idea in control theory and uses the problem states supervised in each period as feedback to design an adaptive strategy which can control the algorithm parameters so as to adapt different situation and obtain good solutions. The design of feedback regulation is simpler than the fitting function, and it is more versatile, but it has the same problems in the design of adaptive control rules as the second method.

Now the accurate modeling of uncertainty and dynamics is still a difficult problem, which is a direct reason of the inefficiency in the decision-making of manufacturing system engineering. Among those methods, cyclical forecasting is quite appropriate for the modeling of dynamics according to the change of time, and has great potential. Feedback control is more fit to the modeling of uncertainty. On the whole, the development of algorithm with the consideration of uncertainty and dynamics which can adaptively adjust the variances in problem is a major trend. For the modeling of uncertainty and dynamics, refer to the references [76–87].

2.3.5 Transformation of Qualitative and Quantitative Features

No matter in manufacturing system simulation or process modeling, qualitative analysis needs to be done at the beginning. Then how to transform the qualitative parameters and variables to quantitative values for decision is also a big challenge. The accuracy and reliability are the main targets in the transformation. Therefore we define the conversion between the qualitative and quantitative characteristics as a quantitative description process for the complex properties and characteristics in problems. Only when the quantitative description possesses certain precision and credibility, the solving will be meaningful. Similar to the simplification of uncertainty and dynamics in manufacturing system, in this aspect we measure the problem attributes mathematically. To improve the exactness of problem modeling and solving efficiency, the transformation of qualitative and quantitative features is an important issue to be considered in manufacturing optimization.

In general, there are several ways to deal with the transformation. A few typical ones are introduced as follows.

(1) Fuzzy quantification: It means to represent different problem qualitative attributes as fuzzy value according to their levels and intensions.
(2) Functional quantification: This method defines a fitting function in a certain range to describe the attribute variations with time or environment.
(3) Discrete quantification: It refers to describe qualitative features with a set of discrete values in a certain domain. It is not only for discrete attributes, but also for continuous ones as a compromise between fuzzy quantification and functional quantification.
(4) Stochastic quantification: It is especially for uncertain features in problem. The quantitative values can usually obtained by a series of Monte-Carlo or other stochastic tests. It is inaccuracy but can better describe the uncertainty of qualitative features.

The above four methods are provided for different kinds of problems as methods of transformation between qualitative and quantitative attributes. The first three methods can be well applied in engineering, while the fourth one is less applied because its accuracy and reliability are hard to verify. The fourth method is only suitable for a few problems which have extremely uncertain attributes.

In existing studies, the studies targeted on the transformation of qualitative and quantitative attributes are quite few in manufacturing. Most of them do quantitative conversion and problem modeling based on the above methods without consideration of the accuracy and reliability verification of the model. Nevertheless, the accuracy and reliability are usually the deciding thresholds of the quantification. If transformation method is not verified, the model will bring unconvincing decision in engineering applications, or even lead to large deviation to the solutions and cause big loss. Therefore, the verification step in this issue is a more important factor in problem modeling and it is more challenging. In the studies of optimal decisions related to manufacturing, the transformation of qualitative and quantitative attributes exists widely, such as [88–95].

2.3.6 Simplification of Large-Scale Solution Space

With the complication of manufacturing system and the whole life cycle of production, manufacturing resources and processes are getting to be abundant, and the solution space of the optimization problems is getting bigger. With the increasing of the solution space, the accuracy and the time efficiency of existing algorithms are decreased a lot. Hence, simplification of large-scale solution space is also one of a big challenge to better adapt the optimal searching. Specifically, the simplification of large-scale solution space is a process to divide or simplify the problem and solve it in multi-steps with lower complexity.

Facing with large-scale complex problems, the common methods for the simplification of large-scale solution space are:

(1) Divide and conquer: It means to separate a problem into several sub-problems and narrow the size of sub solution space in each optimization step.
(2) Decrease and conquer: This method tries to find a mapping relation between the original problem and another problem with small solution space. Getting rid of unfeasible solution regions according to the constraints can be also fully applied in this method.
(3) Transform and conquer: By instance simplification, representation change and problem reduction, this method aims to transform the original problem to another representation and reduce the solution space during the process.

The three kinds of methods are originally used as deterministic algorithms for different sorts of optimization. They are also effective in dealing with large-scale solution space. The simplification is generally done by conducting a mapping scheme between the solution spaces of the original problem and the simplified one. In intelligent optimization algorithm, the way to simplify the large-scale solution space is usually the encoding scheme. Now many studies on the simplification of large-scale solution space have shown up. The most prominent and effective studies are divide and conquer and its improvements.

However, from the perspective of engineering solving, the complexity of existing problems is gradually increasing, and there are endless kinds of problems. The simplification analysis of problem solution space requires a lot of time, and the exponential exploration in deterministic optimization is still not well solved. In the solving process of large-scale complex problem, finding a general method to simplify large-scale solution space for various kinds of special complex problems is still a big challenge. For the existing studies in simplification of large-scale solution space, refer to references [96–102].

2.3.7 Jumping Out of Local Convergence

According to the above discussion, many deterministic algorithms can not find optimal solution in polynomial time owing to the growing complexity and scale of problems in complex manufacturing system or process. Therefore, various kinds of nondeterministic algorithms such as intelligent optimization algorithms are presented. These algorithms aim at giving feasible sub-optimal solutions of problem in a short time, and conduct stochastic and heuristic search in the solution space. The core issue in nondeterministic algorithm is how to jump out of local convergence and find better sub-optimal solutions.

Jumping out of local convergence refers to design strategies in algorithm which can promote the stochastic evolutionary process to find better solutions in the situation of local convergence. When an algorithm is trapped into local convergence, it will search repetitively in a small region until terminal conditions are reached. Early convergence will definitely lead to low efficiency and high time consuming in problem-solving. In over 30 years of theoretical study, researchers performed in-depth analysis about the convergence of many iterative-based algorithms. However, only a few are verified theoretically so far. From the perspective of practice, researchers made efforts on the design of algorithm improvements to escape from early convergence in solving different problems, such as the increasing of search step, the eliminating of similar solutions, the importing of chaos and the adaptive parameter tuning. Many of them have been applied in various kinds of engineering problems. They have high reusability and have their own focus in specific problem.

However, the strategies for jumping out of local convergence have not been effectively improved. Due to the huge scale solution space, high stochastics, unsuitable heuristics and so on, it is harder and harder to improve the efficiency of problem-solving. There is no free lunch. Facing with expansive complex problems, handling the balance between exploration and exploitation with iterative-based local optimization are discussed a lot. Jumping out of local convergence is still one of the huge challenges in today algorithm design and problem solving. For more instances, refer to the references [103–110].

2.4 An Overview of Optimization Methods in Manufacturing

Facing so many challenges, researchers and engineers keep looking for high efficient optimization method to solve those complex problems in manufacturing system and process. On the whole, we may divide the optimization methods into six categories according to their design and solving process, i.e. Empirical-based method, Tool-based method, Prediction-based method, Simulation-based method, model-based method and Advanced-computing-technology-based method. All those methods require the support of intelligent optimization algorithm in solving most problems. Therefore, we briefly describe the six kinds of methods in manufacturing and then show the key elements in design with typical examples.

2.4.1 Empirical-Based Method

Empirical-based method generally refers to the optimization according to the reasoning and analysis based on experience information in problem modeling. It is mainly applied in the situation that some properties of the problems, such as the variable domain and range, can not be defined, or problems with stochastic and large solution scale that can not be traversed. Typical instances are in the process control of complex system, and the parameter selection in product design and so forth. Some classical schemes applied in empirical-based methods are as follows.

(1) Empirical local search: It is defined as empirical selecting and narrowing the domain of variables to be solved in the problems according to environmental information, and searching locally in a small solution region. This process is actually an empirical selection of searching domain.
(2) Empirical stochastic search: By dividing the solution-space, it tries to set the search probability and success rate of different solution area according to the experience and information, and obtains the feasible sub-solutions by random searching. This process is an empirical selection of search probability.

These methods mostly use empirical environment data to define the properties of problem and divide and check the searching area so as to simplify the optimization process. There are many studies on the modeling of empirical data or features in complex problems. In empirical-based methods, the key point is that the verification and selection of reliable empirical data and priori knowledge. It is mainly used to deal with the challenges like the modeling of uncertainty and dynamics, and the simplification of large-scale solution space for different manufacturing environments. In the problems which require empirical information, accuracy requirements in optimization are usually low, while the requirements of feasibility and efficiency are high. Therefore, the empirical-based solving is mostly indispensably combined within intelligent optimization algorithms and other uncertainty algorithms like approximation algorithms to design. It is largely problem independent with manual regulation. Besides the cases given in the literature in last section, refer to references [111–117] for the studies in empirical-based solving.

2.4.2 Prediction-Based Method

Prediction-based method generally refers to the optimization in which some problem or algorithm attributes are trained and predicted based on environmental information during the process. The solving process is guided by the changing predicted parameters. It is mainly applied in the situations that the accuracy requirement of decision is high, but the problems are real-time, dynamic and uncertain and the dynamic and uncertain parameters can be modeled with time-stepping iteration. These problems usually appear in dynamic time-based scheduling, and real-time control for manufacturing process and so forth. Thus in manufacturing, prediction-based strategies are used in problems like parameter optimization, combinatorial scheduling, and detection and classification the most. The most applied prediction schemes are listed as follows:

(1) Prediction with fitting function: Based on the change rules of the prior tested data and attributes along with the time and environment advance, this strategy tends to carry out prediction with the fitting function of these rules. In each step of prediction, the problem attributes in the next period are calculated according to the fitting function.
(2) Fuzzy prediction: In this strategy, the attribute values are generally divided into several levels, and the prior tested data are clustered with fuzzy processing. After that, the mapping relation between attribute levels and the changing environment needs to be conducted to guide the prediction.
(3) Prediction with classification: Combining with classification algorithms, this strategy establishes a training model according to the problem priori dataset. On the basis of the training model, the key states of problem in the next period can be predicted for next step decision.

In addition, there are many prediction strategies applied for dynamic optimization. Most of them firstly model and analyze the mapping relation between problem features and environmental dynamics according to priori data, then guide the optimization by predicted model to improve the solving accuracy. They can be widely applied in dealing with challenges like the modeling of uncertainty and dynamics, transformation of qualitative and quantitative features, balance of multi-objectives and handling of multi-constraints. In this method, the key point is the accuracy of the predicted model. Due to the prediction training or mapping and the real-time optimization can be performed in parallel, the requirement on time consumption of prediction are lower. With the increase of the manufacturing and system complexity, the prediction-based method gradually requires the short-time and dynamic invoking of intelligent optimization algorithm to better serve different situation with iterative searching. For the prediction-based method in manufacturing field, refer to references [118–125].

2.4.3 Simulation-Based Method

Simulation-based method generally defines as the optimization which obtains problem states by real-time simulation, and solves the problem by state monitoring with feedback compensation. This method is mainly applied in the situation that the problems are dynamic and have the feature of strong real-time, or the problem attributes and environmental parameters can not be modeled exactly. Furthermore, when the predicted data is hard to obtain or not able to obtain as the time is insufficient, this method is more useful and accurate than the prediction-based method. But sometimes the simulation is hard to implement and its application scope is narrow. It is a mainly applied optimization method in manufacturing for complex product modeling and designing. Thus it often appears in finite element analysis and multi-disciplinary optimization. Here are several simulation-based methods:

(1) Real-time monitoring: It is conducted based on several simulation tests. The outputs of real-time monitor are directly applied as the input of optimization. Then dynamic optimization in real-time can be established.
(2) Multi-run simulation: In this method, the problem parameters are obtained by the record of multi-run simulation. The optimization can be carried out based on either the average value or one stochastic value of the output.
(3) Feedback simulation: It refers to simulate the problem with input and output monitoring, regulate the output status according to the real-time system monitoring, and guide the optimization by both the regulating rules and the simulation output.

In manufacturing systems, the first and the second methods are the most commonly used ones. The methods above all take multiple sets of input and output in the simulation as reference variables or objective values. The key point is not the accuracy of simulation, but the dynamic processing ability of algorithm based on simulation. This method can be widely applied in various manufacturing problems with high real-time and dynamics. However, due to the difficulties in simulation construction, research in simulation-based method in recent years is relatively less. It is more used in the modeling of real-time process or product design and can be combined with intelligent optimization algorithms to construct an adaptive evolution and dynamic feedback to solve complex problems. For simulation-based method, refer to references [126–132].

2.4.4 Model-Based Method

The model-based method is the simplest and most commonly used method, which solve the problem according to the mathematical description of its variables, objectives and constructions. Although most of the application and research in manufacturing simplify the problems with quantitative mathematical description, owing to the simplification of some key dynamic factors, the decision result of this method is somewhat inaccurate. It is mainly applied in the situation that the properties and features of problems are clear without uncertainty and dynamics, such as in balance of multi-objectives and handling of multi-constraints. The key point of this method is the accuracy and reliability of the mathematical description of the problem. Due to its universal application, we will not repeat it here.

2.4.5 Tool-Based Method

The tool-based method mainly refers to the optimization that dynamically extracts problem features and solves the problem with the assistance or guidance of system or process management tools. In a broad sense, the tool-based method can be defined as a kind of simulation-based method. But narrowly speaking, the tool-based method focuses on the use of assistant tools to obtain features and its guidance on optimization in practical, while the simulation-based method focuses on the modeling of problem features in virtual simulation. Like simulation-based method, the tool-based method is mainly applied in the situation that the problem is dynamic and uncertain with time and its properties are hard to extract quantitatively. Compared to the building process of simulation-based system, because of the tool functions, the tool-based extraction of assistant features with monitoring has a lower difficulty to achieve. Its application scope is also wider. It is mainly applied in manufacturing problems such as finite element analysis, parameter optimization and multi-disciplinary optimization.

To the extraction of problem attributes and environmental characteristics, there are various kinds of existing assistant tools. There are also many studies focus on the algorithm design based on the tool assistance, and trying to integrate those tools to improve the reliability and efficiency of algorithms applied in specific environment. The key point is the collaboration process of inputs and outputs of the tools and the designed algorithms, which is similar to the methods mentioned above. For the tool-based method, refer to references [8, 81, 133, 134, 135, 136].

2.4.6 Advanced-Computing-Technology-Based Method

Advanced-computing-technology-based method refers to conduct optimization by ways of parallel, distributed multi-steps and collaborative computing with the help of existing advanced computing resources. This method is different from previous design of solving methods. It is combined with ideas of problem and algorithm partitioning, designed to use existing advanced technology to allocate sub-problems or sub-optimization tasks in distributed resources and solve them in parallel. This method is mainly applied in large-scale problems and multi-disciplinary optimization. It has the characteristics of high scalability and efficiency. Because of the high requirements of professional technology, the solving process of specific problems based on this method is quite complex in design, and hard to realize. From the perspective of algorithm design based on existing advanced computing technology, there are several typical ways:

(1) Design based on distributed computing: Perform multi-step decision-making by dividing the problem into several steps and design or call different resource service for the module design of each step.
(2) Design based on parallel computing: Divide the problem into several parallel execution modules that the data or control dependencies among them are the fewest. Then divide the modules into different resources and perform parallel computing.
(3) Design on collaborative computing: Divide the problem into several execution modules which have the hybrid relations of series or parallel. Then solve these modules by different resources and algorithms.

The basic ideas of these solving methods based on advanced computing technology are all dividing the problems and using distributed resources efficiently. In this method, because of the dependencies among the modules in the divided problems or algorithms, we have to consider the time-consuming of task communication. Thus the dividing scheme should be customized according to the environment and complexity of the problem. No matter which technology is used to divide the problems and package the modules, the key is always the partitioning and allocation of the whole optimization. In recently, the problem scales in existing manufacturing systems and processes are increasing gradually, and these problems involve several kinds of multi-disciplinary optimization. Therefore, the studies based on advanced computing technology gained prominence gradually. The solving efficiency brought by those methods can't be ignored. The collaboration and hybridation of various kinds of algorithms for solving different subproblems make the analysis and decision of complex problems easier and the application of distributed computing resources make the whole optimization more efficient. But facing the bottlenecks of the communication costs and the implementation of related technologies, general module partition schemes with less communication becomes another challenge. For advanced-computing-technology-based method, refer to references [137–144].

2.4.7 Summary of Studies on Solving Methods

We found that the algorithms designed for complex manufacturing optimizations are hybrid with different kinds of strategies. Therefore there are no clear boundaries among these methods of algorithm design, like the empirical-based method uses prediction at the same time, and the simulation-based method uses assistant tools in many situations, etc.

From the perspective of the application of different methods, it can be seen that the model-based method is mostly used. No matter in dynamic optimization, black-box-based optimization or monitor-based optimization, objectives functions (or evaluation functions), the abstract variables and parameters and their relations are presented as mathematical models. In actual projects, the empirical-based method is right after the model-based method in the number of application times, and it mostly get the assistance of simulation and tools to optimize. In recent years, there are more studies on the prediction-based method and advanced-computing-technology-based method because the prediction of major cases will make the optimization process more intelligent, and the advanced computing technology can always take the advantages of distributed resources to make the solving faster. In addition, because of the complexity and particularity of manufacturing process, there are few studies on simulation-based method and tool-based method appeared in recent years. On the whole, after sampling of recent literatures, we roughly get the mapping of solving methods and the optimization problems, as shown in Fig. 2.5. **The numerical orders we get from the research literatures on different methods are: model-based method > advanced-computing-technology-based method > prediction-based method > empirical-based method > simulation-based method > tool-based method. And the application numbers of them in actual projects are: empirical-based method > tool-based method > simulation-based method > prediction-based method > advanced-computing-technology-based method**. From the difference between the solving methods used in actual projects and in research, we can find out that we have to consider the integration of tools, simulation, and the existing advanced intelligent technology further to narrow the gap between optimization design in projects and in research.

2.5 Intelligent Optimization Algorithms for Optimization Problems in Manufacturing

In summary, there are varieties of complex problems in manufacturing. Decision and optimization face multiple large challenges simultaneously. To deal with these challenges, a series of solving methods including several problem modeling and algorithm design methods have already been proposed. Throughout the studies on the optimizations in manufacturing field, the improvement and application of intelligent optimization algorithms are one of the major parts. No matter modeling the problems by experience, prediction or tools, or designing algorithms based on simulation, heuristics, or advanced computing technology, intelligent optimization algorithm are widely applied in the optimization process because of its independence, versatility and efficiency. At the same time, with multiple challenges in manufacturing problems, there are a series of studies and improvements carried out

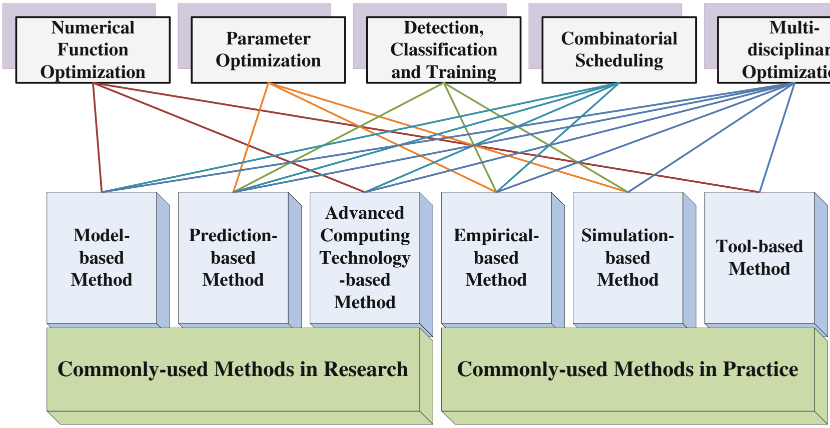

Fig. 2.5 The mapping of the solving methods and the optimization problems in manufacturing

in encoding schemes, operator designs and evolutionary strategies of intelligent optimization algorithms. These achievements are fully reflected in the six sorts of optimization methods

Specifically speaking, in the empirical-based and prediction-based method, intelligent optimization algorithms are required for the selection of empirical or prediction factors and data. In simulation-based and tool-based method, it is needed for adaptive control the modeling and simulation. In advanced-computing-technology-based method, intelligent optimization algorithms are also used to partition the optimization tasks and modules to some extent. On the contrary, all these methods are used and studied as assistant strategies for intelligent optimization algorithms to solve these complex problems with higher efficiency. Facing with the combinatorial explosion generated by complex problems, we can only take advantage of the support of intelligent optimization algorithms with independent iterations and the characteristics of high versatility and scalability to avoid the combinatorial explosion in the problems, and get the satisfying solutions in a short time.

The application review of common intelligent optimization algorithms in manufacturing field can be listed as shown in Table 2.1. It can be clearly seen the research emphasis on different kinds of problems corresponding to different optimization methods in recent years.

Typical cases of each kind of problem in manufacturing field are listed in the table. Each case faces all the seven challenges described before. However, due to the different characteristics of problems, they have different emphasis in dealing with those challenges. In this table, the main challenges to be overcome in these typical cases are shown by check marks. Furthermore, the solving methods mainly taken for different kinds of problems are listed in the next paragraph.

Based on the literature review, it can be found that 60–70 % of the research and applications in solving complex manufacturing problems use different styles of

Table 2.1 Summary of the characteristics, research emphasizes and major methods of complex problems in manufacturing field

Problems	Typical cases in manufacturing	Multi-objectives	Multi-constraints	Priori knowledge	Uncertainty and dynamics	Qualification and quantification	Large-scale solution space	Local convergence	Major methods
Numerical function optimization	Functions in process modeling [2, 9]		✔				✔	✔	Model-based method Advanced-computing-technology-based method Tool-based method
	Finite element analysis [1, 4]		✔				✔	✔	
Parameter optimization	Parameters in control [14, 15]	✔	✔	✔	✔				Empirical-based method Prediction-based method Simulation-based method
	Parameters in machining (Wu et al. [10, 11]	✔	✔	✔					
	Parameters in system [12, 17]	✔	✔	✔					

(continued)

Table 2.1 (continued)

Problems	Typical cases in manufacturing	Multi-objectives	Multi-constraints	Priori knowledge	Uncertainty and dynamics	Qualification and quantification	Large-scale solution space	Local convergence	Major methods
Detection and classification	Detection in machining [28, 29]			✔	✔	✔	✔		Empirical-based method Prediction-based method Simulation-based method
	System training and assessment [23, 24]			✔	✔	✔	✔		
	Classification of model features [21, 27]			✔	✔	✔	✔		
Combinatorial scheduling	Workflow management [31, 35]	✔	✔		✔		✔		Model-based method Prediction-based method Empirical-based method Advanced-computing-technology-based method
	Job shop scheduling [38, 40]	✔	✔		✔		✔		
	Production partner selection [33, 39]	✔	✔		✔		✔		

(continued)

Table 2.1 (continued)

Problems	Typical cases in manufacturing	Multi-objectives	Multi-constraints	Priori knowledge	Uncertainty and dynamics	Qualification and quantification	Large-scale solution space	Local convergence	Major methods
Multi-disciplinary optimization	Collaborative process optimization in manufacturing [41, 43]	✔	✔	✔	✔	✔	✔	✔	Empirical-based method Prediction-based method Simulation-based method Tool-based method Advanced-computing-technology-based method

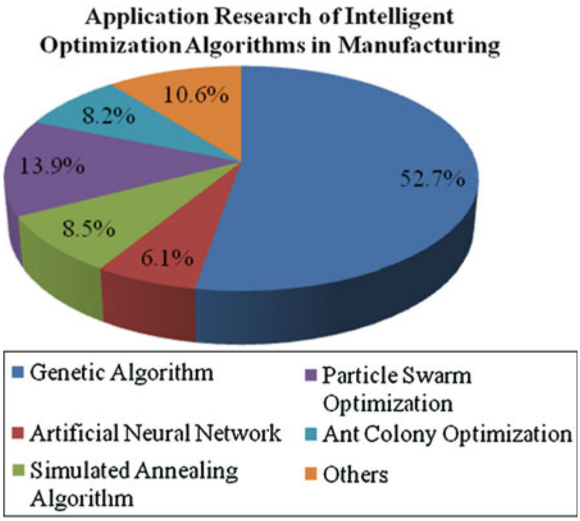

Fig. 2.6 The approximate application distribution of common intelligent algorithms

intelligent optimization algorithms, in which genetic algorithms, artificial neural network algorithms, simulated annealing algorithms, particle swarm algorithms and ant colony algorithms are the most typical and applied. The approximate application distribution of common intelligent optimization algorithms is shown in Fig. 2.6. The most applied one in different kinds of problems is genetic algorithm. Because it is proposed the earliest, and its operators are simple and independent, which means the algorithm is appropriate both for continuous and discrete optimization. However, with the characteristic of premature convergence, different kinds of improvements and combination to the genetic algorithms are designed based on various benchmarks and practical problems. Except genetic algorithm, ant colony algorithm and particle swarm algorithm are applied a lot. The self-learning mechanism of particle swarm algorithm is designed for continuous numerical optimizations. In discrete combinatorial optimization, the updating mechanism of individuals needs to be changed and improved. Most of the improvements transform the original change to the crossover between individual. In this situation, the original particle swarm optimization is transformed as a kind of hybrid genetic particle swarm algorithm to some extent. Overall, particle swarm algorithm is mostly applied in numerical function optimization, parameter optimization and multi-disciplinary optimization. On the contrary, ant colony algorithm is designed for path finding related combinatorial optimization, such as route optimization of robots, task scheduling and so on. In continuous numerical optimization, the searching step size needs to be set beforehand. If the step size is large, the accuracy cannot be guaranteed, if the step size is small, a large-scale pheromone vector is required, which slow down the search. Moreover, ant colony algorithm generally needs the extraction of priori knowledge, which makes the algorithm not very versatile in application. In addition, because typical simulated annealing algorithm and tabu search have strong randomness, and they are carried out with single individual-based iterations, the probability of getting better solutions in a short time is low. Currently, the hybrid of intelligent optimization algorithms and other algorithms is mostly applied in application, which offers great assistance to improve the searching diversity and guidance.

In addition, there are several typical intelligent optimization algorithms applied in the continuous numerical optimization like simulated-annealing algorithm and differential evolution algorithm, and in the discrete combinatorial optimization like memetic algorithm, ant colony optimization and so on. Other algorithms like immune algorithm, DNA computing algorithm, culture algorithm and newly appeared bee colony algorithm are not mature in development and application. Most of the engineers are not familiar with these new algorithms. As a result, few studies are applied on the problems in manufacturing field. It also reflects an important thing in the study of optimization, that the new better research results are not effectively used in actual projects.

2.6 Challenges of Applying Intelligent Optimization Algorithms in Manufacturing

Currently, intelligent optimization algorithm has become an integral expertise in manufacturing system and process optimization. It helped to breakthrough a lot of difficulties in optimization, like the decision of job shop scheduling and finite element analysis, etc. However, the problem will change with the environment. Therefore, in complex manufacturing systems, to improve the efficiency of problem optimization and decision, the design and development of intelligent optimization algorithm becomes a research hotspot. Although thousands kinds of improvement, hybridation of intelligent optimization algorithms have been proposed, their solving effects on different kinds of specific problems are still unknown. In algorithm design process, different challenges still exist in all of the steps, i.e. problem modeling, algorithm selection, encoding scheming and operator designing. According to the characteristics and requirements of different designing parts, we will analyze the main challenges separately.

2.6.1 Problem Modeling

From the perspective of problem modeling, the three basic elements are variables, objectives and constraints. Modeling of the three elements directly influences the quality of decision. Thus it is the basis of the designing of intelligent optimization algorithm.

Firstly, there is one-to-one relationship between the problem variables and the individuals in algorithm. If the variables are continuous, the factors as domain, search step, accuracy requirements should be explicitly given. If the variables are discrete, besides the domain, we have to clarify the direct relationships between variables, which will make the following encoding easier.

Secondly, no matter for single-objective or multi-objective problem, the objective functions as evaluation criteria in the algorithm are the essential foundation of search. For the problems can be mathematically modeled, clear objective functions need to be given. For the problems that objective function can not be given, like parameter optimization in process control and detection, we have to test the solution with simulation or monitoring of actual systems to get the result. Then the outputs of the system are taken as the objective values to evaluate the population in algorithm iteration in real-time. It is important to note that too simplified evaluation functions will result in low accurate optimization, while too complex assessment model will lead to large time loss.

Finally, in dealing with constraints, no matter put them in the objective functions as penalty functions or define them as population check during iteration, they will greatly influence the algorithm optimization. Inappropriate handling of the constraints will easily lead to invalid iteration search in unfeasible solution space, which will seriously reduce the efficiency of algorithm.

It is thus clear that in the establishing process of optimization model, the main difficulties are:

(1) The precisely clarification of the properties of problem variables;
(2) The appropriate construction of objective functions (or assessment methods);
(3) The suitable handling way of constraints.

2.6.2 Algorithm Selection

Based on the establishment of problem model, full research and comparison of hundreds of hybrid and improved intelligent optimization algorithms and selection of suitable one for the specific problem is almost impossible. To most engineers, it is even not easy to choose in a set of basic intelligent optimization algorithms. Because almost all of the existing intelligent optimization algorithms have not been verified theoretically, large experiments are mostly required for compare their efficiency to specific problem. Hence, it is quite hard to figure out which algorithm is suitable and which improvement or hybrid strategies can bring about enhancement for specific problems.

Currently, people usually select the most commonly used algorithm according to different application extents of the algorithms and the recommendations from existing research when facing complex problems in manufacturing field. Most of them select the most classic genetic algorithm, and ignore many new intelligent optimization algorithms. Based on the selected algorithm, according to the former procedure, the algorithm is implemented and improved again, which extends the algorithm design cycle and produce a lot of repetitive work. In addition, if the selected algorithm is inappropriate for the problem, such as using a highly evaluated intelligent optimization algorithm specializing in discrete combinatorial problems to solve specific continuous problem, the algorithm requires to be transformed a lot, and the result of the optimization is still possible to be substandard.

It is thus clear that algorithm selection is the core to decide the optimization efficiency in solving a problem. Currently, there are many integrated libraries which can provide some typical intelligent optimization algorithms. But with different kinds of problems, the merits and demerits of various algorithms can not be compared directly. And also, there are still less study and emphasis on the construction of related algorithm libraries. People are more willing to select mature and convenient method to solve the problem.

Generally speaking, in the process of algorithm selection, the main questions need to be note are as follows:

(1) Less analysis, verification and classification on existing typical intelligent optimization algorithms in solving different problems;
(2) Lacking of unified evaluation methods for various algorithms in solving different problems;
(3) Lacking of a standard integrated algorithm library for algorithm design, comparison and application.

2.6.3 Encoding Scheming

Encoding scheming for problem refers to the process of transforming the key variables into individual genes. It is the band between intelligent optimization algorithm and specific problem combined with fitness function. Population updating in each generation performed by combined operators is also based on encoding scheme.

Normally, binary-coding, real-number-coding and vector-coding are the most commonly used. In some encoding schemes, individuals and variables have one-to-one mapping relationship. However, in most coding form for particular problems, individuals and variables are not one-to-one mapping. When they are having one-to-many relationship, i.e. one individual corresponds to several solutions, the decoding can not be well implemented. When they are having many-to-one relationship, i.e. several individuals correspond to one solution just like the situation of real-number-coding in task scheduling, then invalid searching with repeated iteration and local convergence can easily occur, which is detrimental to the whole evolutionary optimization. Moreover, in combinatorial optimizations like job shop scheduling and traveling salesman problem, the targets are finding the best permutations of variables, which means that the values of different variables can not equal to each other, then the requirement to the coding in such situations is very high. Not only so, operators like crossover in genetic algorithm and self-studying in particle swarm optimization are designed and varied with different coding scheme. Therefore, coding scheming is crucial in the application of intelligent optimization algorithms.

Currently, the main study aspects and difficulties in encoding scheming for specific complex problems are:

(1) Coping with "many-to-one mapping situation" to avoid repetitive searching.
(2) The avoiding of inflexible solutions with encoding scheming.
(3) The adjustment of encoding scheme for specific operators.

2.6.4 Operator Designing

Based on these three steps of design, people have to improve the algorithm after the selection to adjust the problem and achieve a higher efficiency of optimization. But there are too many operators and improvement strategies. Thus users and designers have to perform further research and analysis on the existing operators and improving strategies based on the encoding scheme and improve the algorithm again. Based on the requirements of the problem and the selected algorithm, the design of improvement strategies for the operators can be seen as a matching combination process. Different type of operators and improvement strategies can form several hybrid and improved algorithms after different permutations and combinations.

In the existing research and application, people usually adjust and combine the operators according to the existing experience, or perform single-step fine tuning in operators and try improved strategies one-by-one to specific problem. The interactions among the operators and the balance between exploration and exploitation in iterations are less considered and analyzed, which leads to great limitations in the performance of existing operators and improvement strategies.

Therefore, in the improvement design process of operators, the main difficulties are:

(1) Lack of analysis in features and combination effect of the operators for different problems.
(2) Lack of balance between exploration and exploitation in algorithm design.
(3) Many good improvement strategies have not been well extended and applied to different types of problems.

2.7 Future Approaches for Manufacturing Optimization

Challenges are not only existed in the above mentioned four steps. With the gradual complication of manufacturing process and system and the proposal of advanced manufacturing model such as networked and service-oriented manufacturing, the problems in manufacturing field are more evolving into multi-disciplinary large-scale ones. Thus, single deterministic algorithm or intelligent optimization algorithm is far from sufficient to meet the requirement of solving.

From the algorithm designing perspective, it requires the assistances of more than one of the six kinds of solving methods. For the optimization process, different adaptive, exploitation and exploration strategies are needed in different solving stage. For solving the problem in real-time, we need to handle the dynamics and uncertainties with the adaptation and combination of various operators. In a word, the whole solving process for multi-disciplinary complex problems needs multi-level or multi-stage operators combined with multiple decision methods and technologies.

It can be concluded that for the optimization problems in manufacturing field, the requirements are diverse, and the corresponding solving methods have to meet not only the requirements of the system dynamics and uncertainty, but also the ability of efficient collaborative optimization and management. The trend of development can be briefly summarized into several points:

(1) Hybridation of diversified methods.
(2) Multi-stage processing of uncertainties and dynamics.
(3) Technologies for rapid real-time responding and decision-making.

2.8 Future Requirements and Trends of Intelligent Optimization Algorithm in Manufacturing

With such a general trend, intelligent optimization algorithms need further digging to enhance its efficiency, flexibility and scalability according to the requirements of the three main users in manufacturing, i.e. algorithm beginner, algorithm employer and senior researcher, to adapt practical complex decision in manufacturing engineering. Although the theoretical analysis of operators on the intelligent optimization algorithms is of great importance, yet from the aspect of engineering application, implementations of integrated, configurable, parallel and service-oriented intelligent optimization algorithms are becoming the key development trends in solving complex manufacturing problems.

2.8.1 Integration

During the digital industrial producing process, every step in the whole life cycle of manufacturing contains simulation and tool-aided analysis with varieties of professional software. As for every single optimization module in manufacturing, engineers need to implement and encapsulate different kinds of intelligent optimization algorithms according to I/O interfaces of the module to realize automatic and systematic decision. With the collaboration of diversified tools, integration of intelligent optimization algorithms and other technologies as modules are necessary for implementing high efficient comprehensive decision.

Specifically, on one hand, based on a rich mixture of assistant tools, many environmental parameters and problem attributes can be obtained easier. The support of multiple technologies can be seen as a combination of simulation-based, tool-based and advanced-technology-based methods. Integrating the intelligent optimization algorithms with these assistant tools and technologies can makes them easier to adapt in specific systems and perform better function. On the other hand, integration and encapsulation of multiple intelligent optimization algorithms can make multi-methods' decision possible. For specific problems, engineers can compare different algorithms in practical environment and apply more than one to do optimization in different stages.

On the contrary, if we design and implement the intelligent optimization algorithm in each different environment, the design process of optimization will be more complicated and time consuming. Therefore, in order to achieve simplified and high efficient collaborative optimization, the most convenient way is to integrate diversified intelligent optimization algorithms and multiple technologies together in the form of tools, and provide uniform standard interfaces to connect with different kinds of systems.

In recent years, some research has turned to the integration of basic intelligent algorithms based on the uniform search of population-based iteration. However, most of the existing integrated algorithm platforms or libraries are inapplicable to complex optimization in collaborative manufacturing. They are generally constructed according to traditional continuous numerical benchmarks. The universal connections with different tools or systems are out of consideration. Most of them require the users to familiar with the optimization process of intelligent optimization algorithms and make improvement based on complex program code. The whole design and comparison process are still quite time consumption.

2.8.2 Configuration

On the other hand, though some existing libraries integrate multiple typical intelligent optimization algorithms, they still have difficulty to adapt to the frequent changing manufacturing problems with efficient research ability. Likewise, as for the whole digital manufacturing process, optimization problems exist in every module. But they are quite different with diverse stages and environments. The dynamic adaptation of intelligent optimization algorithms is needed during the procedure. Thus we do not only need the collaborative decision of several algorithms, but also need that the algorithms to be configured dynamically in the process of optimization in manufacturing systems. Not only the parameters should be configured, but also the operators, improving strategies and the whole algorithm should be adjusted dynamically.

The existing studies consider rarely about the flexible configuration of intelligent optimization algorithm. To the adaptive processing, the studies of recent years focus more on specific problems and design adaptive processing mechanisms or improving strategies in a single algorithm structure. These mechanisms and structures are mostly unchangeable during optimization. When apply such algorithm in dynamic optimization, comprehensive high efficient searching in all stages can not be realized. For changing environments or properties in problem, engineers have to stop the decision process, store the middle data and redesign the algorithm again. It results in not only a loss of time, but also a repeated waste of program code.

Therefore, though intelligent optimization algorithm has the versatility in structure, with specific problems, it still has weaknesses in adaptability and scalability. The existing intelligent optimization algorithm library only code and store various algorithms independently and it is hard to achieve dynamic configuration. To break through the limitations in the collaborative multi-stage optimizations in manufacturing systems, the studies on the dynamic adjustment and configuration of intelligent optimization algorithms and the scalable combination of the algorithms for complex problems are quite important.

2.8.3 Parallelization

With the development of large-scale cluster systems and distributed computing technology, the design of parallelization mode of intelligent optimization algorithm and its application in large-scale projects are extended gradually, and have achieved some effect. From the perspective of algorithm structure, intelligent optimization algorithm generally can performs collaborative search with population provision, thus it has natural parallelism. From the perspective of parallel computing environment, not only the problems can be separated and solve parallel, but also various solution spaces can be searched in parallel. The combination of intelligent optimization algorithm and the parallelized technologies can save much time for the optimization of various complex projects.

Now more and more design and application studies in the parallelization of intelligent optimization algorithm have shown up. Most of them carry out the research from three key elements: parallel topology, individual migration time and number of individuals to be migrated. And the parallelization in intelligent optimization algorithm design is primarily based on population provision, in which the topology is the main consideration. However, different types of parallel intelligent optimization algorithms show different performance in different problems and environments with the influences of the three key elements. Currently, although there are many parallel intelligent optimization algorithms for specific problems, the scalability and effectiveness are still to be improved and verified.

Therefore, driven by high performance computing technologies, to further improve the solving efficiency of intelligent optimization algorithms in manufacturing, their parallelization design with the consideration of topology and individual migration elements and the extend application of parallel optimization are urgent.

2.8.4 Executing as Service

Similarly, with the spread of service-oriented manufacturing and computing modes, in distributed manufacturing process, the users mostly get the support from invoking the remote services with different functions through network. Thus multiple services invoked by multi-users can realize resource sharing and high efficient collaborative design and production in manufacturing. Now, as the generalization of the concept of service, some simple algorithms have been encapsulated as services and provided in service center. When users are invoking these services, they only need to concern about the inputs and outputs. At the same time, the users expect the transparency of service computing to achieve real-time monitoring, intelligent interruption and dynamically adjustment. From the perspective of the application of intelligent optimization algorithm in manufacturing, the idea of encapsulating these algorithms as services for different users is already realizable. Currently there are some algorithm libraries which provide typical intelligent optimization algorithms in the form of services on the internet for the users to invoke. However, flexibly and efficiently solving complex optimization problems in the manufacturing systems by intelligent optimization algorithms in form of services has not been implemented and there still exist many challenges in improving the performances of algorithm services in wide area network.

Firstly, from the perspective of application, to different complex manufacturing problems, the users not only have to know the characteristics of the encapsulated intelligent optimization algorithms, but also need to combine them flexibly. It also requires the encapsulated algorithms to be highly configurable. Secondly, from the perspective of process, because of the requirements to the transparent service computing, the intelligent optimization algorithms have to be split into operators and encapsulated as fine-grained modules. Moreover, the clear display and control of the population-based iterative evolutionary process are also needed. These key elements rarely studied but crucial to actual projects. Therefore, the adaptability and flexibility design and implementation of service-oriented intelligent optimization algorithms is very imperative.

2.9 Summary

In this chapter, we mainly talked about the development of the application of intelligent optimization algorithms in manufacturing. From the optimization of manufacturing system and process, the problems are divided into numerical function optimization, parameter optimization, detection and classification, combinatorial scheduling and multi-disciplinary optimization according to the characteristics and objectives of the variables. And we summarized the main challenges faced in solving different kinds of problems. From the perspective of optimization ways, we elaborated the six common optimization design methods, namely, model-based method, empirical-based method, prediction-based method, simulation-based method, tool-based method and advanced-computing-technology-based method, and draw the importance of intelligent optimization algorithm in the large-scale complex optimization problems of manufacturing systems. Developed so far, many intelligent optimization algorithms and improved strategies are proposed for specific problems. However, there are still many challenges in the wide application and targeted design of the intelligent optimization algorithms. Thus, we analyzed those challenges one by one, and gave the major trend of studies and development of intelligent optimization algorithms in manufacturing system and process.

After the above analysis, in the studies and development of intelligent optimization algorithms, though the studies of improvement and design from different perspectives produce a large number of intelligent operators and improving strategies, the requirements of the three kinds of users are far from being fully satisfied. Its main problems are:

(1) Lack of uniformly platform for collection and comparison. Though living in a world with abounds of numerous intelligent algorithms, we still have no idea which one is the best for a particular set of problems due to the lack of integrated centers which are capable of performing standard testing and comparing.

(2) Long design and implementing process. Owning to the sophisticated investigating and programming process of searching and implementing new operators, engineers may limit the usage into several basic algorithms. Such inertia is likely to carry some risks since the generally-used algorithms may not fit well to the given problem, and at the same time, those valuable findings may lose the chance of being used.

(3) Lack of extension and much repetition. Though more and more innovative practices have been designed to enhance algorithms' performance for application-specific demands and general benchmarks, most of them still lack effective testing and extended using. Meanwhile, due to universal unawareness of existing resources, repetitive works have been done in the process of developing same or similar algorithm for different problems in different areas, leading to huge resource wastes and time consuming.

(4) Lack of theoretical foundation. As the scale of the problem (i.e. the solution space) increases, the solving accuracy of the problem drops significantly. Because of the inherent randomness of the algorithm and the searching direction far from completely developed, the balance between the algorithm in exploration and exploitation is still hard to handle. And also, there are not many studies on the theory, convergence and time complexity of intelligent optimization algorithms. The efficiency of the algorithm is obtained by a large number of tests, and lack theoretical foundation.

Moreover, as the high performance computing and service-oriented technologies developing fast, and the scale of combinatorial optimization problems in existing industrial application growing rapidly, the improvement and the application of intelligent optimization algorithms still have long way to go. Maximizing the application of the existing intelligent optimization algorithms and the improvements of them in engineering practices is a difficulty to be solved.

Thus the biggest difficulty turned into: How to effectively employ huge amounts of existing intelligent algorithms and their improvements for various types of uses.

To fully exploit the existing intelligent algorithms and quickly obtain flexible synergies and improvements, new dynamic configuration methods for intelligent optimization algorithms (DC-IOA) is proposed in our work. Based on separated operator modules, three-level configurations, i.e. parameter-based configuration, operator-based configuration and algorithm-based configuration are exploited. Various types of algorithms can be collected and well re-produced by not only arbitrarily combining different operator modules, but also arbitrarily splicing multiple algorithms according to the operational generation separation. Specifically, it can solve the above questions mainly from the following two aspects.

(1) In the view of algorithm employment. Informative workflow with operator modules referring to basic and typical algorithms is provided to algorithm beginners. A friendly interface, where parameter setting, customized operator selecting, and dynamic algorithm combining are involved and provided to senior researchers. In the meantime, various existing algorithms and their improving strategies with only configurable parameters are prepared to for algorithm employers with direct use.
(2) In the view of algorithm development. Comparisons among different strategies are given based on some general benchmarks for algorithm beginner. The encapsulated operator modules and customized interfaces to allow imports of the operators or algorithms and then to support further tests are available for senior researchers. Also recommended algorithms with typical portfolios of operators are given according to the type and feature of submitted problem for algorithm employers.

Starting from the second part of this book, we are going to introduce the theory, design process and application of the configuration method for intelligent optimization algorithm in detail.

References

1. Jung DS, Kim CY (2013) Finite element model updating on small-scale bridge model using the hybrid genetic algorithm. Struct Infrastruct Eng 9(5):481–495
2. Keshavarz S, Khoei AR, Molaeinia Z (2013) Genetic algorithm-based numerical optimization of powder compaction process with temperature-dependent cap plasticity model. Int J Adv Manuf Technol 64:1057–1072
3. Miguel LFF, Lopez RH, Miguel LFF (2013) Multimodal size, shape, and topology optimization of truss structures using the firefly algorithm. Adv Eng Softw 56:23–37
4. Herencia JE, Haftka RT, Balabanov V (2013) Structural optimization of composite structures with limited number of element properties. Struct Multi Optim 47(2):233–245
5. Debout P, Chanal H, Duc E (2011) Tool path smoothing of a redundant machine: application to automated fiber placement. Comput Aided Des 43(2):122–132
6. Hur SH (2011) Modeling and control of a plastic film manufacturing web process. IEEE Trans Industr Inf 7(2):171–178
7. Okaeme NA (2013) Hybrid bacterial foraging optimization strategy for automated experimental control design in electrical drives. IEEE Trans Industr Inf 9(2):668–678
8. Kumar KS, Paulraj G (2011) Genetic algorithm based deformation control and clamping force optimization of workpiece fixture system. Int J Prod Res 49(7):1903–1935
9. Chan KY, Kwong CK, Tsim YC (2010) A genetic programming based fuzzy regression approach to modeling manufacturing processes. Int J Prod Res 48(7):1967–1982
10. Wu Q, Gao L, Li X, Zhang C, Rong Y (2013) Applying an electromagnetism-like mechanism algorithm on parameter optimization of a multi-pass milling process. Int J Prod Res 51(6):1777–1788
11. Yin F, Mao H, Hua L (2011) A hybrid of back propagation neural network and genetic algorithm for optimization of injection molding process parameters. Mater Des 32(6):3457–3464
12. Abd-Elazim SM, Ali ES (2013) A hybrid particle swarm optimization and bacterial foraging for optimal power system stabilizers design. Electr Power Energ Syst 46:334–341
13. Yildiz AR (2013) Cuckoo search algorithm for the selection of optimal machining parameters in milling operations. Int J Adv Manuf Technol 64(1–4):55–61
14. Hassan LH, Moghavvemi M, Almurib HAF, Steinmayer O (2013) Application of genetic algorithm in optimization of unified power flow controller parameters and its location in the power system network. Electr Power Energ Syst 46:89–97
15. Chung IY, Liu WX, Cartes DA, Moon SI (2011) Control parameter optimization for multiple distributed generators in a microgrid using particle swarm optimization. Eur Trans Electr Power 21(2):1200–1216
16. Rai JK, Brand D, Slama M, Xirouchakis P (2013) Optimal selection of cutting parameters in multi-tool milling operations using a genetic algorithm. Int J Prod Res 49(10):3045–3068
17. Juang CF, Chang PH (2010) Designing fuzzy-rule-based systems using continuous ant colony optimization. IEEE Trans Fuzzy Syst 18(1):138–149
18. Rao RV, Kalyankar VD (2013) Parameter optimization of modern machining processes using teaching-learning-based optimization algorithm. Eng Appl Artif Intell 26:524–531
19. Mukherjee R, Chakraborty S (2013) Selection of the optimal electrochemical machining process parameters using biogeography-based optimization algorithm. Int J Adv Manuf Technol 64(5–8):781–791
20. Mehrjoo M, Khaji N, Ghafory-Ashtiany M (2013) Application of genetic algorithm in crack detection of beam-like structures using a new cracked euler-bernoulli beam element. Appl Soft Comput 13(2):867–886
21. Pastrana S, Mitrokotsa A, Orfila A, Peris-Lopez P (2012) Evaluation of classification algorithms for intrusion detection in MANETs. Knowl Based Syst 36:217–225

22. Ak R, Li Y, Vitelli V, Zio E, Droguett EL, Jacinto CMC (2013) NSGA-II-trained neural network approach to the estimation of prediction intervals of scale deposition rate in oil & gas equipment. Expert Syst Appl 40(4):1205–1212
23. Kuo RJ, Tseng WL, Tien FC, Liao TW (2012) Application of an artificial immune system-based fuzzy neural network to a RFID-based positioning system. Comput Ind Eng 63(4):943–956
24. Yang CC (2011) Constructing a hybrid kansei engineering system based on multiple affective responses: application to product form design. Comput Ind Eng 60(4):760–768
25. del Castillo-Gomariz R, Garcia-Pedrajas N (2012) Evolutionary response surfaces for classification: an interpretable model. Appl Intell 37(4):463–474
26. Lin HY (2013) Feature selection based on cluster and variability analyses for ordinal multi-class classification problems. Knowl Based Syst 37:94–104
27. Kaufmann P, Glette K, Gruber T, Platzner M, Torresen J, Sick B (2013) Classification of electromyograpic signals: comparing evolvable hardware to conventional classifiers. IEEE Trans Evol Comput 17(1):46–63
28. dos Santos Fonseca WA, Bezerra UH, Nunes MVA, Barros FGN, Moutinho JAP (2013) Simultaneous fault section estimation and protective device failure detection using percentage values of the protective devices alarms. IEEE Trans Power Syst 28(1):170–180
29. Li R, Seckiner SU, He D, Bechhoefer E, Menon P (2012) Gear fault location detection for split torque gearbox using AE sensors. IEEE Tans Syst Man Cybern Part C Appl Rev 42(6):1308–1317
30. Behdad M, Barone LC, Bennamoun M, French T (2012) Nature-inspired techniques in the context of fraud detection. IEEE Trans Syst Man Cybern Part C Appl Rev 42(6):1273–1290
31. Wu J, Zhang WY, Zhang S, Liu YN, Meng XH (2013) A matrix-based Bayesian approach for manufacturing resource allocation planning in supply chain management. Int J Prod Res 51(5):1451–1463
32. Stadtler H, Sahling F (2013) A lot-sizing and scheduling model for multi-stage flow lines with zero lead times. Eur J Oper Res 225(3):404–419
33. Niu SH, Ong SK, Nee AYC (2012) An enhanced ant colony optimizer for multi-attribute partner selection in virtual enterprises. Int J Prod Res 50(8):2286–2303
34. Chen L, Langevin A, Lu Z (2013) Integrated scheduling of crane handling and truck transportation in a maritime container terminal. Eur J Oper Res 225(1):142–152
35. Rafiei H, Rabbani M, Alimardani M (2013) Novel bi-level hierarchical production planning in hybrid MTS/MTO production context. Int J Prod Res 51(5):1331–1346
36. Bortfeldt A, Homberger J (2013) Packing first, routing second—a heuristic for the vehicle routing and loading problem. Comput Oper Res 40:873–885
37. Xing KYY, Han LB, Zhou MC, Wang F (2012) Deadlock-free genetic scheduling algorithm for automated manufacturing systems based on deadlock control policy. IEEE Trans Syst Man Cybern Part B Cybern 42(3):603–615
38. Xu Y, Wang L (2011) Differential evolution algorithm for hybrid flow-shop scheduling problems. J Syst Eng Electron 22(5):794–798
39. Wu C, Barnes D (2010) Formulating partner selection criteria for agile supply chains: a dempster-shafer belief acceptability optimization approach. Int J Prod Econ 125(2):284–293
40. Li X, Zhang Y (2012) Adaptive hybrid algorithms for the sequence-dependent setup time permutation flow shop scheduling problem. IEEE Trans Autom Sci Eng 9(3):578–595
41. Campana EF, Fasano G, Peri D (2012) Penalty function approaches for ship multidisciplinary design optimization (MDO). Eur J Ind Eng 6(6):765–784
42. Farmani MR, Roshanian J, Babaie M, Zadeh PM (2012) Multi-objective collaborative multidisciplinary design optimization using particle swarm techniques and fuzzy decision making. Proc Inst Mech Eng Part C J Mech Eng Sci 226(9):2281–2295
43. Remouchamps A, Bruyneel M, Fleury C, Grihon S (2011) Application of a bi-level scheme including topology optimization to the design of an aircraft pylon. Struct Multi Optim 44(6):739–750

44. Rafique AF, He LS, Zeeshan Q, Kamran A, Nisar K (2011) Multidisciplinary design and optimization of an air launched satellite launch vehicle using a hybrid heuristic search algorithm. Eng Optim 43(3):305–328
45. Wu DD, Zhang YD, Wu DX, Olson DL (2010) Fuzzy multi-objective programming for supplier selection and risk modeling: a possibility approach. Eur J Oper Res 3(1):774–787
46. Ciavotta M, Minella G, Ruiz R (2013) Multi-objective sequence dependent setup times permutation flows hop: a new algorithm and a comprehensive study. Eur J Oper Res 227(2):301–313
47. Li X, Du G (2013) BSTBGA: a hybrid genetic algorithm for constrained multi-objective optimization problems. Comput Oper Res 40(1):282–302
48. Nourmohammadi A, Zandieh M (2011) Assembly line balancing by a new multi-objective differential evolution algorithm based on TOPSIS. Int J Prod Res 49(10):2833–2855
49. Omkar SN, Senthilnath J, Khandelwal R, Narayana Naik GN, Gopalakrishnan S (2011) Artificial bee colony (ABC) for multi-objective design optimization of composite structures. Appl Soft Comput 11(1):489–499
50. Manupati VK, Deo S, Cheikhrouhou N, Tiwari MK (2013) Optimal process plan selection in networked based manufacturing using game-theoretic approach. Int J Prod Res 50(18):5239–5258
51. Arikat J, Farahandi MH, Ahmadizar F (2012) Multi-objective genetic algorithm for cell formation problem considering cellular layout and operations scheduling. Int J Comput Integr Manuf 25(7):625–635
52. Naderi B, Aminnayeri M, Piri M, Hairi Yazdi MH (2012) Multi-objective no-wait flow shop scheduling problems: models and algorithms. Int J Prod Res 50(10):2592–2608
53. Santana-Quintero LV, Hernandez-Diaz AG, Molina J, Coello CAC, Caballero R (2010) DEMORS: a hybrid multi-objective optimization algorithm using differential evolution and rough set theory for constrained problems. Comput Oper Res 37(3):470–480
54. Minella G, Ruiz R, Ciavotta M (2011) Restarted iterated pareto greedy algorithm for multi-objective flow shop scheduling problems. Comput Oper Res 38(11):1521–1533
55. Pishvaee MS, Farahani RZ, Dullaert W (2010) A memetic algorithm for bi-objective integrated forward/reverse logistics network design. Comput Oper Res 37(6):1100–1112
56. Xu G, Yang ZT, Long GD (2012) Multi-objective optimization of MIMO plastic injection molding process conditions based on particle swarm optimization. Int J Manuf Technol 58(5–8):521–531
57. Puisa R, Streckwall H (2011) Prudent constraint-handling technique for multi objective propeller optimization. Optim Eng 12(4):657–680
58. Nguyen TT, Yao X (2012) Continuous dynamic constrained optimization—the challenges. IEEE Trans Evol Comput 16(6):769–786
59. Qu BY, Suganthan PN (2013) Constrained multi-objective optimization algorithm with an ensemble of constraint handling methods. Eng Optim 43(4):403–416
60. Jiang H, Ren ZL, Xuan JF, Wu XD (2013) Extracting elite pairwise constraints for clustering. Neurocomputing 99(1):124–133
61. Kellegoz T, Toklu B (2012) An efficient branch and bound algorithm for assembly line balancing problems with parallel multi-manned workstations. Comput Oper Res 39(12):3344–3360
62. Mallipeddi R, Suganthan PN (2010) Ensemble of constraint handling techniques. IEEE Trans Evol Comput 14(4):561–579
63. da Silva EK, Barbosa HJC, Lemonge ACC (2011) An adaptive constraint handling technique for differential evolution with dynamic use of variants in engineering optimization. Oper Eng 12(1–2):31–54
64. Wu CY, Ku CC, Pai HY (2011) Injection molding optimization with weld line design constraint using distributed multi-population genetic algorithm. Int J Adv Manuf Technol 52(1–4):131–141
65. Pan QK, Ruiz R (2013) A comprehensive review and evaluation of permutation flow shop heuristics to minimize flow time. Comput Oper Res 40(1):117–128

66. Prakash A, Chan FTS, Deshmukh SG (2011) FMS scheduling with knowledge based genetic algorithm approach. Expert Syst Appl 38(4):3161–3171
67. Xin B, Chen J, Peng ZH, Dou LH, Zhang J (2011) An efficient rule-based constructive heuristic to solve dynamic weapon target assignment problem. IEEE Trans Syst Man Cybern Part A Syst Humans 41(3):598–606
68. Prakash A, Chan FTS, Deshmukh SG (2012) Application of knowledge-based artificial immune system (KBAIS) for computer aided process planning in CIM context. Int J Prod Res 50(18):4937–4954
69. Xing LN, Chen YW, Wang P, Zhao QS, Xiong J (2010) Knowledge-based ant colony optimization for flexible job shop scheduling problems. Appl Soft Comput 10(3):888–896
70. Hsu YY, Tai PH, Wang MW, Chen WC (2011) A knowledge-based engineering system for assembly sequence planning. Int J Adv Manuf Technol 55(5–8):763–782
71. Li BM, Xie SQ, Xu X (2011) Recent development of knowledge-based systems, methods and tools for one-of-a-kind production. Knowl Based Syst 24(7):1108–1119
72. Yu ZW, Wong HS, Wang DW (2011) Neighborhood knowledge-based evolutionary algorithm for multi objective optimization problems. IEEE Trans Evol Comput 15(6):812–831
73. Karimi H, Rahmati SHA, Zandieh M (2012) An efficient knowledge-based algorithm for the flexible job shop scheduling problem. Knowl Based Syst 36:236–244
74. Weiss D (2010) Feature-based spline optimization in CAD. Struct Multi Optim 42(4):619–631
75. Sandberg M, Tyapin L, Kokkolaras M, Isaksson O, Aidanpaa JO, Larsson T (2011) A knowledge-based master-model approach with application to rotating machinery design. Concurrent Eng Res Appl 19(4):295–305
76. Bui LT, Michalewicz Z, Parkinson E, Abello MB (2012) Adaptation in dynamic environments: a case study in mission planning. IEEE Trans Evol Comput 16(2):190–209
77. Zhang XM, Zhu LM, Zhang D, Ding H, Xiong YL (2012) Numerical robust optimization of spindle speed for milling process with uncertainties. Int J Mach Tools Manuf 61:9–19
78. Bekker J, Aldrich C (2011) The cross-entropy method in multi-objective optimization: an assessment. Eur J Oper Res 211(1):112–121
79. Van Hentenryck P, Bent R, Upfal E (2010) Online stochastic optimization under time constraints. Ann Oper Res 177(1):151–183
80. Shnits B (2012) Multi-criteria optimization-based dynamic scheduling for controlling FMS. Int J Prod Res 50(21):6111–6121
81. Dou JP, Wang XS, Wang L (2012) Machining fixture layout optimization under dynamic conditions based on evolutionary techniques. Int J Prod Res 50(15):4294–4315
82. Lee DK, Starossek U, Shin SM (2010) Topological optimized design considering dynamic problem with non-stochastic structural uncertainty. Struct Eng Mech 36(1):79–94
83. Pillac V, Gendreau M, Gueret C, Medaglia AL (2013) A review of dynamic vehicle routing problems. Eur J Oper Res 225(1):1–11
84. Siddiqui S, Azarm S, Gabriel S (2011) A modified benders decomposition method for efficient robust optimization under interval uncertainty. Struct Multi Optim 44(2):259–275
85. Zhang Y, Gong DW, Zhang JH (2013) Robot path planning in uncertain environment using multi objective particle swarm optimization. Neurocomputing 103:172–185
86. Pan F, Nagi R (2010) Robust supply chain design under uncertain demand in agile manufacturing. Comput Oper Res 37(4):668–683
87. Aghezzaf EH, Sitompul C, Najid NM (2010) Models for robust tactical planning in multi-stage production systems with uncertain demands. Comput Oper Res 37(5):880–889
88. Sivakumar K, Balaurugan C, Ramabalan S (2012) Evolutionary multi-objective concurrent maximization of process tolerances. Int J Prod Res 50(12):3172–3191
89. Zhao P, Zhou HM, Li Y, Li DQ (2010) Process parameters optimization of injection modeling using a fast strip analysis as a surrogate model. Int J Adv Manuf Technol 49(9–12):949–959

90. Duffuaa SO, El-Ga'aly A (2013) A multi-objective optimization model for process targeting process sampling plans. Comput Ind Eng 64(1):309–317
91. Lau HCW, Jiang ZZ, Ip WH, Wang DW (2010) A credibility-based fuzzy location model with hurwicz criteria for the design of distribution systems in B2C e-commerce. Comput Ind Eng 59(4):873–886
92. Laili YJ, Tao F, Zhang L, Sarker BR (2012) A study of optimal allocation of computing resources in cloud manufacturing systems. Int J Adv Manuf Technol 63(5–8):671–690
93. Hwang R, Katayama H (2010) Integrated procedure of balancing and sequencing for mixed-model assembly lines: a multi-objective evolutionary approach 48(21):6417–6441
94. Jenab K, Ahi P (2010) Fuzzy quality feature monitoring model. Int J Prod Res 48(17):5021–5030
95. Luo GH, Noble JS (2012) An integrated model for cross dock operations including staging. Int J Prod Res 50(9):2451–2464
96. Kang M, Yoon K (2011) An improved best-first branch-and-bound algorithm for unconstrained two-dimensional cutting problems. Int J Prod Res 49(15):4437–4455
97. Almoustafa S, Hanafi S, Mladenovic N (2013) New exact method for large asymmetric distance-constrained vehicle routing problem. Eur J Oper Res 226(3):386–394
98. Letchford AM, Miller SJ (2012) Fast bounding procedures for large instances of the simple plant location problem. Comput Oper Res 39(5):985–990
99. Sabouni MTY, Logendran R (2013) A single machine carryover sequence-dependent group scheduling in PCB manufacturing. Comput Oper Res 40(1):236–247
100. Karsu O, Azizoglu M (2012) The multi-resource agent bottleneck generalized assignment problem. Int J Prod Res 50(2):309–324
101. Liu Q, Yang XD, Jing L, Li J, Li J (2012) A parallel scheduling algorithm for reinforcement learning in large state space. Front Comput Sci 6(6):631–646
102. Zhu J, Li XP, Shen WM (2012) A divide and conquer-based greedy search for two-machine no-wait job shop problems with makespan minimization. Int J Prod Res 50(10):2692–2704
103. Zhang HF, Zhou JZ, Fang N, Zhang R, Zhang YC (2013) An efficient multi-objective adaptive differential evolution with chaotic neuron network and its application on long-term hydropower operation with considering ecological environment problem. Int J Electr Power Energ 45(1):60–70
104. Alam MS, Islam MM, Yao X, Murase K (2012) Diversity guided evolutionary programming: a novel approach for continuous optimization. Appl Soft Comput 12(6):1693–1707
105. Perez E, Posada M, Herrera F (2012) Analysis of new niching genetic algorithms for finding multiple solutions in the job shop scheduling. J Intell Manuf 22(3):341–356
106. Kaveh A, Zolghadr A (2012) Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. Comput Struct 102:14–27
107. Boussaid I, Chatterjee A, Siarry P, Ahmed-Nacer M (2011) Two-stage update biogeography-based optimization using differential evolution algorithm (DBBO). Comput Oper Res 38(8):1188–1198
108. Chang PC, Huang WH, Ting CJ (2010) Dynamic diversity control in genetic algorithm for mining unsearched solution space in TSP problems. Expert Syst Appl 37(3):1863–1878
109. Chowdhury S, Tong WY, Messac A, Zhang J (2013) A mixed-discrete particle swarm optimization algorithm with explicit diversity-preservation. Struct Multi Optim 47(3):367–388
110. Tao F, Zhao DM, Hu YF, Zhou ZD (2010) Correlation-aware resource service composition and optimal-selection in manufacturing grid. Eur J Oper Res 201(1):129–143
111. Soulier B, Boucard PA (2013) A multiparametric strategy for the two step optimization of structural assemblies. Struct Multi Optim 47(4):539–553
112. Sakata S, Ashida F, Tanaka H (2010) Stabilization of parameter estimation for kriging-based approximation with empirical semivariogram. Comput Methods Appl Mech Eng 199(25–28):1710–1721

113. Bozca M, Fietkau P (2010) Empirical model based optimization of gearbox geometric design parameters to reduce rattle noise in an automotive transmission. Mech Mach Theory 45(11):1599–1612
114. Kondayya D, Krishna AG (2013) An integrated evolutionary approach for modeling and optimization of laser beam cutting process. Int J Adv Manuf Technol 65(1–4):259–274
115. Paralikas J, Salonitis K, Chryssolouris G (2010) Optimization of roll forming process parameters—a semi-empirical approach. Int J Adv Manuf Technol 47(9–12):1041–1052
116. Raja SB, Baskar N (2011) Particle swarm optimization technique for determining optimal machining parameters of different work piece materials in turning operation. Int J Adv Manuf Technol 54(5–8):445–463
117. Ding TC, Zhang S, Wang YW, Zhu XL (2010) Empirical models and optimal cutting parameters for cutting forces and surface roughness in hard milling of AISI H13 steel. Int J Adv Manuf Technol 51(1–4):45–55
118. Chou CJ, Chen LF (2012) Combining neural networks and genetic algorithms for optimizing the parameter design of the inter-metal dielectric process. Int J Prod Res 50(7):1905–1916
119. Sedighi M, Afshari D (2010) Creep feed grinding optimization by an integrated GA-NN system. J Intell Manuf 21(6):657–663
120. Lou P, Liu Q, Zhou ZD, Wang HQ, Sun SX (2012) Multi-agent-based proactive-reactive scheduling for a job shop. Int J Adv Manuf Technol 59(1–4):311–324
121. Hsieh L, Chen WS, Liu CH (2011) Motion estimation using two-stage predictive search algorithms based on joint spatio-temporal correlation information. Expert Syst Appl 38(9):11608–11623
122. Norouzi A, Hamedi M, Adineh VR (2012) Strength modeling and optimizing ultrasonic welded parts of ABS-PMMA using artificial intelligence methods. Int J Adv Manuf Technol 61(1–4):135–147
123. Zhu HP, Liu FM, Shao XY, Zhang GJ (2010) Integration of rough set and neural network ensemble to predict the configuration performance of a modular product family. Int J Prod Res 48(24):7371–7393
124. Jeung HS, Choi HG (2012) Particle swarm optimization in multi-stage operations for operation sequence and DT allocation. Comput Ind Eng 62(2):442–450
125. Chan KY, Kwong CK, Tsim YC (2010) Modelling and optimization of fluid dispensing for electronic packaging using neural fuzzy networks and genetic algorithms. Eng Appl Artif Intell 23(1):18–26
126. Guan YJ, Yuan GP, Sun S, Zhao GQ (2013) Process simulation and optimization of laser tube bending. Int J Adv Manuf Technol 65(1–4):333–342
127. Pirard F, Iassinovski S, Riane F (2011) A simulation based approach for supply network control. Int J Prod Res 49(24):7205–7226
128. Azadeh A, Negahban A, Moghaddam M (2012) A hybrid computer simulation-artificial neural network algorithm for optimization of dispatching rule selection in stochastic job shop scheduling problems. Int J Prod Res 50(2):551–566
129. Varthanan P, Murugan N, Kumar G, Parameswaran S (2012) Development of simulation-based AHP-DPSO algorithm for generating multi-criteria production-distribution plan. Int J Adv Manuf Technol 60(1–4):373–396
130. Varthanan PA, Murugan N, Kumar GM (2012) A simulation based heuristic discrete particle swarm algorithm for generating integrated production-distribution plan. Appl Soft Comput 12(9):3034–3050
131. Azadeh A, Moghaddam M, Geranmayeh P, Naghavi A (2010) A flexible artificial neural network—fuzzy simulation algorithm for scheduling a flow shop with multiple processors. Int J Adv Manuf Technol 50(5–8):699–715
132. Zhang Y, Agogino AM (2012) Hybrid evolutionary optimal MEMS design. Int J Adv Manuf Technol 63(1–4):305–317
133. Kafashi S (2011) Integrated setup planning and operation sequencing (ISOS) using genetic algorithm. Int J Adv Manuf Technol 56(5–8):589–600

134. Vishnupriyan S, Majumder MC, Ramachandran KP (2011) Optimal fixture parameters considering locator errors. Int J Prod Res 49(21):6343–6361
135. Kumar KS, Paulraj G (2012) Geometric error control of workpiece during drilling through optimization of fixture parameter using a genetic algorithm. Int J Prod Res 50(12):3450–3469
136. Huang X, Xie YM, Jia B, Li Q, Zhou SW (2012) Evolutionary topology optimization of periodic composites for extremal magnetic permeability and electrical permittivity. Struct Multi Optim 46(3):385–398
137. Tsai CC, Huang HC, Chan CK (2011a) Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation. IEEE Trans Industr Electron 58(10):4813–4821
138. Tsai CC, Huang HC, Lin SC (2011b) FPGA-based parallel DNA algorithm for optimal configurations of an omnidirectional mobile service robot performing fire extinguishment. IEEE Trans Industr Electron 58(3):1016–1026
139. Bozejko W, Uchronski W, Wodecki M (2010) Parallel hybrid metaheuristics for the flexible job shop problem. Comput Ind Eng 59(2):323–333
140. Goncalves JF, Resende MGC (2012) A parallel multi-population biased random-key genetic algorithm for a container loading problem. Comput Oper Res 39(2):179–190
141. Jin JY, Crainic TG, Lokketangen A (2012) A parallel multi-neighborhood cooperative tabu search for capacitated vehicle routing problems. Eur J Oper Res 222(3):441–451
142. Boyer V, El Baz D, Elkihel M (2012) Solving knapsack problems on GPU. Comput Oper Res 39(1):42–47
143. Ostrosi E, Fougeres AJ, Ferney M, Klein D (2012) A fuzzy configuration multi-agent approach for product family modeling in conceptual design. J Intell Manuf 23(6):2565–2586
144. Misra R, Mandal C (2010) Minimum connected dominating set using a collaborative cover heuristic for ad hoc sensor networks. IEEE Trans Parallel Distrib Syst 21(3):292–302