Design for Cost: Module-Based Mass Customization

Catherine Da Cunha, Bruno Agard, Andrew Kusiak

▶ To cite this version:

HAL Id: hal-00229758

https://hal.archives-ouvertes.fr/hal-00229758

Submitted on 31 Jan 2008

Design for Cost: Module-Based Mass Customization

Catherine da Cunha, Bruno Agard, and Andrew Kusiak, *Member, IEEE*

Abstract—The assemble-to-order (ATO) production strategy considers a tradeoff between the size of a product portfolio and the assembly lead time. The concept of modular design is often used in support of the ATO strategy. Modular design impacts the assembly of products and the supply chain, in particular, the storage, transport, and production are affected by the selected modular structure. The demand for products in a product family impacts the cost of the supply chain. Based on the demand patterns, a mix of modules and their stock are determined by solving an integer programming model. This model cannot be optimally solved due to its high computational complexity and, therefore, two heuristic algorithms are proposed. A simulated annealing algorithm improves on the previously generated solutions. The computational results reported in this paper show that significant savings could be realized by optimizing the composition of modules. The best performance is obtained by a simulated annealing combined with a heuristic approach.

Note to Practitioners—In today's markets, companies tend to compete with product costs and delivery times. Mass customization principles have emerged as ways of gaining competitiveness. To individualize offerings of products, some companies manage options and alternatives around the same product category. Large product portfolios allow customers to select products closely matching their needs, however, it usually implies a larger cost to the producer. The importance of demand for final products when allocating assembly operations in a supply chain is emphasized. The research reported in this paper demonstrates that for the same level of anticipated product demand, the level and mix of stock impact the customer delivery time. The latter implies that the same degree of customer satisfaction could be realized with different anticipated stock levels and costs. An assemble-to-order (ATO) case study is presented. The ATO production strategy considers a tradeoff between the size of a product portfolio and the assembly lead time. The concept of modular design is often used in support of the ATO strategy. Modular design impacts the assembly of products and the supply chain, in particular, the storage, transport, and production are affected by the selected modular structure.

Index Terms—Assemble to order (ATO), demand integration, design for cost, mass customization, supply chain, time constraint.

I. INTRODUCTION

TODAY's markets call for elaborate competition schemes as a large variety of products is available to meet customer requirements. Companies tend to compete on product costs and delivery times. Mass production principles have emerged as ways of gaining competitiveness. To individualize offerings of products, some companies manage options and alternatives around the same product category. Enlarged product portfolios make it possible for a customer to select and purchase a product closely matching his or her requirements.

If the manufacturing lead time of a product is longer than the acceptable customer wait time, a company needs to anticipate the future by producing products before the orders are received. While any unmet product demand is not well received by the customers, overproduction lowers profits.

The research reported in this paper demonstrates that for the same level of anticipated product demand, the level and mix of stock impact the customer delivery time. The latter implies that the same degree of customer satisfaction could be realized with different anticipated stock levels and costs.

In this paper, an assemble-to-order (ATO) case study is presented. A simulated annealing and two heuristic algorithms are developed to determine the best stock composition to ensure satisfactory product delivery time.

The importance of information about customer requirements is highlighted. Some previous studies have used data about physical characteristics of products. In this paper, the demand information is used to minimize the cost induced by the diversity of products.

The literature of interest to the problem discussed in the paper is reviewed in Section II. The industrial background supporting the problem discussed in the paper is presented in Section III. The model developed in the paper is presented in Section IV. Section V discusses heuristic algorithms for solving the problem of Section IV. Computational results are also discussed. Section VI concludes the paper.

II. LITERATURE REVIEW

The design of a product portfolio impacts the cost and delivery time of products. The main questions are: What product portfolio to offer?; How to produce the products?; and How to integrate the demand data to optimize the overall process?

A way of addressing these questions is by the design of product families. Numerous approaches have been developed for the design of product families. Some are based on negotiation [11], other approaches distinguish stable and variable aspects of the customer requirements [27], and some are based on coupling between functional/technical domains [8], [13]. Martin [21] proposed a robust product family architecture to

support the redesign of products. Agard and Tollenaere [3] discussed a methodology for the design of a product family based on the analysis of customer needs with data mining tools [1].

Two main approaches that support large product diversity are modular design and postponement (delayed product differentiation). Modularity involves creating largely independent blocks (called modules) for building different products and/or processes. A combination of a limited number of modules permits diversification of the product.

A limited number of modules may result in a large number of final products. Modular design impacts the stock mix of modules (the number of modules stocked and the constitution of each module), the delivery time (through the final assembly time), and the production cost. The design of modular products has received some attention in the literature (e.g., [12], [17], and [18]).

Implementation of the modularity concept can be accomplished by partitioning a product into semi-independent or mutually separable elements. It permits for independent design, manufacture, and service of the modules. Fujita [9] optimized the content of modules and their mix in a fixed modular architecture. Yigit *et al.* [26] solved a similar problem by determining the best subsets of modules that minimize the reconfiguration cost. Briant [4] defined modules by covering a set of products with standardized components.

A large market diversity of products supported by a low technical diversity reduces process diversification, thus guaranteeing acceptable product development and manufacturing costs [7], [20].

Postponement relies on delaying as much as possible the moment when the product attains its uniqueness (identity). For example, packaging postponement [25] involves delayed differentiation of the product until the packaging operation takes place.

Lee and Tang [19] highlighted advantages of the product delayed differentiation to manufacture diversified products. Product and process standardization are highlighted as well as process restructuring.

Some research results have been reported in the design-to-cost and in the ATO context [3], [12]. Swaminathan and Tayur [24] optimized production capacity using preassembly operations.

ATO is "a production environment where goods (or services) are assembled after the receipt of a customer's order".[1] It enables managing a wide diversity of products with a limited number of modules [22]. These modules are preassembled in advance, shipped to the assembly location, and stocked. The final assembly of the product (from the modules) is initiated when an actual order is received.

The product demand pattern (demand volume of a specific product) impacts a supply chain (i.e., the storage, transportation, and production costs). Knowing the information about customer demands and assigning the workload to different actors of the chain could reduce this cost.

Da Cunha [7] used product demand data to design modules minimizing the final assembly cost (by minimizing the mean assembly time). The computational results showed that significant savings could be realized when taking such data into account .

---

[1]*American Production and Inventory Control Society Dictionary*, 9th Edition, Falls Church, VA, 1998.
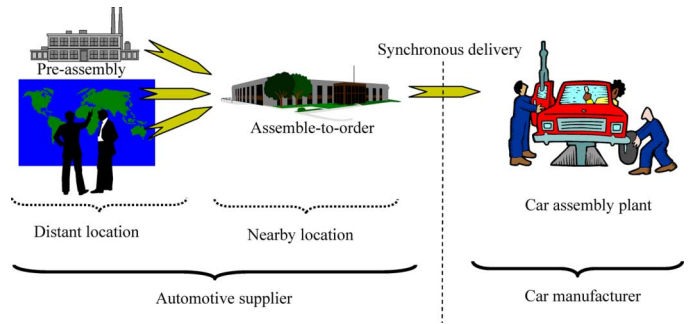

Fig. 1. Supply chain structure in the automotive industry.

This literature review stresses three questions: 1) What product portfolio to offer?; 2) How to produce the products?; and 3) How to integrate the demand data to optimize the overall process? These are often treated separately, each question raising complex issues. A simultaneous consideration of the three views, which would be the ideal, still seems utopian. Here, two views are considered: the question of how to produce is addressed while considering the information about the demand. A design-for-cost method is chosen in order to integer notions of the optimization of the overall process.

In this paper, the product-mix data are used to design a supply chain in the presence of different labor costs. The proposed approach guarantees that the mean delivery time does not exceed the delay acceptable by the customer. A cost metric is used to evaluate performance of the designed supply chain.

III. INDUSTRIAL CASE STUDY

In this section, an industrial case study is discussed. Consider the example of a midrange car—Peugeot 307. One of many assembles used in this car to distribute power and information within the vehicle is a wire harness. Potentially, there are seven million different wire harnesses that could be ordered [2]. The annual sale volume for this car model is about 350,000. Kocher and Rolland [15] stated that 90% of the sales of this car model constitute 20% of the portfolio.

The entire automotive supply chain faces the challenge of offering growing product diversity at competitive prices, quality, and delivery times. The classical supply chain structure in the automotive industry is illustrated in Fig. 1, which focuses on the automotive supplier's structure.

Car assembly plants tend to be located near the consumer market. The components and subassemblies of a car are produced by automotive suppliers that are contractually committed to deliver the right parts in the sequence and time required by the car manufacturer.

One of those components is a wire harness that has to be assembled before the car interior (floor covering, seats, inner door shells, etc.) is installed. A delay of its delivery causes a major disruption to the production line. Therefore, a contract between the car manufacturer and a wire harness supplier specifies penalties for the delays. The wire harnesses have to meet a mean lead-delivery time that is fixed in advance. As some deviations around the mean lead time may occur, the car manufacturer imposes a delivery ratio on the contractual mean lead time.
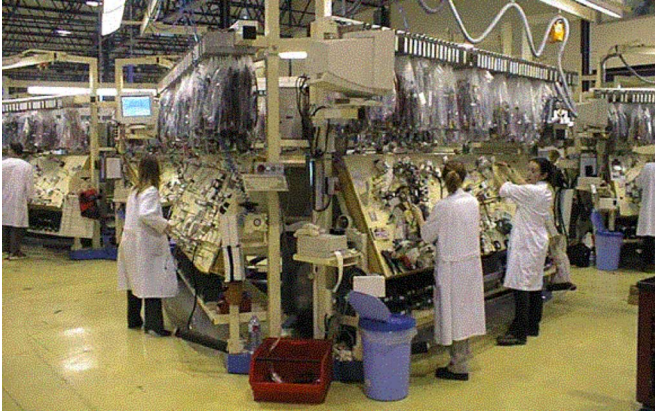
Fig. 2. Final assembly of wire harnesses [7].

Fig. 3. Operations in the supply chain.

To meet the car manufacturer's orders, wire-harness suppliers tend to organize their processes at multiple sites. Distant location plants produce modules to stock and ship them for stocking at a location near the final assembly until an actual order arrives. The final assembly of the wire harness is performed in a relatively short time to meet the arriving orders. This assembly operation (see Fig. 2) does not require highly skilled labor. A single wire is assembled on an assembly table tended by one worker. This type of work environment prevents simultaneous tasks on the same subassembly. The time of the final assembly is therefore mostly dependent of the number of modules to be assembled. The parts are uniformly easy to manipulate and the operations of connections between parts (taping, clipping, welding) are tasks of similar duration.

Each communication function of a wire harness is performed by a set of cables that occupy a designated space in the final product (i.e., the car). Each option can therefore only appear, at most, once. The information about possible incompatibility (between options or alternatives) is included in the demand itself. The demand probability of a product that does not satisfy these constraints is zero. This characteristic enables keeping the model as general as possible.

The model discussed in this paper does not consider production capacity. The capacity assumption has been validated on an industrial case study. Considering the distant locations of assembly facilities, capacity allocation and management are controlled by a supplier (in particular, by means of workforce management). The supplier is responsible for the primal assembly until the modules arrive to the car assembly plant.

IV. PROBLEM MODELING

The operations involved in the supply chain are shown in Fig. 3.

The preassembly of modules from basic components is performed at distant locations (with low labor costs). Then, the modules are transported to the nearby location and stocked until demand occurs. The demand launches the final assembly process for the wire harnesses (i.e., the products are assembled from the modules). The lead time measured at the car assembly plant involves the final assembly and delivery time.

The goal of this paper is to determine the lowest cost composition of modules (a set of basic components) so that the mean
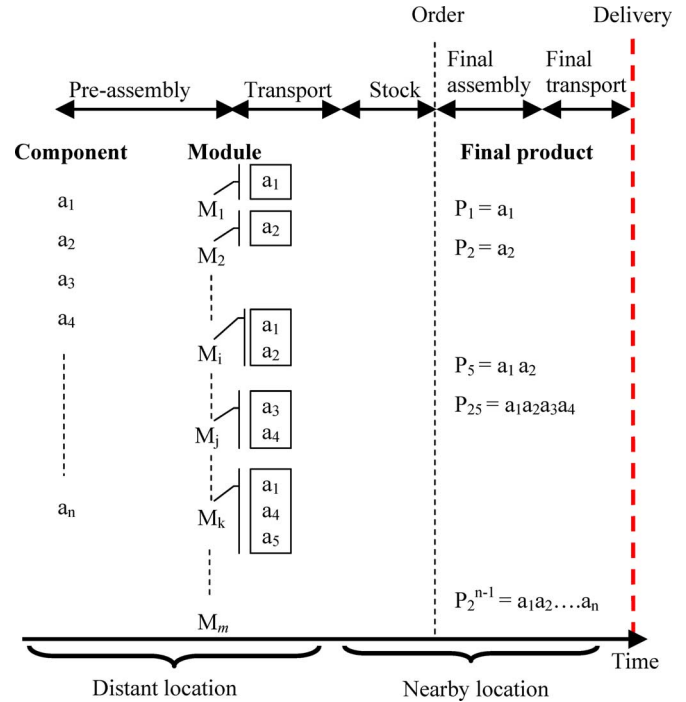
lead time does not exceed x, where x represents the delay that is acceptable to the customers. It is assumed that the transportation cost of a product from the nearby location to the car assembly plant is fixed.

The proposed model ensures that each customer receives the exact product she or he has ordered. This implies that the extra components (free upgrades of the product) are not allowed.

A. Modeling

The model presented next used the data of the wire-harness case study. It specifies that the final portfolio of products is obtained from n basic components that can be assembled as $P = 2^n - 1$ different products. It assumes that there are no assembly constraints between the components. This assumption considers the worst-case scenario (i.e., the largest possible diversity). Of course, in the presence of constraints, the number of products P is smaller. That is to be compared with the annual production on 350,000 cars for the wire harnesses considered.

Let C be the stock mix (i.e., a set of modules in the inventory). The stock mix is represented as a binary vector of size P, the ith dimension, $C_i, (i \in [1, P])$ stands for the presence ($C_i = 1$) or absence ($C_i = 0$) of the module M_i in the module inventory at the nearby location site. The inventory levels of modules may differ. Call m the number of modules in the inventory, then $m = \sum_{i=1}^{P} C_i$ and $m \leq P$.

The preassembly performed at a distant location produces m modules M_i.

Note that $|M_i|$ is the number of components in module i. For example, in Fig. 3, $M_i = a_1 a_2$, $|M_i| = 2$ and $M_k = a_1 a_4 a_5$, and $|M_k| = 3$.

Note that $AT(C)$ is the mean assembly time of the final product from the set of modules C.

The model is formulated as follows:

$$\underset{C}{\mathrm{Min}}\ \mathrm{Cost} \tag{1}$$

$$\mathrm{s.t.}\ \mathrm{AT}(C) < x \tag{2}$$

where

Cost = Preassembly cost + Fixed transportation cost

+ Variable transportation cost + Final assembly cost. (3)

The solution to problem (1), (2) determines the stock composition represented by vector C. Constraint (2) ensures that the mean assembly time is acceptable to the car assembly plant.

To compute the cost C in (3), the following assumptions are made.

- A wire harnesses is principally composed of a set of cables and connectors.
- The cost of each elementary assembly operation is identical due to the similar level of assembly difficulty.
- The preassembly cost is then proportional to the number of components being assembled in each module M_i present in the stock

$$\mathrm{Preassembly\ cost} \propto \sum_{i=1}^{P} C_i \left(|M_i| - 1 \right). \tag{4}$$

For example, the module composed of three components a_1, a_2, and a_3 involves two assembly operations. Therefore, the cost is double that of assembling a_1 and a_3 involving one operation only.

The transportation cost from the distant to the nearby location site includes fixed and variable transportation cost.

- The fixed transportation cost is assumed to be proportional to the number of different items managed (i.e., the number of different modules (m))

$$\mathrm{Fixed\ transportation\ cost} \propto m. \tag{5}$$

- Each module is delivered in fixed intervals and quantities. The variable transportation cost is then proportional to the volume of a module (for simplicity, it is considered to be proportional to the number of components in a module)

$$\mathrm{Variable\ transportation\ cost} \propto \sum_{i=1}^{P} C_i |M_i| \tag{6}$$

The transportation to the car assembly plant is not included in the proposed model. Indeed, this cost depends on the demand but it is independent of the modular design.

The final assembly cost is assumed to be proportional to the mean final assembly time

$$\mathrm{Final\ assembly\ cost} \propto AT(C) \tag{7}$$

The final assembly time (AT) is assumed to be proportional to the number of final assembly operations. This assumption reflects the fact that the assembly of two modules involves the same level of difficulty, whatever the modules are (see Fig. 2).

Let $NA(P_j, C)$ be the number of final assembly operations needed to obtain product P_j from the stock mix C. To determine $NA(P_j, C)$, it is necessary to solve a set partitioning problem [9] determining the minimum number of operations needed to assemble product P_j from the stock mix C. In this paper, this problem is solved with a greedy heuristic presented in [5].

The parameter $p(P_j)$ of this model denotes the probability of demand for product P_j. Each year, the contractor and the supplier negotiate the quantities of each product P_j to be manufactured, however, the sequence of products is not known since it depends on the customers' orders.

The relationship between the mean final assembly time and the number of final assembly operations is

$$\mathrm{AT}(C) = \sum_{j} p(P_j) \times NA(P_j, C). \tag{8}$$

The objective function is hence expressed as

$$\mathrm{Cost} = \alpha \sum_{i=1}^{P} C_i \left(|M_i| - 1 \right) + \gamma.m + \beta \sum_{i=1}^{P} C_i \left(|M_i| \right) + \delta \mathrm{AT}(C). \tag{9}$$

The cost in (9) is a linear function of four different costs. However, the model (1), (2), with the cost function represented by (9), is nonlinear as the final assembly time (AT) is nonlinear.

B. Problem Description

The model (1), (2) with the cost function in (9) is illustrated with the following example. For a simple product family (31 products made from five components), the number of different modules varies from five (make-to-order) to 31 (make-to-stock) products. An exhaustive list of all possible configurations enables determining the extreme assembly times. Fig. 4 represents the minimum and maximum mean final assembly time (y-axis) as a function of the number of different modules (x-axis). This example shows that the mean assembly time varies from 0 to 1.6 time units (t.u.). A mean assembly time of 0 occurs when all possible subassemblies are present in the stock; in this case, whatever the product that is required by a customer, it is present in the stock and final operation is not required.

If the acceptable mean assembly time x is 0.8 t.u. (represented by the dotted line), 16 modules or more guarantee that the average demand is met on time (the maximum mean assembly time for 16 modules is 0.8 t.u.). Note that a "good" configuration of 9 different modules satisfies the maximum time constraint. Different stock compositions (C) make it possible to respect the mean assembly time. Besides, all of them do not have the same cost (9). It is then necessary to select a composition that minimizes this cost.

For the set of modules that satisfy the time constraint (the mean assembly time $< x$), the extreme costs for two sets of parameters $(\alpha, \gamma, \beta, \delta)$ are shown in Figs. 5 and 6.

The proposed set of parameters $(\alpha = 1, \gamma = 2, \beta = 0.4,$ and $\delta = 10)$ is consistent with the industrial practice. The ratio of labor cost between a distant location and a nearby location $\delta/\alpha = 10$ represents the actual situation in Europe (e.g., the
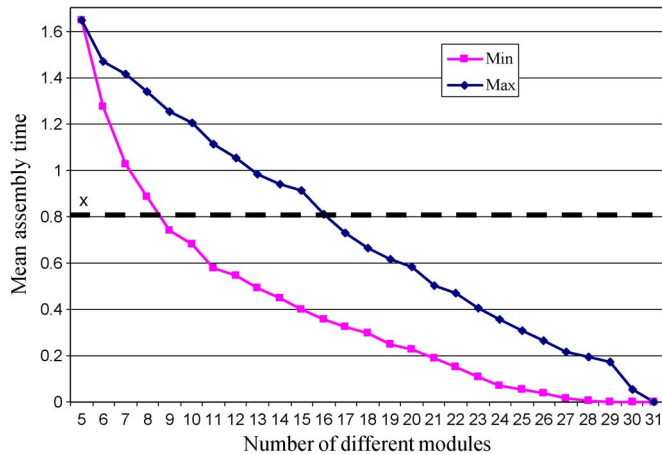
Fig. 4. Function of the min and max mean assembly time and the number of modules.
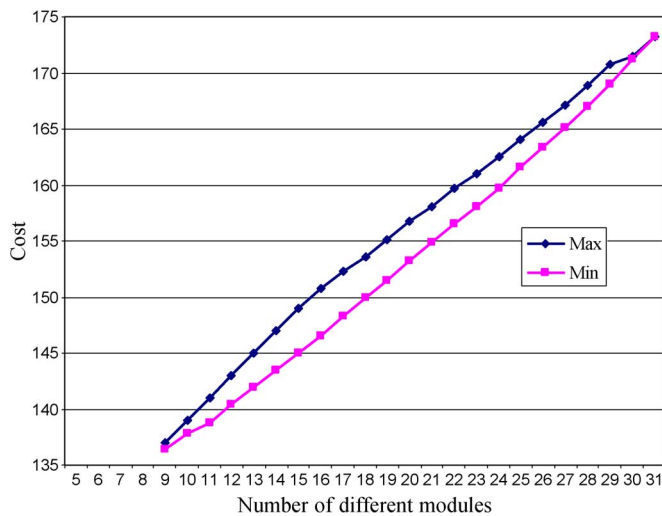

Fig. 5. Extreme costs for $\alpha = 1$, $\gamma = 2$, $\beta = 0.4$, $\delta = 10$.
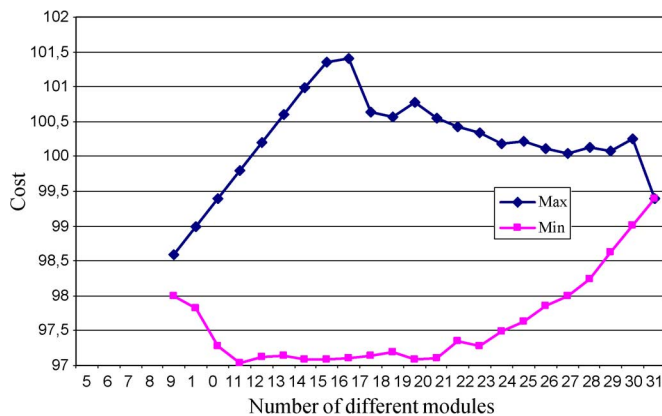

Fig. 6. Extreme costs for $\alpha = 1$, $\gamma = 0.4$, $\beta = 0.1$, $\delta = 10$.

labor cost in France is 25 Euro/h, the labor cost in Latvia is 2.4 Euro/h[2]).

[2] Labor costs in 2000, Ref: IRSOC023 INSEE, 24/12/03, Eurostat European Community.

The parameters $\alpha = 1$, $\gamma = 0.4$, $\beta = 0.1$, and $\delta = 10$ represent a different transportation condition that the supplier wishes to evaluate.

Comparing the results in Figs. 5 and 6, it appears that transportation costs have a significant impact on the best stock composition (C) that minimizes the cost (9). Figs. 5 and 6 are determined by the computation of all possible case with five components. Because of combinational effects, these curves are not determined for a larger number of components.

Using the parameters selected for the case presented (Fig. 5), it is more expensive to stock items than to process them at a nearby location.

In this case, the worst solution (make to order) (i.e., all assemblies are produced at the nearby plant) is 25% more expensive than the optimal one (stocking nine different modules). One observes that the optimal solution minimizes the cost of possessing the modules held as the stock (called a possession stock).

Stocking the minimum number of modules (i.e., the smallest stock that guarantees the customer accepted mean assembly time x) does not always lead to optimum as illustrated in the example presented next.

Consider that the labor is the main source of cost. Stocking the minimum (i.e., nine modules) is not the optimal solution even if it minimizes the transportation cost (6). Interestingly enough, in this case, lower labor costs (the modules are assembled at the distant plant and then transported to the nearby location) is beneficial. Fig. 6 shows that the optimum is reached while stocking 12 modules.

The two examples illustrated in Figs. 5 and 6 show that selecting the right module design is a major issue.

V. SOLVING THE MODEL

What sets of modules should be considered for a product? In fact, this question cannot be answered ex-ante as every possible module mix that enables obtaining all final products has to be considered.

To ensure that each final product could be assembled without extra components (some upgrade of the product), the mix has to include n single components. It is then possible to evaluate the number of compositions to consider. The $P - n$ final products that are not the n single components can be present or not in the stock composition. In total, there are 2^{P-n} different compositions to consider. This number corresponds to all combinations of these $P - n$ elements.

This problem (1), (2) is computationally complex and, therefore, it is reasonable to generate a suboptimal solution. Two heuristics are proposed to minimize the cost (3) under a fixed mean assembly time. Their performance has been studied, and simulated annealing results are provided.

A. Module–Frequency Heuristic

Consider a heuristic algorithm based on the demand frequency for modules (level of demand for each module). This heuristic minimizes the mean assembly time and uses a penalty coefficient that penalizes the modules that are not compatible with the selected module.

For a given number of modules m, the heuristic is as follows:

TABLE I
PRODUCT DEMAND (EXAMPLE WITH 4 COMPONENTS)

	a	b	c	d	ab	ac	ad	bc	bd	cd	abc	abd	acd	bcd	abcd
Product demand	0.01	0.07	0.01	0.06	0.1	0.04	0.09	0.07	0.08	0.01	0.17	0.15	0.05	0.05	0.05

TABLE II
MODULES POTENTIAL USE (EXAMPLE WITH 4 COMPONENTS)

	a	b	c	d	ab	ac	ad	bc	bd	cd	abc	abd	acd	bcd	abcd
Module usage	0.66	0.74	0.45	0.54	0.47	0.31	0.34	0.34	0.33	0.16	0.22	0.2	0.1	0.1	0.05

TABLE III
POTENTIAL MODULES AFTER SELECTION OF MODULE ab WITH PENALTY COEFFICIENT ($pc = 0.05$)

	a	b	c	d	ab	ac	ad	bc	bd	cd	abc	abd	acd	bcd	abcd
Module usage	X	X	X	X	X	0.0155	0.017	0.017	0.0165	0.16	0.0005	0.0005	0.005	0.005	0.0001

1) The n modules composed of unique basic components are selected.
2) For j from 1 to $m - n$:
   a) Select a module that is most frequently used in assembly.
   b) The penalty coefficient (pc) is used: the frequency of the modules sharing common components is multiplied by pc.
   c) Set $j = j + 1$.

The module–frequency heuristic is illustrated with the following example. Consider 4 components that result in 15 final products. The demand probability for each product is given in Table I. In the industrial application, Table I is constructed based on the historical commercial data and marketing projections.

Module ab can be used to obtain the following products: ab, abc, abd, and abcd. Table II represents the usage probability of each module for the product demand in Table I.

To make sure that a component is not used more than once in a given product, a penalty coefficient is used. The same applies to the modules. Assume the penalty coefficient (pc) is 0.05. For six modules ($m = 6$), the first iteration of the heuristic produces the following five modules: a, b, c, d (the basic components have to be available) and ab (highest frequency of assembly). The modules that have components common to ab are penalized. The frequency of each module containing the component a or b is multiplied by pc, the frequency of each module containing components a and b is multiplied by pc^2. The results are summarized in Table III.

In the next iteration, cd is selected.

For six different stocked modules, the composition generated by the module–frequency heuristic with $pc = 0.05$ is then

$$\{a, b, c, d, ab, cd\}.$$

The computational time complexity of the algorithm depends of the number of basic components n. It requires:
- Selection of the n basic components.
- For $i = 1$ to $m - n$.

- Select a module with the highest frequency of assembly (($\boldsymbol{P} - n - i$) elementary selections i).
- After the selection of one of the $m - n$ modules, apply the penalty coefficient.

The total number of elementary operations is then $\sum_{i=1}^{m-n}(\boldsymbol{P} - n - i)$, which is equivalent to a computational time complexity of O($M\boldsymbol{p}$).

The graphs in Fig. 7 show the total cost established with the module–frequency heuristic with the two sets of parameters described above (Section IV-B).

The graphs in Fig. 7 correspond to the ones in Figs. 5 and 6 (maximum and minimum costs for the two instances) and the compositions given by the module–frequency heuristic for two values $pc = 0.05$ and $pc = 1$. The optimal (the solid circle in Fig. 7) and the preferred compositions generated by the module–frequency heuristic algorithm (the dotted circles in Fig. 7) are determined as follows:

$$\text{Preferred composition} = \underset{i=1,P}{\text{Min}} \text{ Result of the heuristic}_{m=i}.$$

The penalty coefficient impacts the results produced by the module–frequency heuristic. The results in Fig. 7 stress the fact that using a small pc value is important when the number of modules stocked is large. The consequences of the penalty coefficient accumulate at each selection.

A small penalty coefficient impacts only the selection of modules after several uses (i.e., the number of different modules (m) that actually benefits is relatively high).

It has been demonstrated that for a large value m, the fixed transportation cost is low ($\gamma \ll \alpha + \beta + \delta$). Therefore, one concludes that a small penalty coefficient works better (i.e., the given composition will lead to smaller costs) when the relative weight of the number of different modules is small.

Table IV provides the excess cost of this composition which is computed as follows:

Excess cost of composition Ci
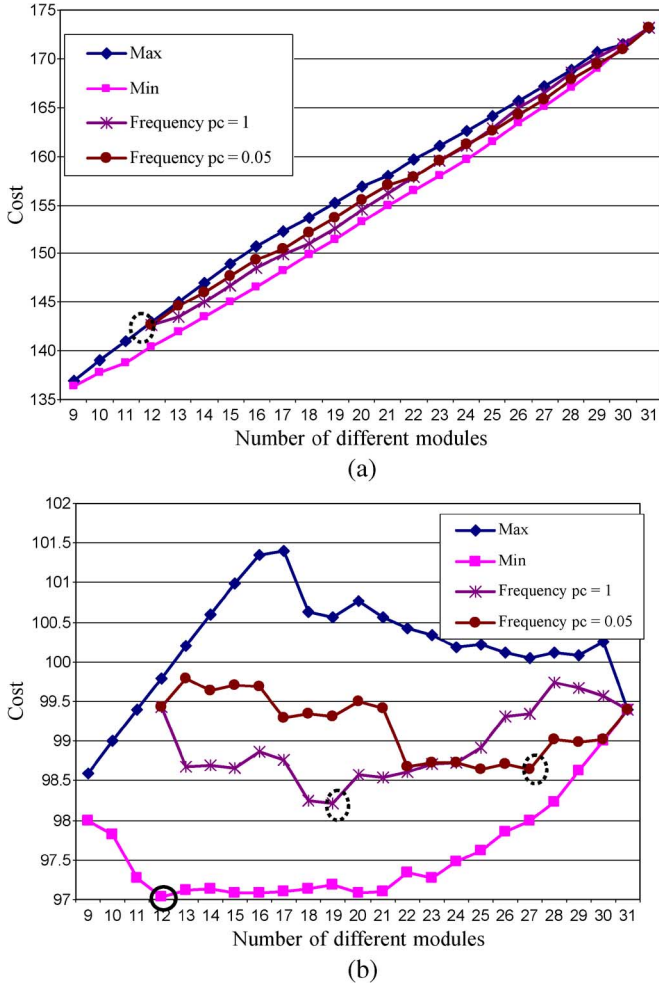$$= \frac{\text{Cost\_of\_composition\_}Cj - \text{Optimal\_cost}}{\text{Optimal\_cost}} \times 100.$$

Fig. 7. Results produced by the module–frequency heuristic. (a)—($\alpha = 1$, $\gamma = 2$, $\beta = 0.4$, $\delta = 10$); (b)—($\alpha = 1$, $\gamma = 0.4$, $\beta = 0.1$, $\delta = 10$).

TABLE IV
COSTS AND RELATIVE EXCESS COSTS WITH
THE MODULE–FREQUENCY HEURISTIC

	Instance 1	Instance 2
Optimal cost	136.4	97.1
Worst case cost	171.2	101.4
Worst excess cost	25.5%	4.4%
Heuristically generated cost ($pc = 0.05$)	142.6	99.4
Heuristically generated excess cost ($pc = 0.05$)	4.1%	2.3%
Heuristically generated cost ($pc = 1$)	142.6	98.9
Heuristically generated excess cost ($pc = 1$)	4.1%	1.8%

The module–frequency heuristic determines solutions that are better than the worst-case solution. Nevertheless, the gap between the generated solution and the optimal one is not negligible (about 4% for the first instance).

The module–size heuristic is discussed in the next section.

B. Module–Size Heuristic

Consider a heuristic based on the module size (the number of components in a module). Using the size of the module as a selection criterion seems logical. Of course, the size of the module

TABLE V
COST AND RELATIVE EXCESS COST WITH THE MODULE-SIZE HEURISTIC

	Instance 1	Instance 2
Optimal cost	136.4	97.1
Worst case cost	171.2	101.4
Worst excess cost	25.5%	4.4%
Heuristically generated cost	140.8	97.95
Heuristically generated excess cost	3.2%	0.87%

does impact the preassembly cost and the transportation cost. Furthermore, small modules are usually used in the assembly of more products than the large modules (e.g., a module composed of a unique component can potentially be used in $2^{(n-1)}$ products).

Given the number of different modules m, the module–size heuristic is as follows.

1) Select all modules of size $\leq j$, such as

$$\sum_{i=1}^{j} \binom{n}{i} \leq m \leq \sum_{i=1}^{j+1} \binom{n}{i}$$

$\binom{n}{i}$ is the number of ways of picking i components from the n possibilities: it is the number of different products constituted of i components.

2) Sort the $\binom{n}{j+1}$ modules that contain $j+1$ components in a descending order of frequency of assembly.

3) Select the $(m - \sum_{i=1}^{j} \binom{n}{i})$ modules that have the highest frequency of assembly.

The computational time complexity of the algorithm is $O(m \, log(m))$.

For six modules, the composition provided by this heuristic is $\{a, b, c, d, ab, ad\}$.

Using the module–size heuristic, a composition can be found for each m. The best selection is performed as in the module–frequency heuristic.

Table V provides the excess cost of this composition. It is computed as explained in Section V-A.

Note in the two examples that the best composition generated by the module–size heuristic is close to optimum.

C. Simulated Annealing

A simulated annealing algorithm was applied for solving the problem discussed in the paper. Introduced in 1980 [14], simulated annealing has been applied successfully to many types of problems [23]. This metaheuristic has already been used for the design of modules. Kreng and Lee [16] integrated functional and physical relationships.

The search space consists of all compositions allowing the assembly of all finished products without standardization. All compositions must contain the necessary elementary components a_i. In total, $2^n - 1 - n$ modules are potentially present in the stock C. All of those that meet the constraints constitute the search space. The size of the search space is too large for an exhaustive search and the objective function does not allow for a gradient approach.
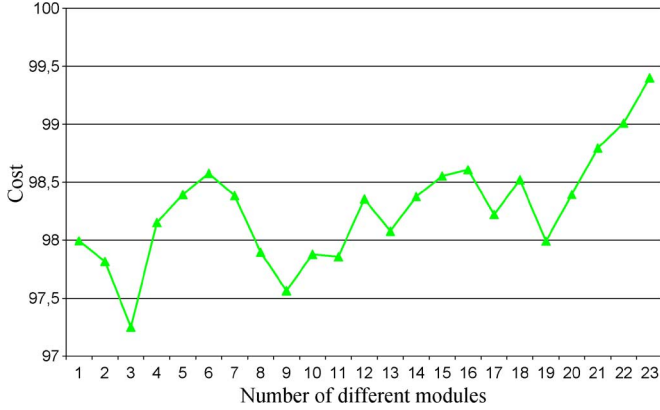
Fig. 8. Simulated annealing of instance 1.

Moreover, it is not possible to determine *a priori* if a stock composition that belongs to the search space respects the time constraint (2).

The constraint represented in (2) is integrated in the evaluation function as follows:

$$\text{Cost} = \begin{cases} \text{Cost}, & \text{if } \text{AT}(C) \le x \\ \text{Cost} + \text{penalty} * (\text{AT}(C) - x), & \text{else.} \end{cases}$$

The proposed move generates only neighbors that belong to the search space (acceptable configurations). Two configurations C and C' are considered as neighbors if they differ in one module, i.e.

$$\exists i \in [[1, 2^n - 1]] \text{ such as } \forall k \in [[1, 2^n - 1]] \setminus i$$
$$(C_k = C'_k \text{ and } C'_i = (1 - C_i))$$

or

$$\sum_{i=1}^{P} C_i = \sum_{i=1}^{P} C'_i$$

The difference between the two costs C and C′ is denoted as δCost.

$$\delta\text{Cost} = \text{Cost}(C') - \text{Cost}(C).$$

A neighbor with a lower cost (δCost = 0) is systematically accepted. A neighbor with a higher cost (δCost > 0) can be also accepted. For acceptance of a higher cost neighbor, a random value (in [0; 1]) is generated. If this random value is under $e^{-(\delta\text{Cost}/T)}$, then the move is accepted. T is updated after each calculation of T as follows:

$$T \leftarrow T \times \text{Cooling\_coefficient}.$$

The cooling coefficient ensures that T decreases to 0, so that the simulated annealing algorithm converges.

As the efficiency of simulated annealing greatly depends of the initial value, the value selected for starting the computation is the best result produced by the previous heuristic. The observed results are summarized in Fig. 8.

	Instance 1	Instance 2
Optimal cost	136.4	97.1
Lowest case cost	171.2	101.4
Lowest excess cost	25.5%	4.4%
Simulated annealing generated cost	136.6	97.25
Simulated annealing generated excess cost	0.15%	0.15%
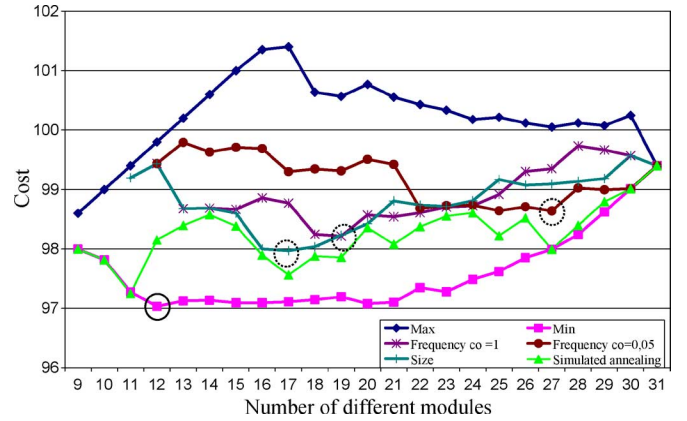
Fig. 9. Comparative analysis of the two heuristics $\alpha = 1, \gamma = 0.4, \beta = 0.1, \delta = 10$.

For the two similar instances similar to the previous ones, the final results with simulated annealing are presented in Table VI.

D. Performance Evaluation

The heuristics were tested on a large number of instances (i.e., different sets of parameters and different demands). The results were consistent with the examples presented in the previous sections of this paper (the graph in Fig. 9 summarizes the performance of the two heuristics). The module–size heuristic quickly determines a good stock composition. The simulated annealing algorithm used the results of the frequency and size heuristics as the initial solutions. The dotted circles show the minimum cost computed with each heuristic. The full circle shows the optimum computed with all of the heuristics.

Note that when the number of components is large, generating an optimal solution is not feasible as the number of possible compositions is excessive (see Section V-B). The performance of a module mix for a large portfolio of products cannot be easily evaluated.

Any of the two heuristics is easy to implement. Therefore, to select the right mix of modules, both heuristics may be applied (using a set of different penalty coefficients for the module–frequency heuristic) and the composition that provides the lower total cost may be selected. In the interest of the decision maker's time, the module–size heuristic is recommended, though both heuristics provide good quality results.

The simulated annealing algorithm solves the problem. Providing initial solutions leads to better quality solutions and reduces computing time.

VI. CONCLUSION

This research emphasized the importance of demand for final products when allocating assembly operations in a supply chain. This decision impacts the overall cost structure, as it determines the assignment of operations and inventory performed at locations with different assembly costs. Based on the industrial case study, a cost function is proposed that integrates operations performed at different locations and the time-to-market constraint.

Two heuristic algorithms were designed to solve this combinatorial problem. The effectiveness of the algorithms was illustrated with examples and an industrial case study. Numerical examples illustrated that a "good" design can save up to 25% of the total production cost. Two heuristics were proposed to select a proper stock mix. The first heuristic is based on the frequency of components usage across different products. The second one is based on the module size and it performed better. Simulated annealing is useful to improve previous solutions.

If no acceptable solution (stock mix) is available for a given mean assembly time, a supplier may need to renegotiate the contract for a longer lead time.

In the model considered in this paper, an assumption was made that dealers were not intervening in the product personalization (e.g., changing wheels or adding a sunroof to a car). Future research will lead to the investigation of less restrictive models. For example, the use of multiple locations with different labor costs and different transport costs could be considered. Furthermore, models with a stock-level decision variable could be developed.

For decision makers, an important parameter is the robustness of the solution they want to implement. In this particular case, it could be interesting to study the performances of the results given by the heuristics, when the weights of the costs and/or the demand change. Obtaining this information could be the subject of further work.

REFERENCES

[1] B. Agard and A. Kusiak, "Data-mining based methodology for the design of product families," *Int. J. Prod. Res.*, vol. 42, no. 15, pp. 2955–2969, 2004.
[2] B. Agard and M. Tollenaere, "Design of wire harnesses for mass customization," presented at the 4th Int. Conf. Integrated Design and Manufacturing in Technical Engineering, Clermont-Ferrand, France, 2002.
[3] B. Agard and M. Tollenaere, "Design of product families: methodology and application," presented at the 14th Int. Conf. Engineering Design, Stockholm, Sweden, 2003.
[4] O. Briant and D. Naddef, "The optimal diversity management problem," *Oper. Res.*, vol. 52, no. 4, pp. 515–526, 2004.
[5] V. Chvátal, "A greedy heuristic for the set covering problem," *Math. Oper. Res.*, vol. 4, no. 3, pp. 233–235, 1979.
[6] P. Child, R. Diederichs, F. H. Sanders, and S. Wisniowski, "The management of complexity," *Sloan Manag. Rev.*, vol. 33, no. 1, pp. 73–80, Fall, 1991.
[7] C. da Cunha, "Définition et gestion de produits semi-finis en vue d'une production de type assemblage à la commande," Ph.D. dissertation, Organis. Ind. Syst. Prod., Inst. Nat. Polytechni. Grenoble, Grenoble, France, 2004.
[8] F. Erens and K. Verhulst, "Architectures for product families," *Comput. Ind.*, vol. 33, no. 2–3, pp. 165–178, 1997.
[9] K. Fujita, "Product variety optimization under modular architecture," *Comput.-Aided Design*, vol. 34, no. 12, pp. 953–965, 2002.
[10] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.
[11] J. P. Gonzalez-Zugasti, K. Otto, and J. Baker, "A method for architecting product platforms," *Res. Eng. Design*, vol. 12, no. 2, pp. 61–72, 2000.
[12] C. C. Huang and A. Kusiak, "Modularity in design of products and systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 28, no. 1, pp. 66–77, Jan. 1998.
[13] J. Jiao and M. Tseng, "A methodology of developing product family architecture for mass customization," *J. Intell. Manuf.*, vol. 10, no. 1, pp. 3–20, 1999.
[14] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.
[15] I. Kocher and P. Rolland, "Quand les structures dessinent les produits," *Gérer comprendre*, vol. 39, pp. 38–48, 1995.
[16] V. Kreng and T. P. Lee, "Modular product design with grouping genetic algorithm—a case study," *Comput. Ind. Eng.*, vol. 46, no. 3, pp. 443–460, 2004.
[17] A. Kusiak and C. C. Huang, "Development of modular products," *IEEE Trans. Compon., Packag., Manuf. Technol. A*, vol. 19, no. 4, pp. 523–538, Dec. 1996.
[18] A. Kusiak, *Engineering Design: Products, Processes and Systems*. San Diego, CA: Academic, 1999.
[19] H. L. Lee and C. S. Tang, "Modeling the costs and benefits of delayed product differentiation," *Manag. Sci.*, vol. 43, no. 1, pp. 40–53, 1997.
[20] J. P. MacDuffie, K. Sethuraman, and M. L. Fisher, "Product variety and manufacturing performance: Evidence from the international automotive assembly plant study," *Manag. Sci.*, vol. 42, no. 3, pp. 350–369, 1996.
[21] M. V. Martin, "Design for variety: A methodology for developing product platform architectures," Ph.D. dissertation, Dept. Mech. Eng., Stanford Univ., Stanford, CA, 1999.
[22] S. K. Martin, "Modular production—a new concept," *Harvard Bus. Rev.*, pp. 131–142, Nov./Dec. 1965.
[23] R. Otten and L. van Ginneken, *The Annealing Algorithm*. Norwell, MA: Kluwer, 1989.
[24] J. Swaminathan and S. Tayur, "Managing broader product lines through delayed differentiation using Vanilla Boxes," *Manag. Sci.*, vol. 44, no. 12, pp. 161–172, 1998.
[25] D. Twede, R. H. Clarke, and J. A. Tait, "Packaging postponement: A global packaging strategy," *Packag. Technol. Sci.*, vol. 13, no. 3, pp. 105–115, 2000.
[26] A. S. Yigit, A. G. Galip-Ulsoy, and A. Allahverdi, "Optimizing modular product design for reconfigurable manufacturing," *J. Intell. Manuf.*, vol. 13, pp. 309–316, 2002.
[27] J. S. Yu, J. P. Gonzalez-Zugasti, and K. Otto, "Product architecture definition based upon customer demands," in *Proc. DETC ASME Design Theory and Methodology Conf.*, Atlanta, GA, 1998, pp. 271–277.

Catherine da Cunha received the Ph.D. degree from the Institut National Polytechnique de Grenoble (INP), Grenoble, France, in 2004.

Currently, she is an Assistant Professor in industrial engineering with the Laboratoire GILCO, École Centrale de Nantes, Nantes, France. Her main interests are in the area of information use within the firm and the production and organizational field. She is particularly interested in the potential of partial or immature information for decision aid.

Bruno Agard graduated in manufacturing from École Normale Supérieure de Cachan, Paris, France, in 1998 and received the Ph.D. degree in industrial engineering from the Institut National Polytechnique de Grenoble, Grenoble, France, in 2002.

Currently, he is an Assistant Professor of Industrial Engineering with the École Polytechnique de Montréal, Montréal, QC, Canada. He is interested in the design of products, processes, and logistics and in applications of data mining in engineering.

Andrew Kusiak (M'89) is a Professor in the Department of Mechanical and Industrial Engineering at the University of Iowa, Iowa City, Iowa. He is interested in applications of computational intelligence in automation, energy, manufacturing, product development, and health care. He has published many books and technical papers in journals sponsored by professional societies, such as the American Association of Artificial Intelligence (AAAI), Institute of Industrial Engineers (IIE), International Federation of Information Processing (IFIP), Society of Manufacturing Engineers (SME), and other professional organizations. He speaks frequently at international meetings, conducts professional seminars, and consults for industrial corporations. He has served on editorial boards of more than thirty journals. He is the Editor-in-Chief of the *Journal of Intelligent Manufacturing*

Prof. Kusiak is an IIE Fellow.