International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: http://www.tandfonline.com/loi/tprs20

A multi-population co-evolutionary genetic programming approach for optimal mass customisation production

Biao Yu, Han Zhao & Deyi Xue

To cite this article: Biao Yu, Han Zhao & Deyi Xue (2016): A multi-population co-evolutionary genetic programming approach for optimal mass customisation production, International Journal of Production Research, DOI: 10.1080/00207543.2016.1194538

To link to this article: http://dx.doi.org/10.1080/00207543.2016.1194538

Published online: 16 Jun 2016.

International Journal of Production Research, 2016
http://dx.doi.org/10.1080/00207543.2016.1194538

Taylor & Francis
Taylor & Francis Group

A multi-population co-evolutionary genetic programming approach for optimal mass customisation production

Biao Yu[a,b], Han Zhao[c] and Deyi Xue[a]*

[a]*Department of Mechanical and Manufacturing Engineering, University of Calgary, Calgary, Canada;* [b]*Institute of Applied Technology, Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, China;* [c]*School of Mechanical and Automotive Engineering, Hefei University of Technology, Hefei, China*

(*Received 15 December 2015; accepted 20 May 2016*)

Development of mass customised products demands various activities in the product development process, such as design, manufacturing process planning, manufacturing resource planning and maintenance process planning, to be considered and coordinated. In this research, a multi-population co-evolutionary genetic programming (MCGP) approach is introduced to identify the optimal design and its downstream product life cycle activities for developing mass customised product considering these different product life cycle activities and their relationships. In this research, two types of relationships between downstream product life cycle activities are considered: sequential relationships and concurrent relationships. The product design and its downstream life cycle descriptions are modelled by a multi-level graph data structure. These product life cycle descriptions are defined at two different levels: generic level for modelling the descriptions in a product family and specific level for modelling the descriptions of a customised product. The optimal design and its downstream life cycle activities are identified through the MCGP approach based on evaluations in different product life cycle aspects. Various methods have been developed to improve computation efficiency for the MCGP. Industrial case studies and comparative case studies have been implemented to demonstrate the effectiveness of the developed approach.

Keywords: mass customisation production; co-evolutionary genetic programming; optimisation; product design; process planning; resource planning

1. Introduction

Mass customisation production is a new manufacturing paradigm to design and produce customised products based on requirements from individual customers with the quality and efficiency for mass production (Tseng and Piller 2003). In the past decade, many methodologies have been developed and applied to solve various mass customisation problems (Fogliatto, da Silveira, and Borenstein 2012; Ferguson, Olewnik, and Cormier 2014). Since mass customisation production demands both the products and the production processes to be customised, these research activities have focused on the improvement of the flexibilities in both product design and product realisation process.

The flexibilities in product design include both the flexibilities in product configurations and the flexibilities in product parameters. Since flexibilities of parameters in a product design can be easily achieved through parametric design methods using the relations defined in equations, constraints and programmes, researches in mass customised product design have focused on flexibilities in product configurations. In this research area, a product family is usually used to model a group of products with similar structure (Jiao, Simpson, and Siddique 2007). The methodologies developed for modular design (Gershenson, Prasad, and Zhang 2004; Jose and Tollenaere 2005) and platform design (Jiao, Simpson, and Siddique 2007) are often used for product family design. When a specific requirement is defined by an individual customer, a specific design configuration of the product is then created from the product family descriptions. Many methods have been developed to identify specific product configurations from product families to satisfy customer requirements. We classify these methods into two categories: knowledge-based methods and optimisation-based methods.

In knowledge-based methods, artificial intelligence techniques such as rule-based methods, model-based methods and case-based methods were employed to model product families and customised products. McDermott (1982) employed production rules to model generic products and to obtain the specific products through rule-based reasoning.

*Corresponding author. Email: dxue@ucalgary.ca

When the rule base is huge, the knowledge base is very difficult to maintain and use. The main assumption behind model-based methods is the existence of a system's model, which usually consists of entities and relations among these entities. Typical model-based approaches include logic-based approach (McGuinness and Wright 1998), resource-based approach (Juengst and Heinrich 1998), ontology-based approach (Yang, Dong, and Miao 2008) and constraint-based approach (Schneeweiss and Hofstedt 2011). The model-based approaches have the advantages in robustness, reusability and compositionality, but the models are difficult to build in general. Tseng, Chang, and Chang (2005) proposed a case-based reasoning (CBR) approach for product configuration. In this CBR, the earlier configuration knowledge could be used. Identification of the proper case with similarity and modification to this old case based on new requirements are difficult. Although the knowledge-based methods sometimes can be used to identify feasible solutions to satisfy individual customer requirements, the solution is usually not an optimal one.

The optimisation-based methods aim at identifying the optimal customised products to maximise customers' satisfaction. Zhou, Lin, and Liu (2008) employed genetic algorithm to optimise a customer-driven product configuration for assemble-to-order manufacturing companies. Hong et al. (2008) developed a genetic programming method to identify the optimal product configuration and its parameters based on individual customer requirements on performance and costs in one-of-a-kind production. Liu, Lim, and Lee (2013) proposed a multi-objective evolutionary algorithm with an embedded feature of configuration incompatibility check to identify the optimal customised product from a product family. Goswami and Tiwari (2015) used mixed integer quadratic programming to identify the products considering commercial objectives of the enterprise and engineering-level constraints of the product. Kumar and Chatterjee (2015) employed mixed integer non-linear programming for the optimisation of production lines under monopolistic competition.

To produce the customised products, flexibilities in production processes should also be considered. Mass customisation considering both design and its downstream life cycle aspects is considered as a typical concurrent engineering design problem. In this area, Jiao et al. (2000) extended the generic bill-of-materials data structure, which was developed by Hegge and Wortmann (1991), into a generic bill-of-materials-and-operations data structure by considering variations in both product descriptions and process descriptions in a product family for mass customisation production. Zhang, Huang, and Rungtusanatham (2008) developed a mixed integer programming model that integrated both platform product design and material purchase decisions based on cost drivers that were sensitive to commonality and modularity. Pitiot et al. (2013) introduced a two-step approach to conduct concurrent product configuration and process planning, where the first step was to capture the customer or internal requirements interactively with a constraint-based approach and the second step was to identify the optimal solution through a multi-criteria constrained evolutional optimisation algorithm.

In our previous research, a mass customisation production approach considering both design and manufacturing aspects has been developed (Hong et al. 2008, 2010; Hong, Xue, and Tu 2010). In this research, design variations in a product family were modelled by an AND–OR tree (Hong et al. 2008). Each design node in the AND–OR tree was composed of design parameters for modelling a partial design solution. Each design node was also associated with a manufacturing process AND–OR graph that was composed of process nodes and their parameters for modelling variations of manufacturing processes to achieve the partial design solution (Hong et al. 2010). A customised product was created based on the requirements from an individual customer and modelled by its design configuration, design parameters, manufacturing process and process parameters. A design configuration for a customised product was modelled by a tree of design nodes with only AND relations and created from the product family through tree-based search. A manufacturing process was modelled by a graph of process nodes with sequential relationships. A multi-level optimisation method was developed to obtain the optimal solution where GP was used to identify the optimal design configuration and its manufacturing process, and numerical search was used to identify the optimal design and process parameter values.

Since various customised designs modelled by design configurations and design parameters can be achieved from the same requirements, and each of these designs can be produced using different manufacturing processes and process parameters, identification of the optimal design and its manufacturing process using the traditional evolutionary optimisation approach requires considerable computation effort. To improve optimisation efficiency, a co-evolutionary genetic programming (CGP) method was developed by (Hong et al. 2010). In this CGP method, two populations of species, i.e. a design population and a manufacturing population, were used to model design configurations and manufacturing processes. The individuals in the design population and the manufacturing population were created separately to improve the computation efficiency. For each design individual representing a design configuration, first whether this design individual can be achieved by one or more manufacturing processes described by manufacturing individuals was checked. Matching design and manufacturing individuals were grouped into pairs for representing design and manufacturing information of feasible customised products. The matched design and manufacturing individuals with better design and manufacturing evaluation measures were more likely to be duplicated into the next generation.

Despite this progress, the following problems need to be further investigated for mass customisation production.

(1) From the perspective of concurrent engineering design, more downstream product life cycle activities should be considered to obtain the optimal custom product design. In most of the presently developed concurrent design methods for mass customisation production, only design and one of the downstream product development life cycle aspects such as manufacturing process are considered (Hong et al. 2010; Pitiot et al. 2013). Quality of the mass customised product can be improved by incorporating considerations in more downstream product development life cycle activities such as manufacturing resource planning and maintenance process planning.

(2) From the perspective of computation efficiency, a more effective optimisation method to identity the optimal design and its downstream product life cycle activities is required. Due to the nature of combinatorial explosion of the problem (also called NP-hard problem), the traditional tree-based search methods are not effective to create feasible solutions when many product life cycle aspects are considered. Although (Hong et al. 2010) developed a CGP method to first create the design solutions and production process solutions separately and efficiently in two populations, and then to match the design solutions and the production process solutions to identify the feasible solutions considering both design and production process, this method is not effective when three or more populations are considered because fewer feasible solutions considering all populations can be identified through the matching process.

In this research, a multi-population CGP approach is developed to address the above two problems.

2. A multi-population co-evolutionary optimisation model for mass customisation production

2.1 *A multi-population co-evolutionary optimisation model*

In multi-population co-evolutionary optimisation, one generation of individuals is modelled by multiple populations as shown in Figure 1. Each population in one generation is composed of n individuals representing n partial solutions considering a particular aspect. Redundant solutions in one population are allowed. A complete solution considering all m aspects is achieved by selecting m matching individuals from all these m populations. Each individual is evaluated based on the evaluation criteria in that particular aspect. Evolution of populations from one generation to the next one is conducted through three operations: reproduction, crossover and mutation (Koza 1992). Reproduction allows the individuals with better evaluation measures to create more children individuals in the next generation. The individuals with poor evaluation measures are not used to create children individuals in the next generation. Crossover and mutation modify the current individuals randomly with small probabilities to seek for the opportunities of dramatic changes in the solution quality.

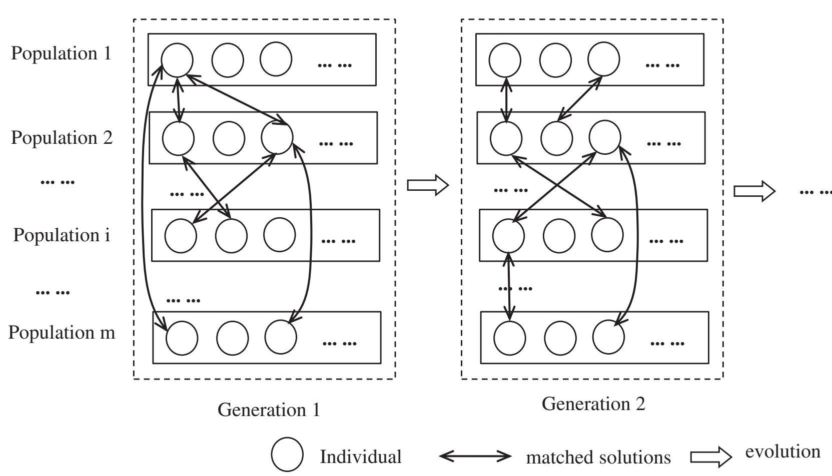


Figure 1. The multi-population co-evolutionary optimisation model.

Due to the difficulty in identification of a complete solution with matched individuals in all the populations in multi-population co-evolutionary optimisation, the following two strategies are considered for reproduction operations to improve the optimisation efficiency.

- The individuals with better evaluation measures in one population are selected to create more children individuals in the same population in the next generation.
- A few individuals with top evaluation measures in one population are selected to create the corresponding individuals in other populations in the next generation.

2.2 *An optimal mass customisation production model based on multi-population co-evolutionary optimisation*

Based on the generic multi-population co-evolutionary optimisation model introduced in Section 2.1, an optimal mass customisation production model is developed in this research to identify the optimal customised product design and its downstream life cycle activities. In this model, the relationships between the individuals in the downstream product life cycle populations are classified into two categories: sequential relationships and concurrent relationships. For example, the relationship between an individual of manufacturing process plan (B) and an individual of manufacturing resource plan (C) is a sequential relationship since the manufacturing resource plan is created based on the manufacturing process plan (Figure 2(a)), while the relationship between an individual of manufacturing process plan (B) and an individual of maintenance process plan (C) is a concurrent relationship since both are created from the individual of design (A) (Figure 2(b)).

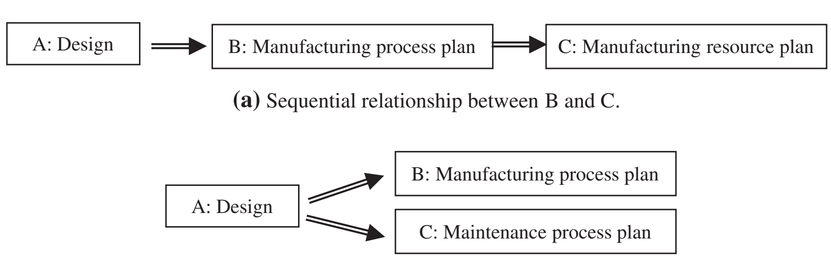
The evolutionary processes of individuals in three populations considering sequential and concurrent relationships between individuals in downstream life cycle aspects are shown in Figures 3 and 4, respectively. During the evolutionary process, an intermediate generation with three populations is created temporarily. Each population in the intermediate generation is composed of three sections. Individuals in Section I of an intermediate population are created based on the traditional reproduction operations such that the individuals with better evaluation measures in one population are selected to create more children individuals in the same population in the intermediate generation. Individuals in Sections II and III are created from the selected top individuals in other populations.

In Figure 3, the individuals in Populations of A, B and C are associated by sequential relationships. The top individuals in Population A are selected to create corresponding individuals in Population B, and then these created individuals in Population B are used to create individuals in Population C. In the same way, the top individuals in Population B are selected to create corresponding individuals in Population A and Population C, and the top individuals in Population C are selected to create corresponding individuals in Population B and then Population A.

In Figure 4, the individuals in Populations of B and C are associated by concurrent relationships. The top individuals in Population A are selected to create corresponding individuals in Population B and Population C. In the same way, the top individuals in Population B are selected to create corresponding individuals in Population A and then in Population C, and the top individuals in Population C are selected to create corresponding individuals in Population A and then Population B.

3. Modelling of product life cycle descriptions

Since different product life cycle aspects, including design, manufacturing process, manufacturing resource and maintenance process, are considered in this research, modelling of these different product life cycle aspects is discussed first.



(a) Sequential relationship between B and C.

(b) Concurrent relationship between B and C.

Figure 2. Sequential and concurrent relationships between individuals.

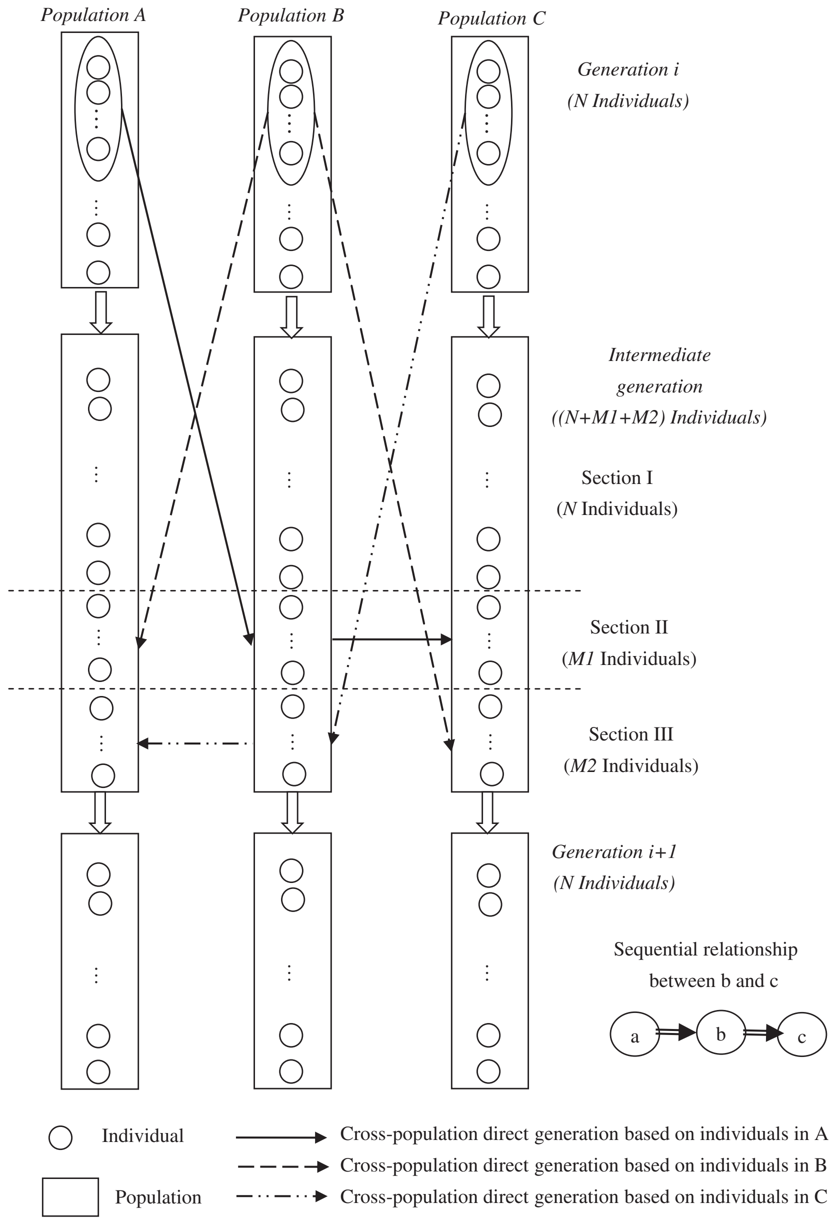


Figure 3. Evolution considering sequential relationships between individuals in different populations.

Modelling of product life cycle descriptions is conducted at two different levels: generic descriptions at product family level and specific descriptions at customised product level.

3.1 *Modelling of product design*

Variations of product configurations in a product family are modelled by an AND–OR tree as shown in Figure 5(a). Each design node in this AND–OR tree is composed of a set of design parameters. When all the sub-nodes need to be selected to support a super-node, all these sub-nodes are associated with an AND relation. For example, a gear-pair design node is composed of two sub-nodes, representing two gears, with an AND relation. When the super-node is supported by one of its sub-nodes, all these sub-nodes are associated with an OR relation. For example, the rotation-to-rotation transmission design node is supported by two sub-nodes, a gear pair and a pulley-belt pair, with an OR relation.

A customised product design is achieved from the product family modelled by an AND–OR tree through tree-based search. Figure 5(b) shows a customised product design created from the design AND–OR tree in a product family

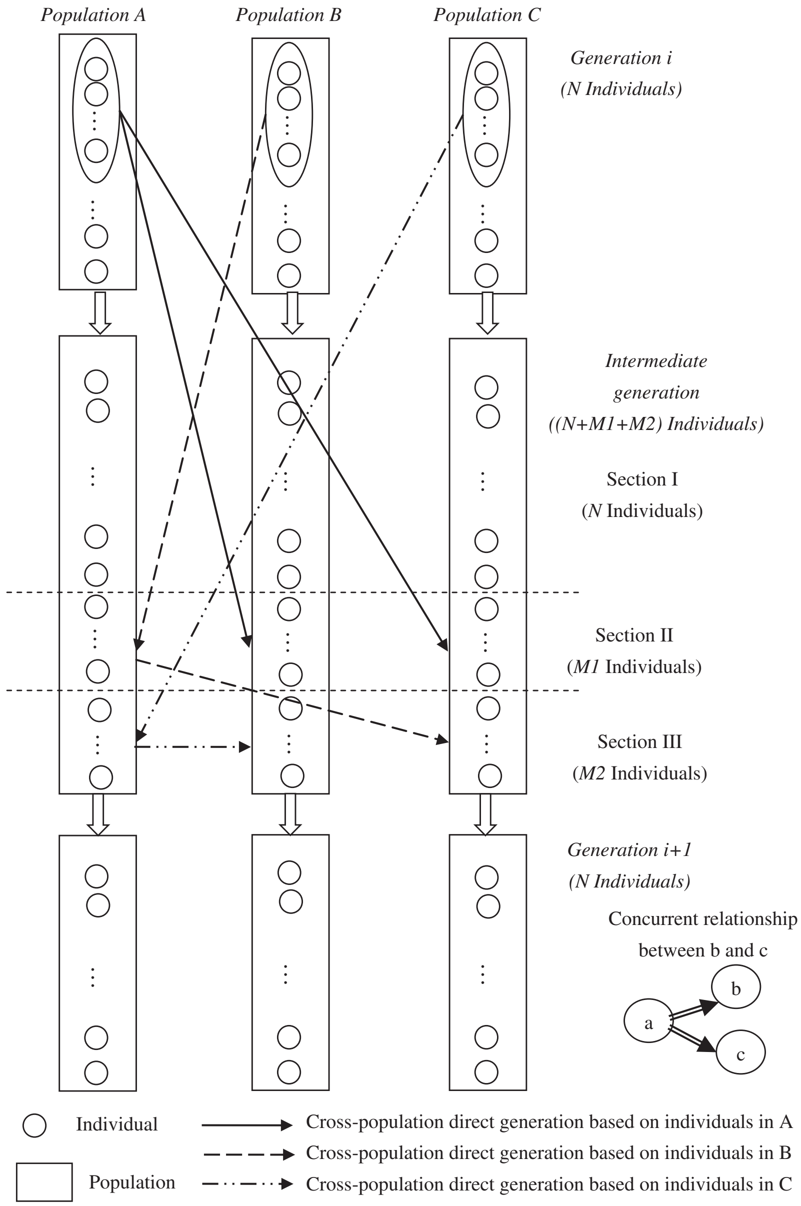


Figure 4. Evolution considering concurrent relationships between individuals in different populations.

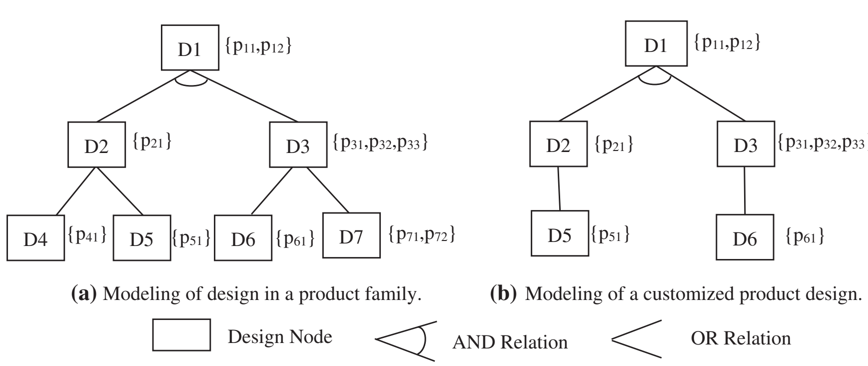


Figure 5. Modelling of design at product family level and customised product level.

shown in Figure 5(a). A customised product is modelled by a tree with only AND relations. Each node in the AND tree is composed of design parameters. In this work, a customised product design modelled by an AND tree is called a *design configuration*. Each design configuration is modelled by a set of *design parameters*.

A customised product design configuration is created from the design AND–OR tree in a product family based on the following algorithm.

Algorithm: creation of a product design configuration from a product design family

(1) Create an empty tree for modelling the customised product design. Select the root node from the product family design AND–OR tree, and use this node as the root node of the customised product design tree.
(2) From the customised product design tree, select a bottom node that has not been checked. If it has sub-nodes with an AND relation in the product family design tree, add these nodes as the sub-nodes of the selected node in the customised product design tree. If the selected node has sub-nodes with an OR relation in the product family tree, select one of these sub-nodes randomly, and add it as the sub-node of the selected node in the customised product design tree.
(3) When all the bottom nodes in the customised product design tree have been checked, this customised product design tree is then identified as a design configuration. Otherwise, go to Step (2).

3.2 *Modelling of manufacturing process and maintenance process*

Each design node in the product family AND–OR tree can be associated with some downstream product life cycle processes such as manufacturing process and maintenance process. An AND–OR graph is used to model the generic process in a product family for a design node in the product family as shown Figure 6(a). The process sub-nodes for a process super-node are also associated with either an AND or an OR relation. For example, when the process of making a hole is defined by a sequence of operations including drilling, broaching and boring, these operations are associated with an AND relation. When the process of making a hole is defined by either a drilling operation or a milling operation, these operations are associated with an OR relation. The process nodes are also linked by sequential relations. An operation is conducted, only when all its precedent operations have been completed. Each process node is associated with process parameters.

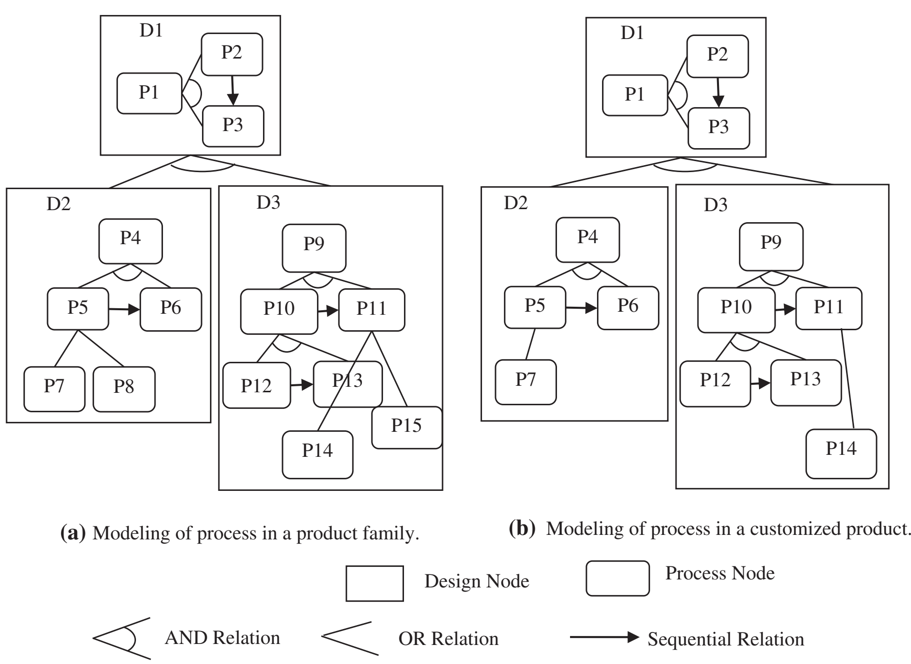


Figure 6. Modelling of process descriptions.

Creation of a feasible process AND graph for a design node in the customised product from the generic AND–OR graph of a design node in a product family is conducted using a similar method as the algorithm for creating a product design configuration from a product design family. Figure 6(b) shows three AND graphs created from the three AND–OR graphs given in Figure 6(a). The process nodes in each graph are linked with sequential relations.

The process graphs in the customised product are further transformed to build a single process graph for a customised product design configuration based on the method introduced by (Hong et al. 2010). Creation of a single process graph from a customised product design configuration is based on the following considerations.

- The bottom nodes in a process graph, called *operation nodes*, are used to represent manufacturing primitives that can be carried out by operators with certain machines and tools. Other process nodes in a process graph are defined using these operation nodes. Only operation nodes are used in the final process graph for a customised product design configuration.
- The sequential constraints defined by the sequential relations in the process graph have to be satisfied by the operations.
- For an operation that is associated with a super-node in a design configuration tree, only when all the operations in the sub-nodes of this super-node in the design tree are completed, the operation in the super-node can then be conducted.

From the individual process AND graphs for a customised product design configuration, the single graph of the process is created using the following algorithm (Hong et al. 2010). Creation of the manufacturing process graph (Figure 7) for the customised design configuration given in Figure 6(b) is used as an example to explain this algorithm.

Algorithm: creation of a process graph from a customised product design configuration

(1) For each design node, create a sequential graph of the feasible manufacturing process.
  (1.1) Pick up a bottom node from the customised process tree and remove this node from the tree.
  (1.2) If this selected node is an operation node at the bottom of the tree, add this node to the sequential graph (Figure 7(a)).
  (1.3) If this selected node is not an operation node, add the sequential relations among the sub-nodes of the selected node to the sequential graph. Also add a start-node, *S*, as the ancestor node of all the nodes without ancestor nodes in the graph, and add an end-node, *E*, as the descendant node of all the nodes without descendant nodes in the graph. When a sub-node is not an operation node, the ancestor node should be linked with the *S* node of the graph created from this sub-node, and the descendant node should be linked with the *E* node of the graph created from the sub-node (Figure 7(a)).
  (1.4) When all the nodes in the customised process tree have been removed, this sequential graph is then identified as the final process for the required design node. Otherwise, go to Step (1.1).
(2) Create the process considering all the design nodes in the customised design configuration tree.
  (2.1) For each design node, add a start-node, *S*, as the ancestor node of all the nodes without ancestor nodes in the process sequential graph, and add an end-node, *E*, as the descendant node of all the nodes without descendant nodes in the process sequential graph (Figure 7(a)).
  (2.2) For any two design nodes with a super-node/sub-node relation, select the *E* node of the graph for the sub-design node as the ancestor node of the *S* node of the graph for the super design node (Figure 7(b)).
  (2.3) Remove unnecessary *S* nodes and *E* nodes from the process graph (Figure 7(c)).

3.3 *Modelling of manufacturing resource*

Each operation node in the process AND–OR graph of a product family is associated with required resource descriptions. These resources include human operators, equipment, materials, etc. The resource requirement for a process node in a product family is defined by an AND–OR tree as shown in Figure 8(a). For a customised product, the resource requirement for each operation node is associated with an AND tree as shown in Figure 8(b). Creation of the specific resource descriptions modelled by an AND tree for a customised product from the generic resource descriptions modelled by an AND–OR tree for a product family is conducted using a similar method as the algorithm for creating a product design configuration AND tree from a product design family modelled using an AND–OR tree.

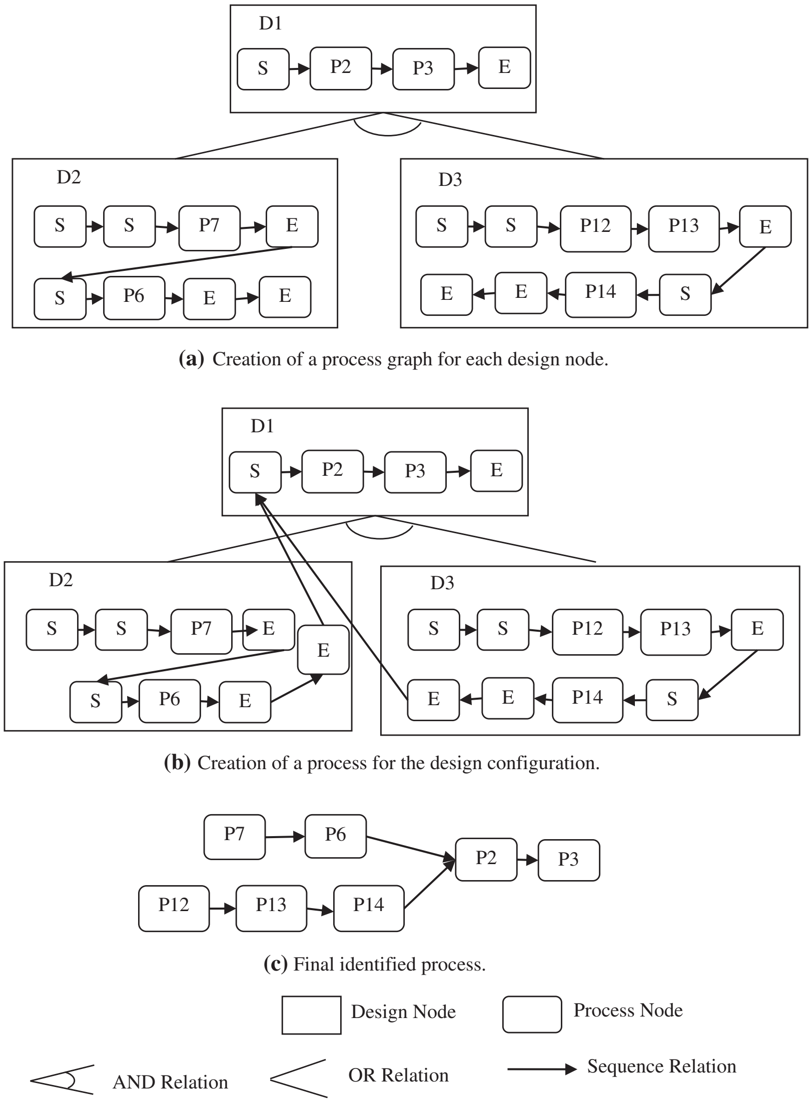


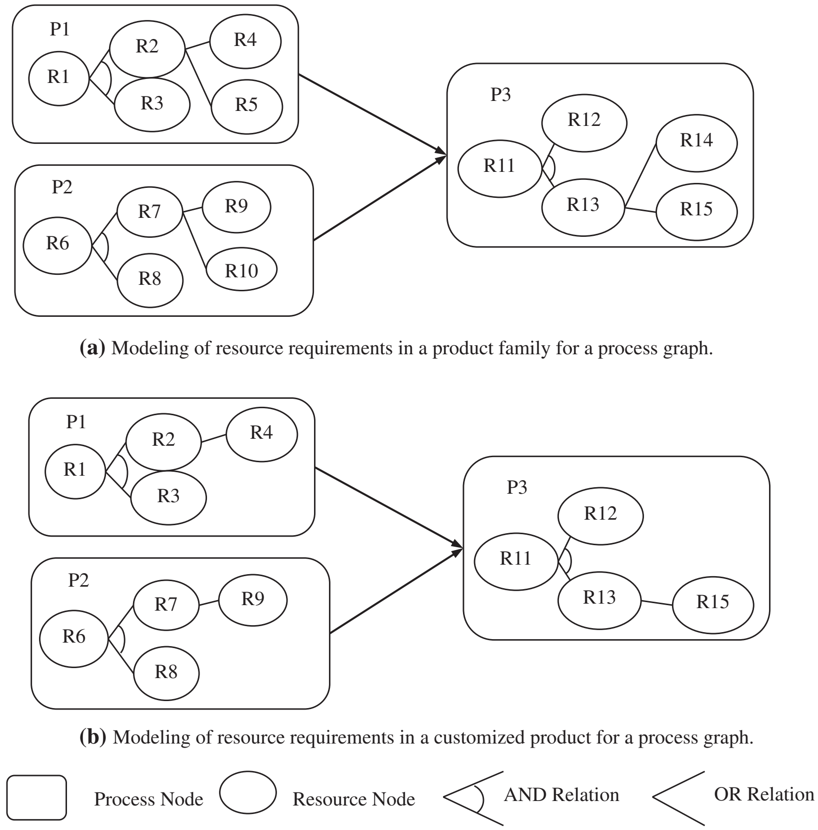
Figure 7. Creation of a customised process from a design configuration.

Since a customised process for a customised product design configuration is defined by a process graph (Figure 7(c)), the customised resource requirements can then be defined using all the created AND trees of requirements for all the operation nodes in the process graph.

4. Identification of the optimal design and its downstream product life cycle activities by multi-population CGP

Since the same requirement from the individual customer can be achieved by different design configurations, design parameters, manufacturing processes, manufacturing process parameters, maintenance processes, maintenance process parameters, manufacturing resources and manufacturing resource parameters, identification of the optimal design and its downstream product life cycle activities has to be carried out. In this research, two cases with different relations between downstream product life cycle activities are considered as shown in Table 1. In each case, three different product life cycle aspects are considered.

Each product life cycle aspect is evaluated by its life cycle evaluation measure. The whole customised product is evaluated considering all relevant life cycle aspects. Among all feasible customised products, the one with the best overall evaluation measure is selected as the optimal customised product.



(a) Modeling of resource requirements in a product family for a process graph.

(b) Modeling of resource requirements in a customized product for a process graph.

Figure 8. Modelling of resource requirements for a process graph.

Table 1. Two cases with different relations between downstream product life cycle activities.

Case	Relation type	Life cycle aspect A	Life cycle aspect B	Life cycle aspect C
Case I	Sequential relationship	Design	Manufacturing process	Manufacturing resource
Case II	Concurrent relationship	Design	Manufacturing process	Maintenance process

4.1 *Evaluation of a customised product considering different product life cycle aspects*

Suppose the i-th product life cycle aspect is evaluated by m_i evaluation measures, $F_j^{(i)}$ ($i = A, B, C; j = 1, 2, \ldots, m_i$). Since these evaluation measures are usually in different unites, these measures are first converted into comparable evaluation indices $I_j^{(i)}$ between 0 and 1 (Yang, Xue, and Tu 2006):

$$I_j^{(i)} = I_j^{(i)}\left(F_j^{(i)}(P)\right), \quad i = A, B, C; \quad j = 1, 2, \ldots, m_i \tag{1}$$

where P is the customised product. The overall evaluation index for the i-th product life cycle aspect is calculated by:

$$I^{(i)} = \sum_{j=1}^{m_i} \frac{w_j I_j^{(i)}}{w_j}, \quad i = A, B, C \tag{2}$$

where w_j is a weighting factor between 0 and 1 representing the importance of the j-th evaluation index for the i-th product life cycle aspect.

The overall evaluation index considering all three life cycle aspects is calculated by:

$$I = \frac{W^{(A)}I^{(A)} + W^{(B)}I^{(B)} + W^{(C)}I^{(C)}}{W^{(A)} + W^{(B)} + W^{(C)}} \tag{3}$$

where I is the overall evaluation index, and $W^{(A)}$, $W^{(B)}$ and $W^{(C)}$ are weighting factors for $I^{(A)}$, $I^{(B)}$ and $I^{(C)}$, respectively.

4.2 *Identification of the optimal customised product*

Identification of the optimal customised product is formulated as:

$$\max_{w.r.t.\boldsymbol{P},\boldsymbol{X}} I \tag{4}$$

where $\boldsymbol{P}$ and $\boldsymbol{X}$ are the optimal product realisation process configuration and parameters, respectively. For Case I, with sequential relationships between downstream product life cycle activities, a product realisation process configuration is described by a design configuration, its manufacturing process and resources for the manufacturing process. For Case II, with concurrent relationships between downstream product life cycle activities, a product realisation process configuration is described by a design configuration and manufacturing process and maintenance process for this design configuration. The optimisation is conducted at two different levels: configuration level and parameter level. First, the optimal parameter values for the i-th product realisation process configuration are achieved through constrained numerical optimisation:

$$\begin{gathered} \max_{w.r.t.\boldsymbol{X}_i} I(\boldsymbol{X}_i) \\ s.t. h_j(\boldsymbol{X}_i) = 0, \quad j = 1, \ 2, \ldots \\ g_j(\boldsymbol{X}_i) \leq 0, \quad j = 1, \ 2, \ldots \end{gathered} \tag{5}$$

Among all the product realisation process configurations, the one with the best overall evaluation index is identified in configuration optimisation. In this work, the parameter optimisation is conducted through numerical search (Arora 1989), while the configuration optimisation is conducted by multi-population CGP.

The overall multi-population co-evolutionary optimisation methods considering downstream product life cycle activities with sequential relationships and concurrent relationships are illustrated in Figures 3 and 4, respectively. The major difference between our co-evolutionary optimisation approach with multiple populations and the previous co-evolutionary optimisation approach with two populations is that an individual with top evaluation measure in one population can be used to create the children individuals in other populations in our method to improve the optimisation efficiency by avoiding creation of mismatched individuals in different populations.

The overall multi-population CGP method is formulated as follows.

Algorithm: multi-population CGP

(1) Generate the initial generation of individuals in populations A, B and C.

Each population is composed of N individuals. Two cases with the sequential and concurrent relationships between downstream life cycle activities are considered as shown in Table 1. In Case I, the individuals in the three populations represent design configurations, manufacturing processes and manufacturing resources, respectively. In Case II, the individuals in the three populations represent design configurations, manufacturing processes and maintenance processes, respectively. The algorithms introduced in Section 3 can be used to create these individuals.

(2) Create the intermediate generation.

(2.1) Copy N individuals in each of the three populations in the current generation to the three populations in the intermediate generation.

(2.2) Identify the matching individuals in populations A, B and C of the current generation as complete configuration solutions of customised products. For each complete configuration solution, conduct the parameter optimisation and calculate its three evaluation indices in the three aspects using Equation (1) and the overall evaluation index using Equation (2).

(2.3) From populations A, B and C of the current generation, select p, q and r percentages of individuals according to their fitness measures, respectively, and then use each of these individuals to generate the other individuals in the other two populations. The total individuals in Sections 2 and 3 (Figures 3 and 4) are limited to $M1$ and $M2$, respectively.

(3) Create the next generation.

Creation of the next generation from the intermediate generation is conducted through three GP operations: reproduction, crossover and mutation.

(4) Check whether the optimal solution has been achieved.

If the average fitness of the solutions cannot be improved in the last m generations (i.e. the improvement is less than a predefined small number ε) or the predefined maximum generation, $g_{\max}$, has been reached, the multi-population CGP needs to be stopped, and the best solution in the current generation is selected as the optimal solution. Otherwise, go to Step (2).

In this algorithm, in additional to the forward cross-population direct generation of individuals considering the relationships between the product life cycle activities such as to create a manufacturing process from a design configuration, backward cross-population direct generation such as to create the design configuration from the manufacturing process has also been considered. Compared with the forward generation process where one individual can be used to create multiple individuals, the backward generation process is much simpler where one individual can be used to create only another individual. In backward generation, when multiple nodes associated with an OR relation are considered, only the node whose information is used in the subsequent life cycle activities is selected.

In this algorithm, matching individuals from the three populations are identified to form a complete solution. For the current generation with N individuals in each of the populations A, B and C, a total of $N \times N \times N$ checks are needed. Checking whether three individuals from the three populations are matched to form a complete solution is conducted based on the following algorithm.

Algorithm: checking whether three individuals* a, b *and* c *from populations* A, B *and* C *are matched

(1) For b and c with a sequential relationship:

(1.1) Use all the process nodes in the manufacturing process individual b to select all their corresponding design nodes. If all of these design nodes can be found from the design individual a, a and b are matched.

(1.2) Use all the resource nodes in the manufacturing resource individual c to select all their corresponding manufacturing process nodes. If all of manufacturing process nodes can be found from the manufacturing process individual b, b and c are matched.

(1.3) When a and b are matched, and b and c are matched, a complete solution with a, b and c is identified. Otherwise, a complete solution is not identified.

(2) For b and c with a concurrent relationship:

(2.1) Use all the process nodes in the manufacturing process individual b to select all their corresponding design nodes. If all of these design nodes can be found from the design individual a, a and b are matched.

(2.2) Use all the process nodes in the maintenance process individual c to select all their corresponding design nodes. If all of these design nodes can be found from the design individual a, a and c are matched.

(2.3) When a and b are matched, and a and c are matched, a complete solution a, b and c is identified. Otherwise, a complete solution is not identified

Evolution from the intermediate generation to the next generation for each population is conducted using the following algorithm (Hong et al. 2010).

Algorithm: evolution from the intermediate generation to the next generation

(1) Select two parent individuals through the reproduction operation.

(1.1) Calculate the sum, S, of the weighted fitness measures, f_i, for all individuals in the population. f_i is obtained by:

$$f_i = \frac{w_s f_{\mathrm{si}} + w_p f_{\mathrm{pi,max}}}{w_s + w_p} \tag{6}$$

where f_{si} is the fitness of individual i in a single population (A, B or C), $f_{pi,max}$ is the maximum fitness measure of matching solutions that are related to individual i and w_s, and w_p are weighting factors for f_{si} and $f_{pi,max}$, respectively.

(1.2) Generate a random number, r, from the interval (0, S).

(1.3) Go through the individuals in the population and obtain the accumulate sums of fitness measures from the first individual to the i-th individual. The i-th individual is selected for reproduction when the sums satisfy the condition:

$$\sum_{j=1}^{i-1} f_j < r < \sum_{j=1}^{i} f_j \tag{7}$$

(2) Apply crossover operation to the two selected parent individuals.

(2.1) Calculate crossover rate based on the equation:

$$p_c = \begin{cases} p_{c1} - \frac{p_{c1}-p_{c2}}{f_{\max}-f_{\text{ave}}}(f_{\text{bigger}} - f_{\text{ave}}), & f_{\text{bigger}} \geq f_{\text{ave}} \\ p_{c1}, & f_{\text{bigger}} < f_{\text{ave}} \end{cases} \tag{8}$$

where $f_{\max}$ is the maximum fitness measure in the population, f_{bigger} is the one of the two selected parent individuals with bigger fitness measure, f_{ave} is the average fitness measure for all individuals in the population, and p_{c1} and p_{c2} ($p_{c1} > p_{c2}$) are two given crossover rate boundaries between 0 and 1.

(2.2) Generate a random number r between 0 and 1. If $p_c > r$, crossover is not conducted.

(2.3) When each node in the individual is associated with a positive integer, the position of crossover is identified by:

$$L_c = \text{int}[(n-1)P_c + 1] \tag{9}$$

where n is the number of nodes in the individual, and P_c is a random number between 0 and 1. The crossover position should satisfy the following conditions:

- The node at the selected location should not be a root node in the AND–OR tree.
- The two nodes at the selected two locations of the parent individuals for crossover should have an OR relation.

The crossover is conducted by swapping the two sub-trees with the selected positions as the root nodes of the sub-trees.

(3) Apply mutation operation to the selected parent individuals.

(3.1) Calculate mutation rate based on the equation:

$$p_m = \begin{cases} p_{m1} - \frac{p_{m1}-p_{m2}}{f_{\max}-f_{\text{ave}}}(f - f_{\text{ave}}), & f \geq f_{\text{ave}} \\ p_{m1}, & f < f_{\text{ave}} \end{cases} \tag{10}$$

where $f_{\max}$ is the maximum fitness measure in the population, f is the fitness measure of the selected individual for mutation, f_{ave} is the average fitness measure for all the individuals in the population, and p_{m1} and p_{m2} ($p_{m1} > p_{m2}$) are two given mutation rate boundaries between 0 and 1.

(3.2) Generate a random number r between 0 and 1. If $p_m > r$, crossover is not conducted.

(3.3) When each node in the individual is associated with a positive integer, the position of mutation is identified by:

$$L_m = \text{int}[(n-1)P_m + 1] \tag{11}$$

where n is the number of nodes in the individual, and P_m is a random number between 0 and 1. The mutation position should satisfy the following conditions:

- The node at the selected location should not be a root node in the AND–OR tree.
- The node at the selected location should have an OR relation with other nodes in the AND–OR tree.

The mutation operation is conducted by (i) removing the sub-tree with the root node at the selected location from the selected individual; (ii) from the AND–OR tree, selecting a different node that has an OR relation with the node at the selected location; (iii) generating a sub-tree with the newly selected node as its root node; and (iv) adding this sub-tree to the selected location in the selected individual.

(4) Repeat Steps (1), (2), (3) and (4) until N individuals are created in each population in the next generation.

5. Case studies

Industrial case studies were conducted for demonstrating the effectiveness of the developed approach in solving real-world problems. Comparative case studies were carried out for demonstrating the improvement in computation efficiency.

5.1 *Industrial case studies*

The case studies were to identify the designs of the customised windows and their downstream life cycle activities based on the requirements from individual customers for a windows manufacturing company (Hong et al. 2010). Figure 9 shows the modelling of life cycle descriptions for design, manufacturing process, manufacturing resource and maintenance process in a product family. Each node was modelled by parameters. In this research, the width and height of the window were selected as the parameters.

In these case studies, four design evaluation measures, one manufacturing process evaluation measure, one manufacturing resource evaluation measure and one maintenance process evaluation measure were selected to evaluate a customised window from the perspectives of design, manufacturing process, manufacturing resource and maintenance process, respectively. These seven evaluation measures are given as follows.

- Ventilation area A_{vent}, (m^2): The ventilation area is the effective area that allows air to the room.
- Viewing area A_{view}, (m^2): The viewing area is the effective area that allows for viewing the outside through the window.
- Rain risk area A_{rain}, (m^2): The rain risk area is defined as the area that rain can possibly come into the room when the window has inadvertently been left open.
- Heat loss H_{loss}, (watt/$^{\text{o}}$C): The heat loss of the window is caused by the heat loss of the frame and the glass, and it is measured in watts per degree of temperature difference between the inside and outside of the window.
- Manufacturing cost C_m, (\$): The manufacturing cost of the window is decided by the selection of the manufacturing process.
- Maintenance cost C_{mt}, (\$): The maintenance cost of the window is determined by the selection of the maintenance process.
- Lead time T, (hours): The lead time is the minimum makespan for producing the custom window. This measure can be obtained through scheduling to allocate available resources to the required manufacturing process.

Because these seven evaluation measures could not be compared directly, they were converted into seven comparable evaluation indices, as shown in Table 2, based on the method developed by Yang, Xue, and Tu (2006).

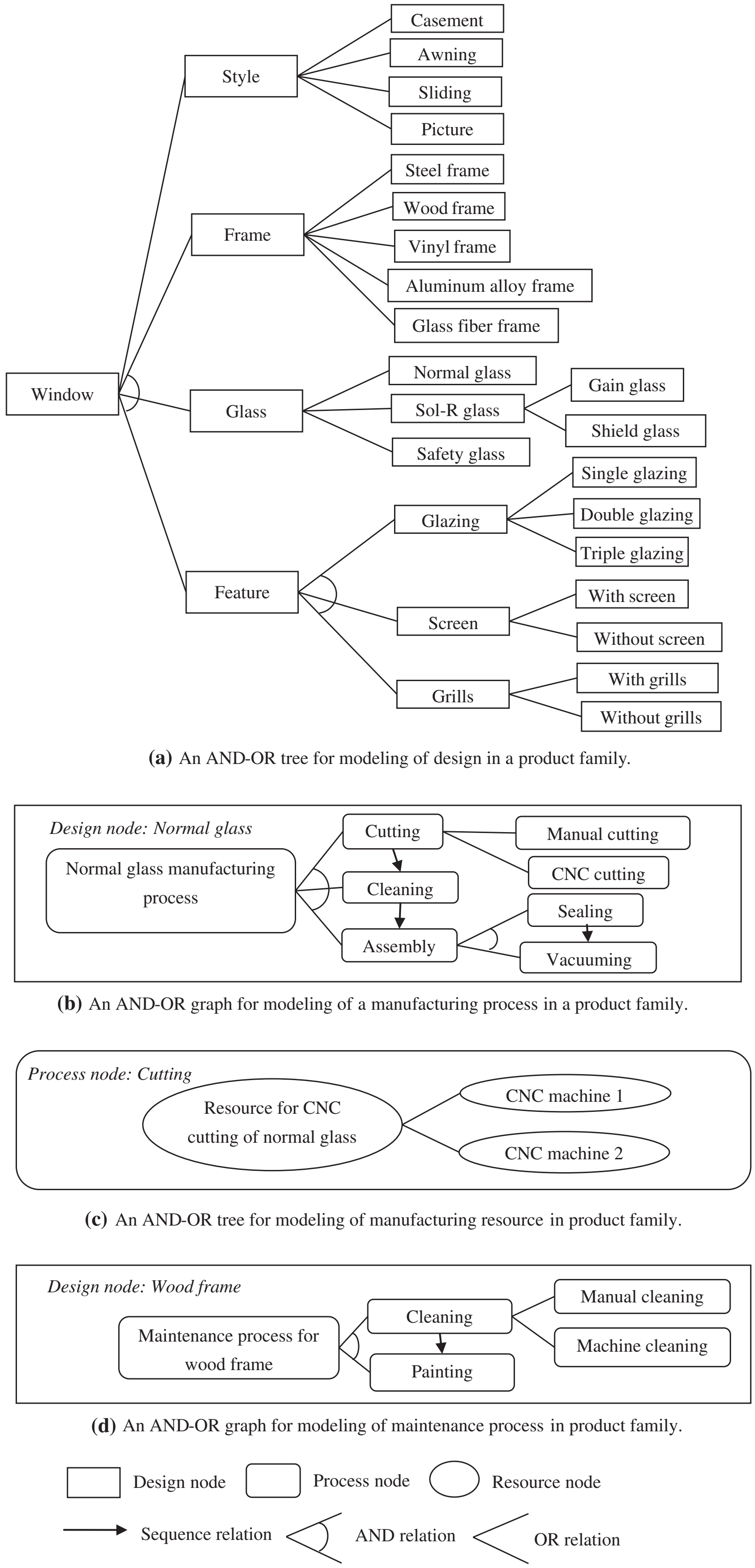
The total evaluation index considering design aspect representing customer satisfaction was defined as:

$$I_D = \frac{W_{\text{vent}} I_{\text{vent}} + W_{\text{view}} I_{\text{view}} + W_{\text{rain}} I_{\text{rain}} + W_{\text{loss}} I_{\text{loss}}}{W_{\text{vent}} + W_{\text{view}} + W_{\text{rain}} + W_{\text{loss}}} \tag{12}$$

where W_{vent}, W_{view}, W_{rain} and W_{loss} are weighting factors.

For Case I, considering design, manufacturing process and manufacturing resource with sequential relationships between downstream life cycle aspects, the optimisation objective function was defined by:

$$\text{Max } I = \frac{W_D I_D + W_m I_m + W_T I_T}{W_D + W_m + W_T} \tag{13}$$



(a) An AND-OR tree for modeling of design in a product family.

(b) An AND-OR graph for modeling of a manufacturing process in a product family.

(c) An AND-OR tree for modeling of manufacturing resource in product family.

(d) An AND-OR graph for modeling of maintenance process in product family.

Figure 9. Modelling of life cycle descriptions for window products in a product family.

Table 2. Calculation of evaluation indices from evaluation measures.

Aspect	Index calculation
Design	$I_{\text{vent}} = 6.00 \times 10^{-2}A^3_{\text{vent}} - 3.99 \times 10^{-1}A^2_{\text{vent}} + 1.06A_{\text{vent}} - 2.42 \times 10^{-1}$
	$I_{\text{view}} = 8.49 \times 10^{-3}A^3_{\text{view}} - 1.36 \times 10^{-1}A^2_{\text{view}} + 6.97 \times 10^{-1}A_{\text{view}} - 1.54 \times 10^{-1}$
	$I_{\text{rain}} = -2.43 \times 10^{-4}A^3_{\text{rain}} - 2.75 \times 10^{-1}A^2_{\text{rain}} + 1.00 \times 10^{-1}A_{\text{rain}} + 9.46 \times 10^{-1}$
	$I_{\text{loss}} = 4.72 \times 10^{-6}H^3_{\text{loss}} - 7.29 \times 10^{-4}H^2_{\text{loss}} + 7.15 \times 10^{-3}H_{\text{loss}} + 9.51 \times 10^{-1}$
Manufacturing process	$I_m = 1.31 \times 10^{-12}C^3_m - 9.04 \times 10^{-8}C^2_m + 4.22 \times 10^{-5}C_m + 9.55 \times 10^{-1}$
Maintenance process	$I_{mt} = -2.55 \times 10^{-9}C^3_{mt} + 2.89 \times 10^{-6}C^2_{mt} - 1.17 \times 10^{-3}C_{mt} + 1.04$
Manufacturing resource	$I_T = 3.84 \times 10^{-5}T^3 - 2.57 \times 10^{-3}T^2 + 2.06 \times 10^{-2}T + 9.41 \times 10^{-1}$

For Case II, considering design, manufacturing process and maintenance process with concurrent relationships between downstream life cycle aspects, the optimisation objective function was defined by:

$$\text{Max } I = \frac{W_D I_D + W_m I_m + W_{mt} I_{mt}}{W_D + W_m + W_{mt}} \tag{14}$$

where W_D, W_m, W_T and W_{mt} are weighting factors.

When the weighting factors, W_{vent}, W_{view}, W_{rain}, W_{loss}, W_D, W_m and W_T (or W_{vent}, W_{view}, W_{rain}, W_{loss}, W_D, W_m and W_{mt}), and the two design parameters, width and height of the window, are given, the optimal customised product design and its life cycle activities can be obtained using the multi-population CGP method developed in this research. To demonstrate the effectiveness of the developed method, two case studies were conducted. In the two case studies, the population size was selected as $N = 20$, p, q and r were selected as 0.1, 0.1 and 0.1, respectively, $M1$ and $M2$ were selected as 3, and p_{c1}, p_{c2}, p_{m1} and p_{m2} were selected as 0.9, 0.6, 0.5 and 0.1, respectively. The weighting factors W_{vent}, W_{view}, W_{rain}, W_{loss}, W_D, W_m and W_T were selected as 0.3, 0.8, 0.9, 0.8, 0.4, 0.3 and 0.6, respectively, for case study I, and the weighting factors W_{vent}, W_{view}, W_{rain}, W_{loss}, W_D, W_m and W_{mt} were selected as 0.3, 0.8, 0.9, 0.8, 0.7, 0.5 and 0.5, respectively, for case study II. For the two case studies, both the width and height of the window were given as 1.5 m. For case study I, the optimal solution was obtained after 20 generations were created, and the best overall evaluation index was identified as $I^* = 0.9183$. Table 3 shows the optimisation result described by the customised product design configuration, manufacturing process and manufacturing resource. The average fitness measures (i.e. the overall evaluation indices) over the 20 generations for case study I are shown in Figure 10.

For case study II, the optimal solution was identified after 20 generations were created, and the best overall evaluation index was achieved as 0.9062. The detail descriptions of the optimal solution are given in Table 4. Figure 11 shows the average fitness measures (i.e. the overall evaluation indices) over the 20 generations for case study II.

Figures 10 and 11 show changes of the average fitness measures in the optimisation processes over 20 iterations for the two case studies. Both the overall fitness measures considering all the three populations and the individual fitness

Table 3. Optimisation result for case study I.

Product life cycle activity	Optimal solution	Evaluation index
Product configuration	Window, Style, Awning, Frame, Vinyl frame, Glass, Sol-R glass, Gain glass, Feature, Glazing, Double glazing, Screen, Without screen, Grills, Without grills	0.8935
Manufacturing process planning	Instal glass, Instal beads/*for Window*/, Manual cutting, Assembly/*for Vinyl frame*/, Manual cutting, Cleaning, Sealing, Vacuuming/*for Sol-R gain glass*/, ...	0.9452
Manufacturing resource planning	Instal tool 1/*for Instal glass*/, Instal tool 3/*for Instal beads*/, Manual cutter 5/*for Manual cutting of Vinyl frame*/, Assembly tool 2/*for Assembly of Vinyl frame*/, Manual cutter 10/ *for Manual cutting of Sol-R gain glass*/, …	0.9212
Overall product		0.9183

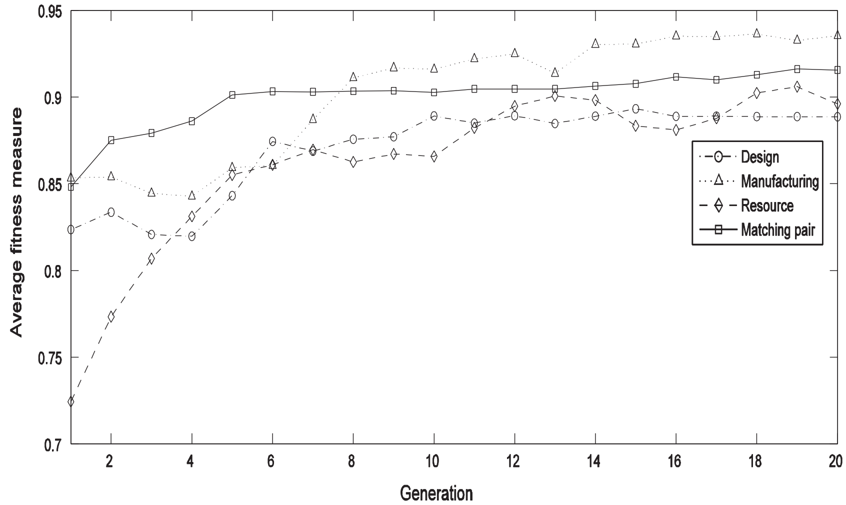


Figure 10. Average fitness measures over 20 generations for case study I.

Table 4. Optimal solution for case study II.

Product life cycle activity	Optimal solution	Evaluation index
Product configuration	Window, Style, Awning, Frame, Vinyl frame, Glass, Safety glass, Feature, Glazing, Double glazing, Screen, Without screen, Grills, Without grills	0.8936
Manufacturing process planning	Instal glass, Instal beads/*for Window*/, Manual cutting, Assembly/*for Vinyl frame*/, Manual cutting, Cleaning, Sealing, Vacuuming/*for Safety glass*/, ...	0.9480
Maintenance process planning	Inspection, Disassembly/assembly/*for Window*/, Cleaning, Repair/*for Vinyl frame*/, … ...	0.8820
Overall product		0.9062

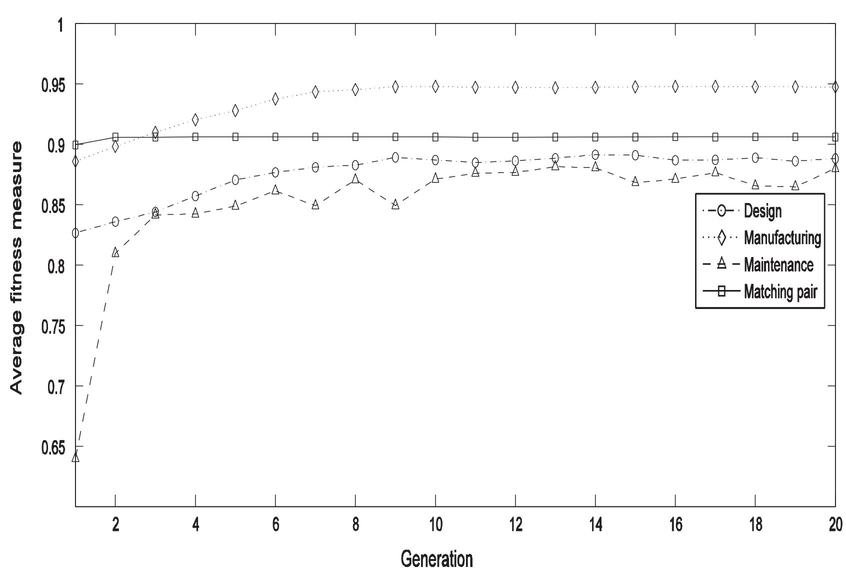


Figure 11. Average fitness measures over 20 generations for case study II.

measures considering each of the three individual populations are included in these curves. For case study I, shown in Figure 10, the four fitness measures are the design performance index, the manufacturing process plan index, the manufacturing resource plan index and the overall index. For case study II, shown in Figure 11, the four fitness measures are the design performance index, the manufacturing process plan index, the maintenance process plan index and the overall index. Apparently, the optimal product was obtained through a trade-off between product design and its downstream product life cycle activities based on the principle of concurrent engineering design.

5.2. *Comparative case studies*

Two comparative case studies have been conducted to demonstrate the effectiveness of the developed approach to improve the computational efficiency. Same as the industrial case studies, the problem with the sequential relationship between downstream life cycle activities is considered as case study I, and the problem with the concurrent relationship between downstream life cycle activities is considered as case study II.

Table 5 provides the main parameters selected for comparative case studies. In the two comparative case studies, the computation quality and efficiency among the traditional GP, the CGP developed by Hong et al. (2010), and our newly developed multi-population co-evolutionary genetic programming (MCGP) method developed in this research were compared. The crossover rates and mutation rates were tuned such that the optimal solutions could be achieved with reasonable efficiency. To demonstrate the advantages of the developed MCGP, smaller population size and smaller maximum generation were selected.

In the traditional genetic programming (GP), an individual in the population was modelled by a hybrid tree with three different types of product life cycle activity descriptions. In the comparative case study I, a hybrid tree was used to describe the design, its manufacturing process and its manufacturing resource plan. In the comparative case study II, a hybrid tree was used to describe the design, its manufacturing process plan and its maintenance process plan. The crossover operation and mutation operation in the traditional genetic programming were conducted by randomly selecting a node that had an OR relation with other nodes in the hybrid tree based on the predefined crossover rate and mutation rate.

In the CGP developed by Hong et al. (2010), only two different populations were created to describe two different types of product life-cycle activities (i.e. design and manufacturing). To demonstrate the importance of the cross-population direct generation mechanism for MCGP, the original CGP method was modified in these case studies to model three different types of product life cycle activities with three populations without the cross-population direct generation mechanism.

In the MCGP, three populations were created to describe three different types of product life cycle activities. For comparative case study I, the three populations were used to model designs, manufacturing process plans and manufacturing resource plans. For comparative case study II, the three populations were used to model designs, manufacturing process plans and maintenance process plans.

Due to the stochastic nature of genetic programming, different results were achieved with the same input conditions. To solve this problem, each algorithm was run 10 times, and the average measures were used as the results for the comparative studies. Tables 6 and 7 show the results for the two comparative case studies.

From Tables 6 and 7, we can see that the MCGP method developed in this research provides the best computation quality and efficiency among the selected three methods. In the 10 runs using the three algorithms, the MCGP achieved 100% success to obtain the optimal fitness in 20 generations. In addition, the MCGP also took the shortest time among all the three algorithms. For the other two algorithms, they failed sometimes to achieve the optimal result in 200 generations. These two algorithms also took longer time to get the solutions. Especially the CGP was the worst for

Table 5. Main parameters for the two comparative case studies.

Parameter	GP	CGP	MCGP
Population size	40	40	20
Maximum generation	200	200	20
Crossover rate	0.9	$p_{c1} = 0.9$, $p_{c2} = 0.6$	$p_{c1} = 0.9$, $p_{c2} = 0.6$
Mutation rate	0.5	$P_{m1} = 0.5$, $p_{m2} = 0.1$	$P_{m1} = 0.5$, $p_{m2} = 0.1$

Table 6. Results for comparative case study I.

Run time	GP		CGP		MCGP	
	Fitness	Time (s)	Fitness	Time (s)	Fitness	Time (s)
1	0.9494	731.4	0.9441	1472.3	0.9494	304.0
2	0.8911	732.1	0.8632	1199.2	0.9494	387.2
3	0.9348	747.7	0.8844	1227.7	0.9494	379.2
4	0.9494	728.1	0.8618	1214.4	0.9494	323.1
5	0.9494	733.8	0.9494	1336.2	0.9494	315.9
6	0.9015	732.3	0.9494	1517.2	0.9494	282.1
7	0.9441	725.1	0.8827	1213.5	0.9494	355.3
8	0.8911	721.7	0.9437	1969.5	0.9494	303.4
9	0.8765	752.5	0.8911	1417.3	0.9494	274.3
10	0.9494	741.8	0.9348	1379.7	0.9494	299.5
Average	0.9237	734.7	0.9105	1394.7	0.9494	322.4
Best	0.9494	721.7	0.9494	1199.2	0.9494	274.3
Worst	0.8765	752.5	0.8618	1969.5	0.9494	387.2

Table 7. Results for comparative case study II.

Run time	GP		CGP		MCGP	
	Fitness	Time (s)	Fitness	Time (s)	Fitness	Time (s)
1	0.9221	417.0	0.9242	923.7	0.9319	107.1
2	0.9242	421.3	0.9221	825.4	0.9319	146.8
3	0.9242	426.6	0.9319	1081.6	0.9319	118.8
4	0.9242	455.5	0.9242	957.6	0.9319	122.5
5	0.9319	484.7	0.9221	956.0	0.9319	123.3
6	0.9221	536.6	0.9242	1050.6	0.9319	126.7
7	0.9221	443.3	0.9242	800.3	0.9319	130.6
8	0.9242	441.0	0.9220	982.2	0.9319	134.5
9	0.9221	441.2	0.9319	1120.3	0.9319	110.1
10	0.922	437.3	0.9242	1178.8	0.9319	124.0
Average	0.9239	450.5	0.9251	987.65	0.9319	124.4
Best	0.9319	417.0	0.9319	800.3	0.9319	107.1
Worst	0.9220	536.6	0.9220	1178.8	0.9319	146.8

vvcomparative case study I, since only a few complete solutions could be matched considering three populations with sequential relationships without using the cross-population direct generation mechanism.

From the case study results, we can find that although smaller population size and smaller generation size were used in the MCGP (i.e. 20 in MCGP vs. 40 in GP and CGP for population size, and 20 in MCGP vs. 200 in GP and CGP for generation size), the MCGP could still obtain the best performance among all the three algorithms. We can conclude that the developed MCGP is effective to identify the optimal product design configuration and its downstream product life cycle activities based on requirements from individual customers.

6. Conclusions and future work

In this research, a MCGP approach has been introduced to identify the optimal customised product design and its downstream life cycle activities for mass customisation production. Advantages of the developed new method are summarised as follows.

(1) The developed modelling scheme is effective to model various product life cycle descriptions at both product family level and customised product level using AND–OR trees and AND–OR graphs.
(2) The MCGP method is effective to identify the optimal customised product design and its downstream product

life cycle activities with both sequential and concurrent relationships.

(3) The cross-population direct generation mechanism for the MCGP method is effective to increase the number of feasible solutions considering all three product life cycle activities based on the evaluation measures in the individual populations, thus further improving the computation efficiency

The following issues need to be addressed in our future work.

(1) The developed approach is only effective to model product family with simple data structure. Knowledge-based methods such as ontology will be investigated to model the complex relationships among various product life cycle activities.
(2) In this work, only sequential and concurrent relationships among several product life cycle activities are considered. Design of a customised product whose activities are modelled by a network of sequential and concurrent relations will be considered in our future work.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was support by the Natural Sciences and Engineering Research Council (NSERC) of Canada [grant number 185788]; the National Natural Science Foundation of China [grant number 61503362], [grant number 61305111], [grant number 91420104]; and the Natural Science Foundation of Anhui Province [grant number 1508085MF133] is acknowledged.

ORCID

Biao Yu http://orcid.org/0000-0002-2015-5777
Han Zhao http://orcid.org/0000-0003-1796-8395
Deyi Xue http://orcid.org/0000-0001-5285-8867

References

Arora, J. S. 1989. *Introduction to Optimum Design*. New York: McGraw-Hill.

Ferguson, S. M., A. T. Olewnik, and P. Cormier. 2014. "A Review of Mass Customization across Marketing, Engineering and Distribution Domains toward Development of a Process Framework." *Research in Engineering Design* 25 (1): 11–30.

Fogliatto, F. S., G. J. C. da Silveira, and D. Borenstein. 2012. "The Mass Customization Decade: An Updated Review of the Literature." *International Journal of Production Economics* 138 (1): 14–25.

Gershenson, J. K., G. J. Prasad, and Y. Zhang. 2004. "Product Modularity: Measures and Design Methods." *Journal of Engineering Design* 15 (1): 33–51.

Goswami, M., and M. K. Tiwari. 2015. "Product Feature and Functionality Driven Integrated Framework for Product Commercialization in Presence of Qualitative Consumer Reviews." *International Journal of Production Research* 53 (16): 4769–4788.

Hegge, H. M. H., and J. C. Wortmann. 1991. "Generic Bill-of-material: A New Product Model." *International Journal of Production Economics.* 23 (1–3): 117–128.

Hong, G., L. Hu, D. Xue, Y. L. Tu, and Y. L. Xiong. 2008. "Identification of the Optimal Product Configuration and Parameters Based on Individual Customer Requirements on Performance and Costs in One-of-a-kind Production." *International Journal of Production Research* 46 (12): 3297–3326.

Hong, G., P. Dean, W. Yang, Y. L. Tu, and D. Xue. 2010. "Optimal Concurrent Product Design and Process Planning Based on Requirements of Individual Customers in One-of-a-kind Production." *International Journal of Production Research* 48 (21): 6341–6366.

Hong, G., D. Xue, and Y. L. Tu. 2010. "Rapid Identification of the Optimal Product Configuration and Its Parameters Based on Customer-centric Product Modeling for One-of-a-kind Production." *Computers in Industry* 61 (3): 270–279.

Jiao, J., M. M. Tseng, Q. Ma, and Y. Zou. 2000. "Generic Bill-of-materials-and-operations for High-variety Production Management." *Concurrent Engineering* 8 (4): 297–321.

Jiao, J., T. W. Simpson, and Z. Siddique. 2007. "Product Family Design and Platform-based Product Development: A State-of-the-art Review." *Journal of Intelligent Manufacturing* 18 (1): 5–29.

Jose, A., and M. Tollenaere. 2005. "Modular and Platform Methods for Product Family Design: Literature Analysis." *Journal of Intelligent Manufacturing* 16 (3): 371–390.

Juengst, W. E., and M. Heinrich. 1998. "Using Resource Balancing to Configure Modular Systems." *IEEE Intelligent Systems* 13 (4): 50–58.

Koza, J. R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: The MIT Press.

Kumar, S., and A. K. Chatterjee. 2015. "A Profit Maximizing Product Line Optimisation Model under Monopolistic Competition." *International Journal of Production Research* 53 (5): 1584–1595.

Liu, Y., S. C. J. Lim, and W. B. Lee. 2013. "Product Family Design through Ontology-based Faceted Component Analysis, Selection, and Optimization." *Journal of Mechanical Design* 135 (8): 081007.

McDermott, J. 1982. "R1: A Rule-based Configurer of Computer Systems." *Artificial Intelligence* 19 (1): 39–88.

McGuinness, D., and J. R. Wright. 1998. "Conceptual Modelling for Configuration: A Description Logic-based Approach." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 12 (4): 333–344.

Pitiot, P., M. Aldanondo, E. Vareilles, P. Gaborit, M. Djefel, and S. Carbonnel. 2013. "Concurrent Product Configuration and Process Planning, towards an Approach Combining Interactivity and Optimality." *International Journal of Production Research* 51 (2): 524–541.

Schneeweiss, D., and P. Hofstedt. 2011. "FdConfig: A Constraint-based Interactive Product Configurator." In *Proceeding of the 19th International Conference on Applications of Declarative Programming and Knowledge Management (INAP 2011)*, Vienna, Austria.

Tseng, M. M., and F. T. Piller. 2003. *The Customer Centric Enterprise: Advances in Mass Customization and Personalization*. New York: Springer.

Tseng, H. E., C. C. Chang, and S. H. Chang. 2005. "Applying Case-based Reasoning for Product Configuration in Mass Customization Environments." *Expert Systems with Applications* 29 (4): 913–925.

Yang, H., D. Xue, and Y. L. Tu. 2006. "Modeling of the Non-linear Relations among Different Design and Manufacturing Evaluation Measures for Multi-objective Optimal Concurrent Design." *Concurrent Engineering* 14 (1): 43–53.

Yang, D., M. Dong, and R. Miao. 2008. "Development of a Product Configuration System with an Ontology-based Approach." *Computer-Aided Design* 40 (8): 863–878.

Zhang, X., G. Q. Huang, and M. J. Rungtusanatham. 2008. "Simultaneous Configuration of Platform Products and Manufacturing Supply Chains." *International Journal of Production Research* 46 (21): 6137–6162.

Zhou, C. J., Z. H. Lin, and C. T. Liu. 2008. "Customer-driven Product Configuration Optimization for Assemble-to-order Manufacturing Enterprises." *The International Journal of Advanced Manufacturing Technology* 38 (1–2): 185–194.