Front. Comput. Sci., 2017, 11(2): 347–357
DOI 10.1007/s11704-016-6154-6

RESEARCH ARTICLE

Optimizing product manufacturability in 3D printing

Yu HAN, Guozhu JIA (✉)

School of Economics and Management, Beihang University, Beijing 100191, China

Abstract 3D printing has become a promising technique for industry production. This paper presents a research on the manufacturability optimization of discrete products under the influence of 3D printing technology. For this, we first model the problem using a tree structure, and then formulate it as a linear integer programming, where the total production time is to be minimized with the production cost constraint. To solve the problem, a differential evolution (DE) algorithm is developed, which automatically determines whether traditional manufacturing methods or 3D printing technology should be used for each part of the production. The algorithm is further quantitatively evaluated on a synthetic dataset, compared with the exhaustive search and alternating optimization solutions. Simulation results show that the proposed algorithm can well combine the traditional manufacturing methods and 3D printing technology in production, which is helpful to attain optimized product design and process planning concerning manufacture time. Therefore, it is beneficial to provide reference of the widely application and further industrialization of the 3D printing technology.

Keywords 3D printing, manufacturability, optimization, discrete products, differential evolution algorithm

1 Introduction

According to the definition published by US 3D Printing technology committee, 3D printing is a process of joining materials to make objects from 3D model data, usually layer upon layer, as opposed to subtractive manufacturing methodologies[1)]. It gains the objection by directly manufacturing the 3D physical entity which is entirely the same as the correspondent model, and covers all processes including the shaping, manufacturing and assembling of products. Charles Hull developed 3D entity printing using digital data from stereo-lithography technology in 1984, which leads to an improvement of 3D printing technology for over 30 years. Nowadays, 3D printing has become more sophisticated and entered an era of laser printing, while materials employed are varied as ceramic and metals instead of resin only. The broad prospect of 3D printing has been admitted so that it has been listed in "Made in China 2025"[2)] and "Industrie 4.0"[3)] and considered as a key technology in the new industrial revolution.

In practical production environment, 3D printing and traditional manufacturing methods have their own strengths. Traditional ways are out of reach with those individualized and complicated products which could be directly attained by 3D printing [1]. Those products which cost much in primary stage (such as mold cost) and have long manufacturing cycles in traditional methods will also lead to a 3D printing decision [2]. Other than that, 3D printing may have no priories compared with traditional methods that are able to attain scale economies effects[4)]. Therefore, it is still imperative to combine 3D printing with traditional methods in a long period to decrease total cost and production cycle practically, even if

Received March 16, 2016; accepted June 7, 2016

E-mail: jiaguozhu@buaa.edu.cn

1) ASTM(2013).ASTM F2792-12a Standard terminology for additive manufacturing technologies. Retrieved from http://www.ASTM.org/Standards/F2792.htm
2) China's State Council. Made in China 2025, 2015
3) The Industrie 4.0 working group. Industrie 4.0, 2013
4) Separating Facts from Fiction About 3D Printing: Knowledge at Wharton (http://knowledge.wharton.upenn.edu/article.cfm?articleid=3322)

the former has broken all technical barriers. Thus, it can be seen that the optimized decision under the combination of the two is of significant importance.

At present, researches on 3D printing mainly focus on technical fields such as manufacturing craftwork, processing materials and quality control, as well as aspects of the application review and development prospects. Among them, foreign scholars including Atzeni and Salmi [3], Dolphin [4], Cesaretti et al. [5] studied 3D printing technical elements, development tendency and application fields. Domestic scholars as Yan and Qi [6], Lu and Li [7], Wang [8] launched research on its domain, the present and future development as well as technical principles. When talking about the influence on production management, Tuck et al. [9], Holmstrom et al. [10], Nyman and Sarlin [11], Rayna and Striukova [12] have papers or monographs on the concept, form and management strategy of supply chain importing 3D printing. However, none of them addressed the principles of the optimized decision under the combination of 3D printing and traditional manufacturing ways.

This paper focuses on manufacturability optimization of discrete products targeting production time based on 3D printing, which involves the determination of how to product each component of the object, under the constraint of the cost budget. To our best knowledge, this is the first work that studies the manufacturability optimization in 3D printing with the cost constraint. The research addresses the problem by formulating it as a linear integer programming based on the graph representation and transforms the selection of the manufacture way into the constraints on the feasible solution. The DE algorithm is adopted to optimize the problem efficiently.

Since it is infeasible to directly solve the above constrained linear integer programming problem, which is NP-hard in fact, heuristic algorithms serve as a promising solution, and there are a number of speeding up techniques [13,14] that can help achieve fast computation. During the past two decades, a lot of swarm intelligence algorithms, which are inspired by swarm intelligence of objects such as birds, ants, bees, etc., have been proposed and studied, among which are particle swarm optimization (PSO), ant colony optimization (ACO), and the artificial bee colony (ABC) algorithm. Evolutionary algorithms (EAs) [15], defined as a set of generic metaheuristic optimization algorithms that are inspired by the biological evolution of animal species, have been successfully applied to complex numerical optimization problems in diverse fields. Unlike classic optimization methods that rely on the gradient of the problem being optimized, EAs do not make any assumption about the underlying fitness landscape and perform well approximating solutions to all types of problems through searching very large spaces of candidate solutions [16]. As a typical representative of EAs, the differential evolution (DE) [17, 18] algorithm was firstly proposed by Storn and Price [19] and has been proven to be a simple but efficient global optimizer in the continuous search domain. As a promising optimization algorithm, DE has been successfully applied to engineering problems including economic dispatch, power distribution, neural network training, and manufacturing process optimization. In this paper, we follow the idea and design a DE algorithm for the manufacturability optimization. Specifically, the DE algorithm solves the optimization problem through maintaining a population of candidate solutions and creating new individuals using mechanisms such as reproduction, mutation, and recombination, where the solutions with better fitness are kept to the next iteration.

The optimized manufacturability of products manufactured under the combination of 3D printing and traditional ways is researched in-depth quantitatively. The proposed scheme clearly figures the way to increase the design and production efficiency as well as reduce waste of raw materials, the optimization from design along with production process is therefore accomplished. For the convenience of reading, all the variables mentioned in Sections 2 and 3 are listed in Table 1.

Table 1 Variable notation list

Variable	Meaning
$\mathbf{S}$	Production node set
S_i	Production node
N	Number of production nodes
M	Number of leaf nodes on the bottom layer
$\mathbf{E}$	Directed-edge set
E_{ij}	The directed-edge directed from S_i to S_j
$\mathbf{G}$	The product structure
c_i^{p}	The 3D printing cost of S_i
$\mathbf{c}^{\mathrm{p}}$	The 3D printing cost of $\mathbf{S}$
$\mathbf{c}$	Assembly cost
$\mathbf{t}^{\mathrm{p}}$	The production time
$\mathbf{p}$	The chosen craftwork
$p_i = 0$	Traditional manufacturing method
$p_i = 1$	3D printing
$\mathbf{t}^{\mathrm{e}}$	Equivalent assembly time
$\mathbf{c}^{\mathrm{e}}$	Equivalent assembly cost
C	The upper limit of total production cost
$\mathbf{n}$	The quantity of leaf nodes included in each node

2 Background description

2.1 Definition

Process planning decides the way of manufacturing products, which is essential during preparation and acts as the basis of all production processes. The craftwork optimization on account of cost factors is drawing attention in modern manufacture management [20]. Yet, the whole problem is changed when bringing in 3D printing technology. When manufacturing discrete product using traditional methods, components that stay at the end nodes in the product structure tree are firstly considered so that the final product could be assembled by them in order. Therefore, the only thing that changed after introducing a new technology is the manufacturing method of a certain component. The optimization of product cost and manufacture cycle in the production stage still largely relies on process planning in traditional manufacturing methods [21]. What is greatly different for 3D printing methods from traditional ones is that any component on any nodes or layers in the structure tree could be produced directly. Consequently, the production and assembling processes of components below the improved one are simplified. It means that the product structure is changed simultaneously when choosing 3D printing technology. The product is re-designed in this way, rather than merely considering process planning. Hence, the craftwork design of the whole manufacture system should be re-defined and re-considered after 3D printing technology are imported.

Manufacturability is about the optimization of given manufacturing resources that meets the requirement of customers, concerning cost, time, fabrication property, assembly property and others. Manufacturability assessment plays a key role in concurrent engineering (CE) project [22]. Since 3D printing technology is brought into the manufacturing system, product design and process planning turn into a process of parallel interaction that significantly illustrates and implements the fabrication-targeted concept in CE project. Manufacturability is therefore employed to research the craft optimization considering costs under the influence of 3D printing.

This paper adopts graph and matrix theory to describe the problem and a DE based approach is proposed to achieve the optimization. The optimization problem is built targeting minimum production time so that the manufacturability optimization based on 3D printing meeting certain cost constraints could be solved.

Meanwhile, this paper has three hypotheses in order to simplify the question and gives prominence to the theme:

- Components can be manufactured by either traditional methods or 3D printing, and make no difference in the prospect of technical characteristic.
- The 3D printing and traditional methods are generally discussed ignoring specific manufacturing craftwork.
- Production cost and the time are the only factors being considered when discussing the manufacturability optimization using two different methods.

2.2 Graph representation

In order to precisely describe the product structure and its changes after using 3D printing technology, a graphic structure model is employed [23–26] which has been proved very powerful in the selection problem. The product structure is defined and abstracted as a digraph constituted by several nodes and directed edges.

Assuming there are N nodes on the product structure tree of a certain product. A node set $\mathbf{S} = \{S_i\}$ is used, where S_i denotes a production node i. For each node, both traditional manufacturing method and 3D printing could be applied. When conducting traditional ways, the component on a typical node is assembled by components on its child nodes and components on leaf nodes could be directly obtained. When 3D printing is adopted, components could be printed directly without concerning child nodes. Leaf nodes on the bottom layer correspond to dissembled components, whose quantity is accounted as M.

In the graphic structure, the directed-edge is defined as $\mathbf{E}$, in which E_{ij} is the directed-edge directed from S_i to S_j in the node set S meaning the manufacturing order (S_j to S_i). The node-set and the directed-edge set could be combined to form the graphic structure as $G = \{S, E\}$. A typical product structure graph is shown in Fig. 1, among which nodes S_3, S_4 and S_5 represent components at the bottom layer, S_1 stands for the final product, and other nodes are intermediate assembled components.

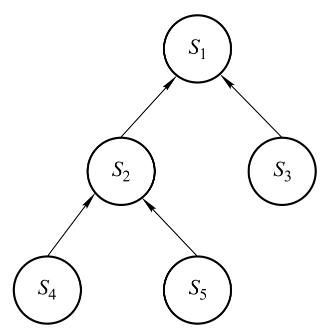

Fig. 1 Typical product structure graph

In the above product structure graph, there are two kinds of nodes: leaf nodes and intermediate nodes. The layer of leaf nodes represents the components that cannot be assembled and could be manufactured by 3D printing or traditional methods. The intermediate layer corresponds to the intermediate components or the final product, which could be assembled by components in the layer of child nodes, or directly 3D printed.

The model of graphic structure helps to abstract the whole product structure and clarify the logical relationship among all components. Yet it cannot be directly used to process the optimization problem, which is just the reason for introducing the matrix as an algebra method to describe the relations among all nodes in the digraph.

Supposing that 3D printing is available for all nodes, the 3D printing cost is defined as c_i^{p} for a certain node S_i and expressed in terms of vectors as $\mathbf{c}^{\mathrm{p}}$. However, the leaf nodes may cover costs generated from both 3D printing and traditional manufacturing processes. Virtual nodes are therefore employed to distinguish two different fabrication methods. Each leaf node has directed connection to a correspondent virtual node. In the earlier graph, virtual nodes S_6, S_7, S_8 have been introduced and directed-connected to S_4, S_5, S_3, respectively. The production cost in the virtual nodes is the cost from traditional ways so that other nodes cover merely production cost generated by 3D printing. The production structure graph containing virtual nodes is shown in Fig. 2.

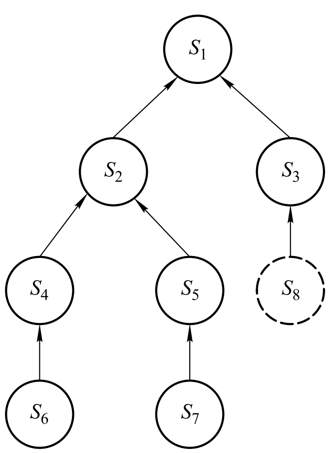

Fig. 2 Production structure graph containing virtual nodes

Apart from the production cost, assembly cost should be raised during traditional manufacture processes for intermediate nodes, which is represented by vector $\mathbf{c}$.

As for the production time, assuming that of each component S_i is t_i^{p} and could be expressed in terms of vectors as $\mathbf{t}^{\mathrm{p}}$. Virtual nodes, similarly, cover production time of their correspondent leaf nodes using traditional ways, while only the time of 3D printing is concluded in other nodes. Assembly time is further considered as well and assumed as vector $\mathbf{t}$ for each node.

3 Manufacturability optimization modeling

3.1 Time-targeted formulation

This paper aims at optimizing the total production time involving production and assembling processes. The time is attempted to be deduced as short as possible while meeting the production cost constraints. The most difficult part of the manufacturing optimization is how to formally describe it and formulate as a mathematical model. Therefore, in this section, we mainly introduce how to formulate the manufacturing optimization problem as a constrained linear integer programming problem, which can be solved using some optimization techniques.

Production time: take p_i to represent the craft chosen for node S_i. $p_i = 0$ means that traditional manufacturing method is employed while $p_i = 1$ stands for 3D printing. As the vector $\mathbf{p}$ is applied as the chosen craftwork, the total production time is $\mathbf{p}^{\mathrm{T}}\mathbf{t}^{\mathrm{p}}$ consequently.

Assembly time: the assembly time referred for the components is actually the time required above the nodes choosing 3D printing method (or leaf nodes). As shown in the following graph, when S_2 and S_8 are mandatory components, the assembly time of S_1 is then the time consumed to assemble the component S_1 from S_2 and S_8. The total assembly time could be calculated by taking account of equivalent assembly time $\mathbf{t}^{\mathrm{e}}$ of each node in advance (deep tree traversal is normally adopted). Specifically, the assembly time of child nodes should be the total time needed from father nodes connected along with the path till root nodes. Without loss of generality, the assembly time of father nodes is equally distributed to correspondent child nodes to achieve better illustration. As shown in Fig. 3, each node has new accumulated assembly time $\mathbf{t}^{\mathrm{e}}$ so that the total assembly time is $\mathbf{p}^{\mathrm{T}}\mathbf{t}^{\mathrm{e}}$.

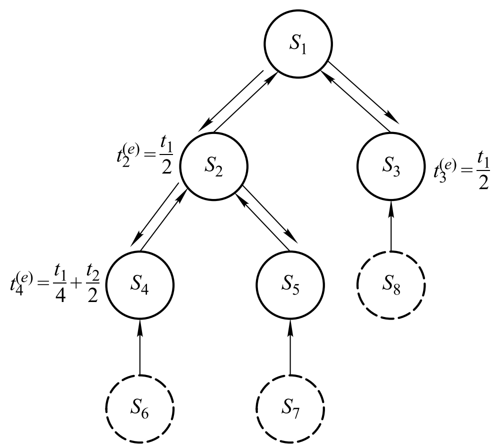

Fig. 3 Equivalent assembly time calculation including virtual nodes

Hence, the total production time is the sum of production time and assembly time, which is:

$$\mathbf{p}^{\mathrm{T}}(\mathbf{t}^{\mathrm{p}} + \mathbf{t}^{\mathrm{e}}). \tag{1}$$

3.2 Cost constraints

From the above analysis, it is obvious that the final production cost is consisted of both production and assembly cost.

Production cost: the sum of the chosen component manufacturing time by either 3D printing or traditional ways,

$$\mathbf{p}^{\mathrm{T}}\mathbf{c}^{\mathrm{p}}. \tag{2}$$

Assembly cost: the assembly cost referred for the components is actually the cost required above the nodes from 3D printing (or leaf nodes). As shown in the following graph, when S_2 and S_8 are mandatory components, the assembly cost of S_1 is then the cost consumed to assemble the component S_1 from S_2 and S_8. The assembly cost is denoted by $\mathbf{c}$, the total assembly cost is therefore calculated by taking account of equivalent assembly cost $\mathbf{c}^{\mathrm{e}}$ of each node in advance. Specifically, the assembly cost of child nodes should be the total cost needed from father nodes connected along with the path till root nodes. Without loss of generality, the assembly cost of father nodes is equally distributed to correspondent child nodes to achieve better illustration. As in Fig. 4, each node has new accumulated assembly cost $\mathbf{c}^{\mathrm{e}}$. Consequently, the assembly cost is:

$$\mathbf{p}^{\mathrm{T}}\mathbf{c}^{\mathrm{e}}. \tag{3}$$

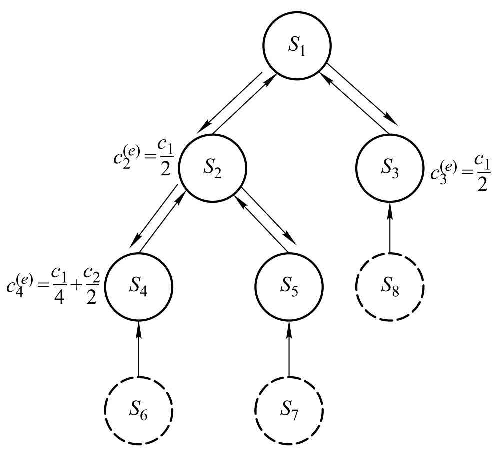

Fig. 4 Equivalent assembly cost calculation including virtual nodes

The total cost is the sum of production cost and assembly cost,

$$\mathbf{p}^{\mathrm{T}}(\mathbf{c}^{\mathrm{p}} + \mathbf{c}^{\mathrm{e}}). \tag{4}$$

In the present research, the total production cost is constrained by the upper limit C,

$$\mathbf{p}^{\mathrm{T}}(\mathbf{c}^{\mathrm{p}} + \mathbf{c}^{\mathrm{e}}) < C. \tag{5}$$

3.3 Feasible solution constraints

The feasible solution vector $\mathbf{p}$ chosen in previous problems should meet the following requirements:

- These nodes are able to form a complete product, which means that the node chosen should equivalently include all leaf nodes. As illustrated in Fig. 5, S_1 could be completed by producing S_2 using 3D printed and S_3 using traditional method. Meanwhile, components S_4 and S_5 make up S_2. S_1 is actually constituted by components S_3, S_4, S_5 as a result, in which way, all leaf nodes are covered. Taking the above constraints into consideration, the quantity of leaf nodes (components) included in each node could be calculated by traversing upward from the bottom layer (leaf nodes), expressing as vector $\mathbf{n}$. The constraint is then indicated as

$$\mathbf{p}^{\mathrm{T}}\mathbf{n} = M. \tag{6}$$

- The child nodes and the father nodes of a chosen node cannot be simultaneously considered. It means that a chosen node produced from 3D printing could not be influenced by its child nodes, meanwhile, the father nodes of which will be assembled by the mentioned 3D-printed component rather than manufactured by 3D printing. In Fig. 5, other nodes except for S_2 and S_8 will not be involved in the 3D printing process.

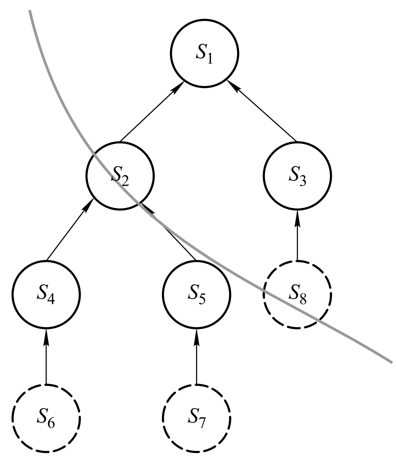

Fig. 5 Process planning diagram

A reachable matrix is introduced to describe the constraint. Since the graphic structure model could only illustrate the quantity of nodes and the directed relation between every two nodes, the information staying in the graph could be re-recorded through a reachable matrix. An element R_{ij} in the reachable matrix $\mathbf{R}$ is defined by

$$R_{ij} = \begin{cases} 1, & \text{if } S_i \text{ and } S_j \text{ are directed-reachable;} \\ 0, & \text{if } S_i \text{ and } S_j \text{ are not directed-reachable.} \end{cases} \tag{7}$$

The subscripts i and j traverse all nodes. The calculation of the reachable matrix of the whole typical product structure in Fig. 1 could be:

$$\mathbf{R} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{8}$$

In this way, adjacent matrix and reachable matrix could give all information about the relations among all nodes in the graphic structure model.

Based on the reachable matrix, the mathematical description of the second requirement of choosing vector $\mathbf{p}$ is as follows: for any two nodes S_i and S_j, if there exists an assembly relationship between (reachable, $R_{ij} = 1$ or $R_{ji} = 1$), they cannot be manufactured through 3D printing simultaneously; i.e., if a node is employed with 3D printing method, those nodes connected should not be chosen.

Hence,

$$p_i(R_{ij} + R_{ji})p_j = 0, \tag{9}$$

and its matrix form is:

$$\mathbf{p}^{\mathrm{T}}(\mathbf{R} + \mathbf{R}^{\mathrm{T}})\mathbf{p} = 0. \tag{10}$$

3.4 Model optimization

The final optimized model on account of all previous research:

$$\begin{aligned} \min_{p}\ & \mathbf{p}^{\mathrm{T}}(\mathbf{t}^{\mathrm{p}} + \mathbf{t}^{\mathrm{e}}) \\ \text{s.t.}\ \ & \mathbf{p}^{\mathrm{T}}(\mathbf{c}^{\mathrm{p}} + \mathbf{c}^{\mathrm{e}}) \leqslant C, \\ & \mathbf{p}^{\mathrm{T}}\mathbf{n} = M, \\ & \mathbf{p}^{\mathrm{T}}(\mathbf{R} + \mathbf{R}^{\mathrm{T}})\mathbf{p} = 0, \\ & \mathbf{p} \in \{0, 1\}^{N+M}. \end{aligned} \tag{11}$$

Thus, the manufacturability optimization problem proposed in this paper is formulated in a linear integer programming with a quadratic constraint on the feasible solution. There is no close-form solution to the above problem. Besides, it is quite hard to find the optimal solution using the traditional programming techniques. Next, we will propose an optimization solution via differential evolution.

4 Optimization via differential evolution

To solve the optimization problem in Eq. (11), the DE algorithm evolves a population of N individuals towards the global optimum of the problem at hand, in which each individual is denoted by a D-dimensional vector $\mathbf{X}_{i,t} = x_i^1, x_i^2, \ldots, x_i^j, \ldots, x_i^D$, in which $i = 1, 2, \ldots, N$ corresponds to each individual in the population, $j = 1, 2, \ldots, D$ corresponds to each element in the solution, and t is the generation index. For the purpose of better covering the whole search space, the population is randomly initialized within the lower bound $\mathbf{X}_{\mathrm{low}} = x_{\mathrm{low}}^1, x_{\mathrm{low}}^2, \ldots, x_{\mathrm{low}}^D$ and the upper bound $\mathbf{X}_{\mathrm{up}} = x_{\mathrm{up}}^1, x_{\mathrm{up}}^2, \ldots, x_{\mathrm{up}}^D$ respectively as

$$x_{i,0}^j = x_{\mathrm{low}}^j + \mathrm{rand}(0, 1)(x_{\mathrm{up}}^j - x_{\mathrm{low}}^j), \quad j = 1, 2, \ldots, D, \tag{12}$$

where $x_{i,0}^j$ is the jth element of individual i and rand(0, 1) is a random number drawn from a uniformly distributed space within [0, 1]. Typically, the DE algorithm is consisted of the following three operators, namely mutation, crossover, and selection, which are detailed as follows.

4.1 Mutation

Following the initialization stage, the mutation operation aims to create a mutate vector $\mathbf{M}_{i,t}$ with respect to each target individual $\mathbf{X}_{i,t}$ in the current population. According to different mechanisms, there exist many different mutation strategies and the most widely used one “DE/rand/1” is given as follows:

$$\mathbf{M}_{i,t} = \mathbf{X}_{r_1^i,t} + F(t)(\mathbf{X}_{r_2^i,t} - \mathbf{X}_{r_3^i,t}), \tag{13}$$

where r_1, r_2, r_3 are three random integers within $[1, N]$ that are mutually exclusive from each other and for the mutation operator of each individual, the random numbers are generated once. The scaling coefficient $F(t) > 0$ is a free parameter used to control the mutation process. In this paper, a time variant F is employed, where $F_t = F_{\max} - t(F_{\max} - F_{\min})/T_{\max}$, with $F_{\max}(F_{\min})$ being the maximum (minimum) scaling factor and $T_{\max}$ being the maximum iteration number respectively.

4.2 Crossover

Following the phase of mutation operation, the crosser operator is implemented to produce a trial vector $\mathbf{T}_{i,t}$ from each pair of target vector $\mathbf{X}_{i,t}$ and $\mathbf{M}_{i,t}$. By taking advantage of a random number $rand_{i,t}^j$ within the range of [0, 1], the basic crossover operator applies a binomial function defined as

$$t_{i,t}^j = \begin{cases} m_{i,t}^j, & \text{if } rand_{i,t}^j < CR \text{ or } j = j_{rand}; \\ x_{i,t}^j, & \text{otherwise}, \end{cases} \tag{14}$$

where $m_{i,t}^j$, $t_{i,t}^j$, and $x_{i,t}^j$ are respectively the jth element corresponding to the ith mutate vector $\mathbf{M}_{i,t}$, trial vector $\mathbf{T}_{i,t}$, and target vector $\mathbf{X}_{i,t}$ in the tth generation, $CR \in [0,1]$ is a predefined constant used to control the fraction of values that are copied from the mutate vectors, and $j_{rand} \in [0,1]$ is a randomly selected number. In this way, the crossover operator generates a trial vector $\mathbf{T}_{i,t}$, the jth element of which is equal to that of $\mathbf{M}_{i,t}$ if the random number $rand_{i,t}^j$ is less than the crossover probability CR or $j = j_{rand}$. Otherwise, the element is copied from the same position of the target vector $\mathbf{X}_{i,t}$.

4.3 Selection

After the crossover operation, the selection operator is applied to keep better solutions to the next generation by evaluating and comparing each pair of $\mathbf{X}_{i,t}$ and $\mathbf{T}_{i,t}$. Firstly, each $\mathbf{T}_{i,t}$ is checked whether it is within the feasible space between $\mathbf{X}_{low}$ and $\mathbf{X}_{up}$. If the feasibility is violated, the vector will be re-initialized within the feasible space. After this, each vector $\mathbf{X}_{i,t}$ is evaluated using the fitness function Fit as follows:

$$\begin{aligned} &\text{Fit}\,(\mathbf{X}_{i,t}) = f_t(\mathbf{p}) + f_c(\mathbf{p}) + f_s(\mathbf{p}), \\ &\mathbf{p} \in \{0,1\}^D, p_j = \begin{cases} 0, & \text{if } x_{i,t}^j < 0.5; \\ 1, & \text{if } x_{i,t}^j \geqslant 0.5, \end{cases} \\ &f_t(\mathbf{p}) = \mathbf{p}^{\mathrm{T}}(\mathbf{t}^{\mathrm{p}} + \mathbf{t}^{\mathrm{e}}), \\ &f_c(\mathbf{p}) = \begin{cases} 0, & \text{if } \mathbf{p}^{\mathrm{T}}(\mathbf{c}^{\mathrm{p}} + \mathbf{c}^{\mathrm{e}}) \leqslant C; \\ L, & \text{else}, \end{cases} \\ &f_s(\mathbf{p}) = \begin{cases} 0, & \text{if } \mathbf{p}^{\mathrm{T}}\mathbf{n} = m \text{ and } \mathbf{p}^{\mathrm{T}}(\mathbf{R} + \mathbf{R}^{\mathrm{T}}); \\ L, & \text{else}, \end{cases} \end{aligned} \tag{15}$$

where L is a large enough integer predefined to punish candidate solutions that do not satisfy the constraints of cost requirement as well as feasible production selection. Based on the fitness value obtained above, the selection operator is implemented by comparing Fit($\mathbf{X}_{i,t}$) and Fit($\mathbf{T}_{i,t}$), of which the better is copied to the next generation. Specifically, the selection operator is described by

$$\mathbf{X}_{i,t+1} = \begin{cases} \mathbf{X}_{i,t}, & \text{if Fit}(\mathbf{X}_{i,t}) \geqslant \text{Fit}(\mathbf{T}_{i,t}); \\ \mathbf{T}_{i,t}, & \text{otherwise}, \end{cases} \tag{16}$$

which means that a target vector remains in the next generation only when it is better than the corresponding trial vector. Otherwise, it will be replaced by the trial vector.

In the DE algorithm, the three operators defined above are repeated until the pre-defined termination criteria are satisfied. The specific steps for the problem in Eq. (11) based on DE are given as follows:

Step 1 Initialize the population according to Eq. (12) and set $t = 0$;

Step 2 For each individual in the population

Step 2.1 Generate a mutate vector $\mathbf{M}_{i,t}$ according to Eq. (13);

Step 2.2 Produce a trial vector $\mathbf{T}_{i,t}$ from each pair of target vector $\mathbf{X}_{i,t}$ and $\mathbf{M}_{i,t}$ according to Eq. (14);

Step 2.3 Select the better vector to the next generation according to Eq. (15);

Step 3 $t = t + 1$;

Step 4 If the termination criterion is satisfied, end and output the best solution, else go to Step 2.

5 Simulation results and analysis

In this section, the DE algorithm based solver for the 3D manufacturing problem is examined and compared with traditional search approaches, including particle swarm optimization (PSO) and brute-force search (BFS). Equality and inequality constraints are derived in Section 3 to describe the fitness of all the possible solutions. Both constraints of cost requirements and selection are treated as penalty factors in the optimization problem. To demonstrate the feasibility and effectiveness of the proposed approach for problems with different sizes, numerical simulations are carried out by using small-scale and large-scale test cases respectively, whose objectives are to find the best manufacturing strategies with least cost. In the first case, the population sizes of different algorithms are set to be 50 and the maximum iteration number is 150. For the second case, the population size and the maximum iteration number are both set to be 1 000. All individuals are randomly initialized in the feasible solution space according to Eq. (12). The specific parameter setting is given in Table 2.

Table 2 Parameter setting for DE

Parameter	Value	Meaning
N	50	Population size
$T_{\max}$	150	Maximum iteration number
$F_{\max}/F_{\min}$	0.8/0.4	Maximum (Minimum) scaling factor
CR	0.9	Crossover probability
C	14	Cost threshold
L	100 000	Penalty

5.1 Small case

In this testing case, an element tree with 15 nodes is used (Fig. 6), in which the number of original nodes $n = 10$ and that of virtual nodes is $m = 5$. To verify the effectiveness of the proposed algorithm, the problem setting is given as follows:

$$\mathbf{t}^e = [1.791, 0, 0.005, 0, 0.5621, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],$$
$$\mathbf{c}^e = [2.445, 0, 0.498, 0, 0.6200, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],$$
$$\mathbf{t}^p = [20.00, 0.866, 0.862, 0.497, 0.714, 0.467, 0.634, 0.232, 0.84, 0.677, 0.2485, 0.2335, 0.317, 0.42, 0.3385],$$
$$\mathbf{c}^p = [8.737, 0.675, 0.94, 0.146, 0.323, 0.978, 0.216, 0.0633, 0.907, 0.522, 0.073, 0.489, 0.108, 0.4535, 0.2610].$$

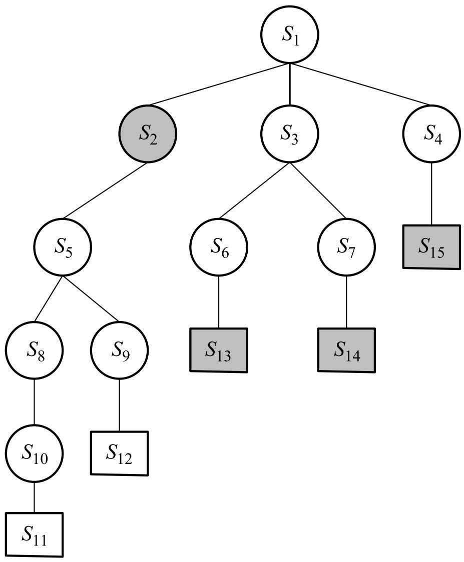

Fig. 6 Element tree (small scale)

In this case, the best manufacturing scheme is to produce nodes 4, 6, and 7 using conventional methods, 3D print node 2, and assemble all those parts to finally obtain the production of node 1. The result is demonstrated also in the element trees shown in Fig. 6 where the nodes are changed to gray color to correspond to 1 and the rest represent 0. The result comparison with BFS and PSO is presented in Table 3 and the evolution curves of DE and PSO are shown in Fig. 7. From these comparison results, we come to the following conclusions. Firstly, for small-scale cases with 15 element nodes, the BFS method is capable of finding the best solution in an acceptable computation time of around 6s. Secondly, the DE algorithm and PSO effective achieve the same optimal solution but with a computation time far less than BFS. Such an advantage should be attributed to the global search ability of those population based meta-heuristic algorithms. Thirdly, compared with PSO, DE has a better performance in terms of convergence speed.

Table 3 Result comparison for D=15

Index	BFS	PSO	DE
Fitness	3.737 5	3.737 5	3.737 5
CPU time/s	5.551	0.512	0.516
Solution	[0,1,0,0,0,0,0,0,0,0,0,0,1,1,1]		

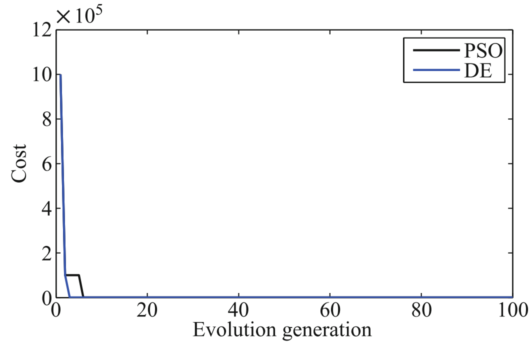

Fig. 7 Evolution curves of different algorithms for D=15

Thus, the structure of the product in the small case is changed to a new form as shown in Fig. 8, and the result represents the planning process under the optimal manufacturability.

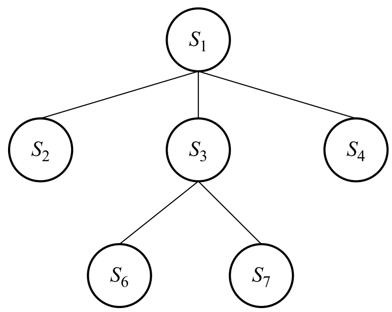

Fig. 8 New product structure of the small case

5.2 Large case

In this case, we test the effectiveness of the proposed algorithm using the element tree shown in Fig. 9, whose node number is extended to $D = 35$, with $n = 20$ and $m = 15$. Specifically, the problem setting is given as follows:

$$\mathbf{t}^p = [20.0000, 0.0393, 0.8927, 0.5117, 0.9655, 0.0337, 0.4788, 0.8883, 0.6108, 0.7066, 0.4445, 0.6172, 0.4525, 0.0940, 0.2991, 0.0462, 0.0998, 0.6195, 0.8810, 0.0226, 0.2559, 0.0169, 0.2394, 0.3054, 0.3533, 0.2223, 0.3086, 0.2263, 0.0470, 0.1495,$$

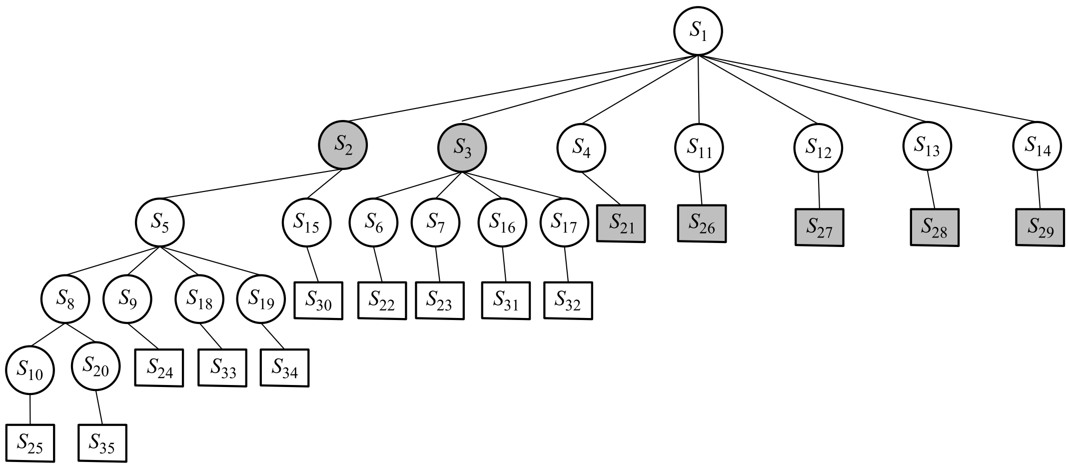

Fig. 9 Element tree (large scale)

$$
\begin{aligned}
&.0231, 0.0499, 0.3098, 0.4405, 0.0113],\\
\mathbf{t}^{\mathrm{e}} &= [5.4, 0.7873, 0.9080, 0, 0.0119, 0, 0, 0.5595, 0, 0, 0,\\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,\\
&\quad 0, 0, 0, 0, 0, 0],\\
\mathbf{c}^{\mathrm{p}} &= [10.0000, 0.7254, 0.0126, 0.2149, 0.6584, 0.9539,\\
&\quad 0.9320, 0.3155, 0.5374, 0.4546, 0.6245, 0.6070,\\
&\quad 0.6898, 0.0647, 0.8118, 0.9197, 0.4555, 0.6091,\\
&\quad 0.3833, 0.3037, 0.1074, 0.4770, 0.4660, 0.2687,\\
&\quad 0.2273, 0.3122, 0.3035, 0.3449, 0.0324, 0.4059,\\
&\quad 0.4599, 0.2277, 0.3046, 0.1917, 0.1518],\\
\mathbf{c}^{\mathrm{e}} &= [4.8000, 0.5346, 0.0162, 0, 0.7082, 0, 0, 0.6768, 0,\\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,\\
&\quad 0, 0, 0, 0].
\end{aligned}
$$

The comparison result and evolution curves are given in Table 4 and Fig. 9 respectively, and in Fig. 9 the nodes in red color correspond to 1 and the rest represent 0. With the growth of the dimension D from 15 to 35, the method of BFS suffers from exponential explosion, i.e., it takes about 3.4e+10 seconds (112 days) to traverse all the combinations. As a result, BFS is no longer able to provide a desirable solution in an acceptable period of time. On the contrary, the DE based solver can still search a near optimal solution in 197s, which is to manufacture nodes 21 and 26–29 in conventional ways, produce nodes 2 and 3 using 3D printing, and assemble all those elements to get the target production. The result is demonstrated in the element trees in Fig. 10. Compared with DE, the basic PSO does not converge to an acceptable solution in the current parameter setting. The advantage of the proposed approach is attributed to the global search ability in the whole feasible space as well as the usage of heuristic information taken from the evolution behavior of species in nature. As a consequence, the solver based on DE is capable of obtaining optimal or suboptimal solutions in a reasonable time, which leads to a better performance than the state-of-the-arts.

Table 4 Result comparison for D = 35

Index	BFS	DE
Fitness	–	9.08
CPU time/s	3.4e+10	197
Solution	[0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,1,1,1,0,0,0,0,0,0]	

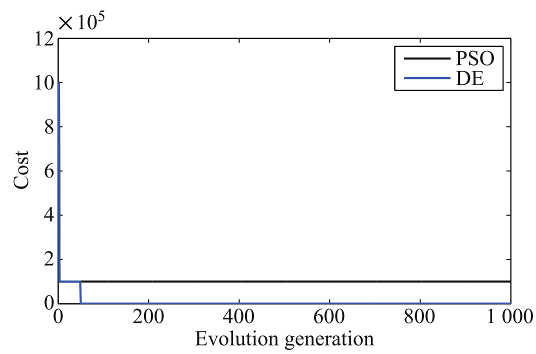

Fig. 10 Evolution curves of different algorithms for D = 35

Thus, the structure of the product in the large case is changed to a new form as shown in Fig. 11, and the result also represents the planning process under the optimal manufacturability.

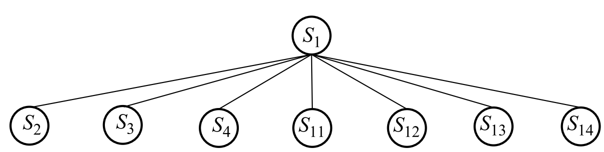

Fig. 11 New product structure of the large case

6 Conclusion

This paper addresses the manufacturability optimization of discrete products using 3D printing technology and traditional methods simultaneously and presents a novel approach based on DE. After modeling the problem using a tree structure, we formulate it as an optimization problem with both equality and inequality constraints. A DE based solver is proposed, whose effectiveness is compared with that of brute-force search and other optimization methods. Simulation results show that the proposed algorithm well combines traditional and 3D printing technology in production, which helps to improve production efficiency in terms of time and money. Therefore, it is beneficial to provide reference of the widely application and further industrialization of the 3D printing technology.

The status quo is that research is mostly focusing on the advantages of 3D printing qualitatively. A quantitative illustration of the irrelevance between the complexity of target component and the production time/cost in 3D printing is provided in this paper, which could further describe the solution of decreasing production cost while increasing efficiency in varying levels when manufacturing complicated products. At the same time, the model and algorithm proposed could solve the manufacturability optimization problem with a combination of both 3D printing and traditional ways in manufacturing processes to acquire specific product design and process planning scheme. The problem of developing the wide application and its further industrialized implementation of 3D printing has been primarily worked out, but now this research is still in its infancy. Also, the model which is built in this paper is relatively simple. In the next step, some new factors are planed to be involved in order to obtain more accurate optimization model that provides more precise guidance for the actual production.

Acknowledgements This paper was supported by the National Natural Science Foundation of China (Grant No. 71372007). We also would like to extend our sincere gratitude to the reviewers for their instructive advices and useful suggestions on this paper. Special thanks should go to the friends who have put considerable time and effort into their comments on the draft.

References

1. Oropallo W, Piegl L A. Ten challenges in 3D printing. Engineering with Computers, 2016, 32(1): 135–148
2. Barnatt C. 3D Printing: The Next Industrial Revolution. Charleston: Create Space Independent Publishing Platform, 2013
3. Atzeni E, Salmi A. Economics of additive manufacturing for end-usable metal parts. International Journal of Advanced Manufacturing Technology, 2012, 62(9–12): 1147–1156
4. Dolphin J. 3D printing: piracy or opportunity? Keeping Good Companies, 2012, 64(5): 300–303
5. Cesaretti G, Dini E, De Kestelier X, Colla V, Pambaguian L. Building components for an outpost on the lunar soil by means of a novel 3D printing technology. Acta Astronautica, 2014, 93(1): 430–450
6. Yan Y, Qi H. The connotation and application of rapid manufacturing. Aviation Manufacturing Technology, 2004: 26–29
7. Lu B H, Li D C. Development of additive manufacturing (3D printing) technology. Machine Building & Automation, 2013, 42: 1–4
8. Wang H M. Materials fundamental issues of laser additive manufacturing for high-performance large metallic components. Acta Aeronautica Et Astronautica Sinica, 2014, 35: 2690–2698
9. Tuck C, Hague R, Burns N. Rapid manufacturing: impact on supply chain methodologies and practice. International Journal of Services & Operations Management, 2006, 3(1): 1–22
10. Holmström J, Partanen J, Tuomi J, Walter M. Rapid manufacturing in the spare parts supply chain alternative approaches to capacity deployment. Journal of Manufacturing Technology Management, 2010, 21(6): 687–697
11. Nyman H J, Sarlin P. From bits to atoms: 3D printing in the context of supply chain strategies. In: Proceedings of the 47th IEEE Hawaii International Conference on System Sciences. 2014, 4190–4199
12. Rayna T, Striukova L. Adaptivity and rapid prototyping: how 3D printing is changing business model innovation. In: van den Berg B, van der Hof S, Kosta E, eds. 3D Printing, Vol 26. Hague: T.M.C. Asser Press, 2015, 167–182
13. Liu X L, Deng C, Lang B, Tao D C, Li X L. Query-adaptive reciprocal hash tables for nearest neighbor search. IEEE Transactions on Image Processing, 2015, 25(2): 907–919
14. Liu X L, Mu Y D, Zhang D C, Lang B, Li X L. Large-zcale unsupervised hashing with shared structure learning. IEEE Transactions on Cybernetics, 2015, 45(9): 1811–1822
15. Nachtigall K, Voget S. A genetic algorithm approach to periodic railway synchronization. Computers & Operations Research, 1996, 23(5): 453–463
16. Canellidis V, Giannatsis J, Dedoussis V. Evolutionary computing and genetic algorithms: paradigm applications in 3D printing process optimization. In: Tsihrintzis G A, Virvou M, Jain L C, eds. Intelligent Computing Systems, Vol 627. Berlin: Springer-Verlag, 2016, 271–298
17. Das S, Suganthan P N. Differential evolution: a survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation, 2011, 15(1): 4–31
18. Vasile M, Minisci E, Locatelli M. An inflationary differential evolution algorithm for space trajectory optimization. IEEE Transactions on Evolutionary Computation, 2011, 15(2): 267–281
19. Storn R, Price K. Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, 1997, 11(4): 341–359
20. Xu H, Li D. Review and outlook process planning research. Manufacturing Automation, 2008, 30: 1–7
21. Ba L, Li Y, Yang M, Liu Y. Integrated process planning and scheduling problem with consideration of assemble and transportation. Computer

Integrated Manufacturing Systems, 2015, 9: 2332–2342
22. Pan X. Principle and application of concurrent engineering. Beijing: Tsinghua University Press, 1998
23. Liao W, Guo Y, Cheng X. BOM modeling based on multi-color graph. Journal of Shandong University (Engineering Science), 2008: 70–75
24. Chao Y, Yang J, Wu Z. Automatic positioning design based on graph theory. Journal of Zhejiang University (Engineering Science), 2005, 39(12): 1925–1929
25. Liu X L, He J F, Lang B, Chang S F. Hash bit selection: a unified solution for selection problems in hashing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2013, 1570–1577
26. Shen F, Shen C, Shi Q, Hengel A, Tang Z, Shen H T. Hashing on nonlinear manifolds. IEEE Transactions on Image Processing, 2015, 24(6): 1839–1851

Yu Han is a PhD candidate of the School of Economics and Management, Beihang University, China. Her research domains focus mainly on operational process management. Her subject of the thesis is the influence of 3D printing on manufacturability of product design.

Guozhu Jia is a professor of the School of Economics and Management, Beihang University, China. His research domains focus mainly on the operational strategy for multi-variety and high-volume production, cellular manufacturing, and operational process management. Now he is interested in studying the influence of 3D printing on supply chain management and manufacturability of product design.