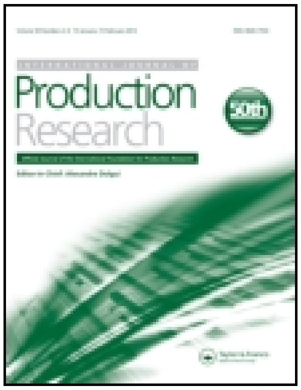

International Journal of Production Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tprs20

Modular design of product families for quality and cost

Bruno Agard [a] [b] & Samuel Bassetto [b]

[a] Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport, CIRRELT

[b] Départment de Mathématiques et Génie Industriel, École Polytechnique de Montréal, Montreal, Québec, Canada

Version of record first published: 06 Jul 2012.

To cite this article: Bruno Agard & Samuel Bassetto (2013): Modular design of product families for quality and cost, International Journal of Production Research, 51:6, 1648-1667

To link to this article: http://dx.doi.org/10.1080/00207543.2012.693963

International Journal of Production Research
Vol. 51, No. 6, 15 March 2013, 1648–1667

Modular design of product families for quality and cost

Bruno Agard[ab*] and Samuel Bassetto[b]

[a]*Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport, CIRRELT;*
[b]*Départment de Mathématiques et Génie Industriel, École Polytechnique de Montréal, Montreal, Québec, Canada*

(*Received 24 October 2011; final version received 19 April 2012*)

The purpose of this article is to help managers early in the design of new product families. Based on product structures, sales forecasts, and constraints imposed by the marketplace, like quality and cost, the proposed method selects the product modules that meet customer requirements for the products, while respecting those constraints. The proposal includes a single-level module design formulation that considers quality and cost simultaneously. The method for testing the proposed algorithm is based on a case study of an electro–mechanical assembly device (headlamp). The performance of the algorithm is compared to that of the zero module case, where often the constraint problem cannot be resolved. The main result is a model and an algorithm that optimise quality and cost under the constraints of quality and cost. It shows what modules to manufacture, in what quantities, and in which products to use them. The output also provides the predicted quality and cost, based on improvements made to the modules. To conclude, this research enables the joint optimisation of quality and cost by defining the modules to be manufactured. It provides input for managers seeking modules designed for their supply chain. The algorithm provides key input for managing production ramp-up.

Keywords: modularity; design for quality; design for cost; assembly; optimisation

1. Introduction

Mass customisation and pricing competition force companies to develop new strategies to cope with greater flexibility, while remaining competitive in terms of price and delivery time (da Cunha *et al.* 2010). These strategies are undoubtedly key elements in gaining competitive advantage, or at least remaining competitive. However, customers require fully functional products, whatever the price. Their tolerance of product malfunctions is often very low. If a product is labelled "industrial", whether it is a low cost one (T-shirt, computer flash drive, pen) or a low volume one (airplane, substation circuit breakers, wind turbine), the manufacturer is expected to have fully understood its characteristics. The product is supposed to work properly and faultlessly, and any variability in its functions can be considered a risk to meeting the customer's requirements.

Manufacturers put controls in place to master every level of their processes (Baud-Lavigne *et al.* 2010), and barriers are deployed throughout the manufacturing system to prevent faults from occurring (Hollnagel 2008). While the integration of quality and quantity has been investigated in classical manufacturing lines (Tapiero 1987, Colledani and Tolio 2011), the interaction between quality and supply chain design in modular design has not, and constitutes an opportunity for investigation, as revealed in our literature review.

The concept of quality adopted in this paper conforms to the view of manufacturers and supply chain managers. It is the degree of conformance of products to predefined specifications and standards. This degree is measured by process controls and inspections. Actions on quality have an impact on a global defect rate. Through this paper, the degree of conformance will be appreciated by the final failure rate of a product.

It is sensible to preassemble parts, or at least to manufacture them together, for a specific set of products. These sets are called modules, and they are employed to solve diversity issues, like determining an optimal threshold for manufacturing quantity. The creation of modules should lead to efficiencies in terms of reduced assembly time and overall cycle time, while maintaining high potential for diversity. When modules are produced from components, resulting modules may have different quality levels than their components, depending on actions that have been performed during the manufacturing. The resulting quality of a module could be increased (for instance modules

*Corresponding author. Email: bruno.agard@polymtl.ca

ISSN 0020–7543 print/ISSN 1366–588X online

http://dx.doi.org/10.1080/00207543.2012.693963
http://www.tandfonline.com

could be "sort" or "test") or decreased (for instance, in the case of handling that produce scratches or faults on modules). There is then a possible action on quality each time a module is created – either a positive one or a negative one.

This paper investigates module design considering quality, cost, and the product family mix.

It is structured in five main parts: in Section 2, we present a literature review of modular design and risk management; in Section 3, our model; in Section 4, a case study; and in Section 5, the simulation and its results, followed by a discussion.

2. Literature review

2.1 *Diversity management*

Diversity in medium and large product families is supported by two main approaches: modular design and postponement (delayed product differentiation). Diversity in product families is supported by different approaches such modularity, commonality, postponement (delayed product differentiation), scalable design, standardisation and flexible manufacturing. All those approaches are related and support different fundamental issues in the product portfolio, product platform, process platform and supply platform (Jiao *et al.* 2007). Modular design involves the creation of largely independent blocks of components (called modules) for building different products. By combining limited numbers of modules, a product can be diversified, potentially resulting in a large number of final products.

Modular design has a major impact on the manufacturing system. The number of modules stocked and the make-up of each module (Agard and Kusiak 2004b), the delivery time (based on the final assembly time) (Agard *et al.* 2009), and the production cost (da Cunha *et al.* 2007) can vary, depending on the modules chosen. Modular design has received a great deal of attention in the literature (Cheung and Leung 2000, Wang and Wang 2003, Li and Lin 2006). To expand the concept of production cost, the quality aspects of production can be considered, as every module behaves differently in terms of quality. The potential for loss of quality generates several quality loss functions, which depend essentially on the quality policies in place throughout the supply chain. Ultimately, if quality is taken into account, the rationale for the choice of modules can be revised. Quality improvement and design team costs are studied by (Wu *et al.* 2009).

Modularity concepts can be implemented by partitioning a product into semi-independent or mutually divisible elements, as this makes it possible to design, manufacture, and service the modules individually (Kusiak and Huang 1996, Kusiak 1999). (Fujita 2002) optimised the content and selection of modules in a fixed modular architecture. (Yigit *et al.* 2002) solved a similar problem by determining the subsets of modules that minimise the reconfiguration cost. A wide diversity of products supported by minimal technical diversity reduces process diversification, which guarantees acceptable product development and manufacturing costs (MacDuffie *et al.* 1996). Reducing variety in production is often considered to result in yield improvement, which makes modular design a promising strategy for contributing to lean manufacturing (Stump and Badurdeen 2009).

Postponement is the second approach that supports product diversity. Its purpose is to delay as much as possible the moment when the product attains its uniqueness (identity). For example, packaging postponement (Twede *et al.* 2000) involves delayed differentiation of the product until the packaging operation takes place. (Lee and Tang 1997) have highlighted the advantages of delayed differentiation of the product in the manufacture of diversified products.

Product and process standardisation are highlighted as optimisation strategies, as is process restructuring. Some research results have been reported in both the design-to-cost and assembly-to-order (ATO) contexts (Agard and Penz 2009). (Swaminathan and Tayur 1998) optimised production capacity using preassembly operations.

ATO is "a production environment where goods (or services) are assembled after the receipt of a customer's order" (Agard and Penz 2009). It enables a wide diversity of products to be managed with a limited number of modules that are preassembled, shipped to the assembly location, and stocked. Final assembly of the product (from modules) is initiated when the actual order is received.

The product demand pattern (volume of demand for a specific product) impacts a supply chain (in terms of storage, transportation, and production costs). Having information about customer demands and assigning the workload to various actors in the chain could reduce this cost. (da Cunha *et al.* 2010) used product demand data to design modules to minimise the final assembly cost (by minimising the mean assembly time). The computational results showed that significant savings could be realised by taking such data into account. As explained in (Agard and Kusiak 2004a), the data on product make-up are used to design a supply chain where differing labour costs must be considered.

So far, however, the authors have found few papers that link either modular design or postponement to quality. Only one recent research has been retrieved that integrates quality and modularity in a reverse logistic perspective and closed loop supply chain (Das and Chowdhury 2012). They propose an optimal design of supply chain, while maximising the quality of the final product. The source of variation of component quality is found in their origin (reused components). Their decision on quality is then based on the reuse or not of a particular component.

In fact, quality can play a central role in both strategies. Modules without it can seriously disrupt the entire supply chain. If a module is used in a large number of products, quality issues (non conforming parts, for example) could lead to major production system disruptions, underlining the brittleness of the optimal solutions proposed to date.

2.2 *Quality analysis – for product, process, and supply chain*

Plenty quality reviews and analysis methods have been issued to drive designs in an acceptable, if not optimal, quality zone. The fight against the risk of poor quality is part of the design check and the industrial process. There are many techniques available for analysing the risks associated with products, processes, and supply chains, 62 of which have been presented in (Tixier *et al*. 2002). Industry often employs failures, modes, effects, and criticality analysis (FMECA) (Department of Defense 1980) to systematically scrutinise every component or function of a system, with the objective of signalling potential failures, and ultimately preventing them. Classical applications of FMECA can be found in (Eubanks and Ishii 1997, Kmenta *et al*. 1999, Shahrokhi and Bernard 2004). Although this method is much criticised in the literature (Carmignani 2009), FMECA is a robust way to structure a systematic investigation of a system. It can also act as a risk analysis deliverable in supplier–customer contracts.

In the design community, an effort has been made to base design decisions on updated data on earlier products, observing the failures that have occurred during the life cycle of those products and taking them into account during the design stages of similar products. Along with the key characteristics approach (Thornton 1999), the function failure design method (Stone *et al*. 2005) and the risk in early design method (Lough *et al*. 2006a) provide a taxonomy of failures to help designers update their knowledge of functions and part failures (Tumer and Stone 2003) by bringing them to their attention. These methods have been applied to mechanical design in the aerospace industry (Stone *et al*. 2002), and a number of metrics have been evaluated to select the risks on which to focus (Lough *et al*. 2006b). Another approach to creating a risk taxonomy has been presented by Stamatis (2003) in his reference book. A selection of classic failure types from several domains is used as a checklist to ensure that analyses are complete. (Ebrahimipour *et al*. 2010) built an FMECA-based ontology with the Protégé tool, to better structure and enrich such a checklist. A link between the design of manufacturing processes and FMECA has also been investigated by (Hassan *et al*. 2010). The IRAD method was proposed by (Ghemraoui *et al*. 2009) to integrate the requirements of safety considerations early in the design process, instead of adding them at the end of it. With a knowledge of failure versus function, the design team can modify the product structures and their associated modules.

Another side of FMECA has also been explored in the less theoretical, more operational literature, in which the failure modes, failure causes, and failure effects of several kinds of FMECA are linked. In the VDI norm, the links between system and subsystems are reflected as links in risk analysis (Bertsche 2008). Failure functions can be a mode, an effect, or a cause, depending on the level of decomposition of the system analysed. Viewed as a system, the supply chain can be decomposed, and the associated risks can be found and treated. As links exist among organisations in a supply chain, their risks are also connected, according to (Bertsche 2008). In supply chain design, risk analysis has been used to identify key parameters to monitor during the design and setup of a particular supply chain (Larson and Kusiak 1996, Tuncel and Alpan 2010).

FMECA and other risk analysis techniques are not enough to control the risks that can occur during operations. The concept of the risk barrier has been developed in order to help prevent the occurrence, and propagation, of negative events at several levels (Hollnagel 2008). Examples of such a barrier are the final acceptance test and incoming quality controls.

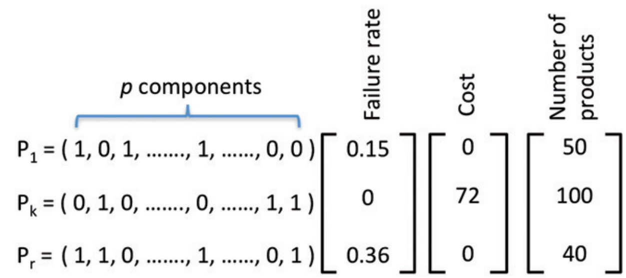

Figure 1. Product modelling.

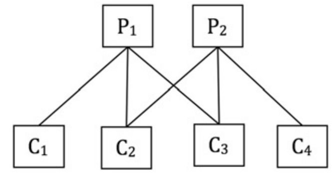

Figure 2. Products P_1 to P_3 contain components C_1 to C_5.

2.3 *Literature conclusion*

The literature review encompasses a broad spectrum of articles. On the one hand, modules have been developed with a view to simplify the supply chain and make it more competitive. However, the notion of quality has not been seen as a major concern in these developments. It seems that module design has little concern for risks in general, and quality in particular.

On the other hand, risk management is a well-developed field, and major advances have been made on product and process design. However, the authors have not found any research on the connection between risk and modular design in the risk management literature. Of course, some manufacturers may make that connection and reflect it in their practices without publicising the fact.

We have concluded that the opportunity exists to develop a full connection between risk and modular design.

3. Problem formulation

3.1 *Description of the product*

Consider P to be a set of products to manufacture. Product P_k is made up of a set of components C_i. r products are considered: $k \in [1, r]$. The components are called C_i, $i \in [1, p]$. In its description, a product is represented as a vector of size p that expresses the components that are present (1) or absent (0) in it (see Figure 1). For example, $P_1 = (1, 0, 1, \ldots, C_i = 1, \ldots, C_p = 0)$ means that product P_1 contains components $C_1, C_3, \ldots C_i, \ldots$, but not $C_2, \ldots, C_p$, and so on. Every product Pk has a failure rate $\rho(Pk)$ and a cost $Cost(Pk)$, and must be produced in a certain quantity $Q(Pk)$.

For each product Pk, if a failure rate (resp. cost) constraint is to be solved ($p(Pk) \neq 0$; resp. $Cost(Pk) \neq 0$) then $Cost(Pk)$ (respectively $p(Pk)$) indicates 0. This modelling is used for the solving branching in Equations (1) and (2).

Different products P_k may contain the same components C_i. In Figure 2, products P_1 and products P_2 share components C_2, which means that P_1 contains a specimen of C_2 and that P_2 contains a specimen of C_2. P_1 and P_2 also share component C_3.

A module M_j is a set of components C_i.

In Figure 3, M_1 is the module that contains components C_2 and C_3.

Various options are available in product manufacturing:

- Option (A): use all the necessary individual components for each product. This is the basic assembly process.
- Option (B): use a mix of components and modules for each product. This is the module creation process and its uses.

For instance in Figure 4, following option A, product P1 would be made of raw assembly of C1, C2, and C3, and P2 would be made of C2, C3, C4. Option B would generate the module M1, made of C2 and C3. Then product P1 would be made of C1 and M1, and C2 would be made of C4 and M1. The design of M1 enables actions on its costs and its quality.

Figure 4 shows the options for manufacturing products P_1 and P_2.

A quality issue is understood as a failure that occurs in a product, a component, or a module. A failure can occur during manufacture or during an assembly operation. It can also appear later, at which point it is referred to as a

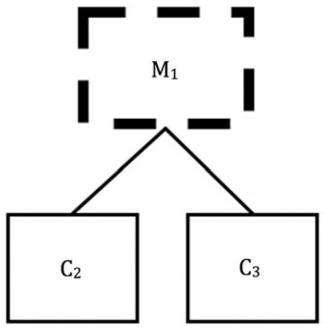

Figure 3. Description of module M_1.

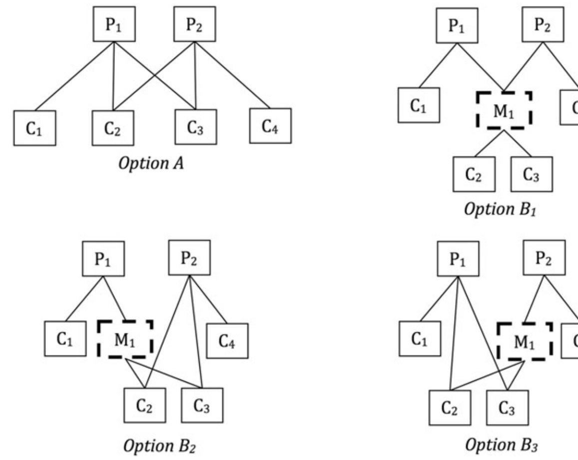

Figure 4. Problem description.

reliability issue. Its main characteristic is to propagate along the supply chain, and such problems are rarely detected by classical functional tests.

In manufacturing the products in P, workers select the components (or modules) needed from different lots. Each lot comes from a specific contractor who guarantees the reliability of the entire lot. The failure rate for the set of modules Mj is $\rho(M_j)$. For each lot, $\rho(M_j) \in [0, 1]$, $\rho(M_j) = 0$ means that all the products are reliable, $\rho(M_j) = 1$ means that 100% of the modules M_j are faulty. Different scenarios of failure rates are studied below. Failure rates depend on:

- the ability of suppliers to produce reliable components,
- the ability of suppliers to identify unreliable components in their processes, and
- the ability of shipping and incoming departments to identify quality defects.

The same applies to the cost of each component and module.

In option (A) (Figure 4), the failure rate and cost of P1 will depend on components C_1, C_2, C_3, and in option (B_1), the failure rate and cost of P_1 will depend on C_1 and M_1, and so on.

Depending on the objective for each type of product (in terms of cost and failure), we look to answer the following questions: *Is it better to buy a module M_1 instead of two components C_2 and C_3?* and *Is it better for P_1 and/or P_2 to use it?*

3.2 *Mathematical modelling*

3.2.1 *Notations*

The notations are the following:

- C_i is a component;
- C is the set of components Ci, $Ci \in [1, p]$;
- M_j is a binary vector of size p, called module j. The vector represents the components it contains; for example, module $M1 = (1, 0, —, 0)$ contains only component C_1. A component can also be considered like a module (with only one component).
- M is the set of modules Mj, $Mj \in [1, q]$;
- P_k is a binary vector of size p, called product k. The vector represents the component it requires; for example, product $P_1 = (1, 0, 1, \ldots, C_i = 1, \ldots, C_p = 0)$ means that product P_1 contains components C_1, $C_3, \ldots C_i, \ldots$, but not $C_2, \ldots, C_p$, and so on.
- P is the set of products Pk, $Pk \in [1, r]$;
- $Cost(Ci)$, $Cost(Mj)$, and $Cost(Pk)$ are the cost of Ci, Mj, and Pk respectively;
- $\rho(Ci)$, $\rho(Mj)$, and $\rho(Pk)$ are the failure rates of Ci, Mj, and Pk respectively;
- $Q(Pk)$ is the quantity of products Pk to manufacture;
- x is a binary vector of size q, such that $xj = 1$ if $Mj \in M'$. It is the decision variable.

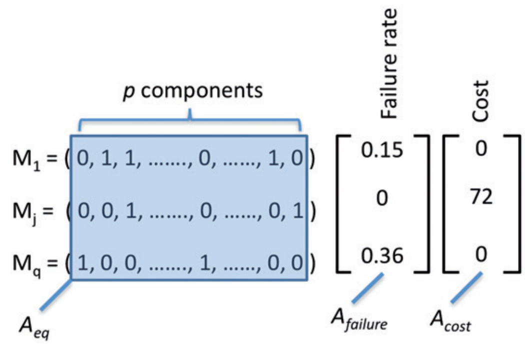

Figure 5. Module modelling.

The goal is to determine the subset of modules M′ Є M of minimum cost, such that all products in P can be built, each product Pi respecting its own constraints. If a product Pk has no failure rate constraint, then $\rho(Pk) = 0$, in which case product Pk has a maximum cost constraint. If a product Pk has no cost constraint, then $Cost(Pk) = 0$, in which case product Pk has a maximum failure rate constraint.

Two kinds of constraints exist in P: for r_1 products, there is a maximum cost constraint, and for r_2 products, there is a maximum failure rate constraint ($r_1 + r_2 = r$). Each product in P may have a different constraint.

We call (Figure 5):

- Aeq a binary matrix of size $q \cdot p$ formed by all products in P.
- $A_{failure}$ a vector of size q that contains failure rates for all modules in M.
- A_{cost} a vector of size q that contains costs for all modules in M.

3.2.2 *Problem formulation*

The formulation is the following:

$\min_x C(x)$ such that for all k in $[1, r]$,

$$\begin{cases} \text{if } Cost(Pk) = 0, \quad A_{failure} \cdot x \le \rho(Pk) & (1) \\ \text{if } \rho(Pk) = 0, \quad A_{cost} \cdot x \le Cost(Pk) & (2) \end{cases}$$

$$Aeq^T \cdot x = Pk \tag{3}$$

where

$$C(x) = \sum_j \delta_{jk} Cost(Mj) \cdot Q(Pk) + \sum_j x_j \cdot G \tag{4}$$

$\delta_{jk} = 1$, if product Pk contains module Mj

If $Cost(Pk) = 0$ a quality constraint is to be solved for product Pk (Equation (1)), if $\rho(Pk) = 0$ a cost constraint is to be solved for product Pk (Equation (2)).

$C(x)$ is the total cost of the whole product family, and represents the sum of the costs of all the necessary modules (based on the quantity of products, and so the number of each type of module), plus the total number of modules multiplied by a management cost G per module.

The management cost has been shown to have a major impact on the number of modules in the final product solution (da Cunha *et al.* 2007, Agard *et al.* 2009). For computation purposes, G is assigned a fixed value for all the experiments, so that the quality and cost of modules can be compared for analysis.

This problem includes the set partitioning problem (Equation (3)), which is then NP-hard in the strong sense (Garey and Johnson 1979).

3.3 *Problem solving*

This optimisation problem cannot be solved by standard optimisation software for large instances. As explained previously, the problem is an NP-hard 0–1 optimisation problem.

In order to arrive at an approximate solution, we adopted a simulated annealing procedure. Figure 6 presents the general scheme of the algorithm.

(1) After the input data have been read (a description of the products to manufacture, Figure 1, a list of possible modules, Figure 5, and a parameter solution are generated), the first step, called ''weight and filter'', follows: for each module, the number of use cases is evaluated (a simple comparison of Mj and Pk, where $Pk(i)$ should always be higher than or equal to $Mj(i)$). The use case value of each module represents its weight. A weight of 2 means that the module could potentially be used in two different products from the input data. All modules with a weight equal to 0 are deleted from the search space. This is the filter operation.

(2) An initial solution is selected and evaluated. The initial solution (x) is constructed, such that it contains nothing but modules with only one component. This means that only components are considered at the start of the process. Modules from x are added/removed to improve the solution, as follows: With an initial value x, $C(x)$ (Equation (4)) is evaluated, which is the initial temperature. For every constraint, Equations (1)–(3), that is not respected, a penalty is added. $Best(x) \leftarrow C(x)$, $x^* \leftarrow x$, $Level \leftarrow 0$ and $Iteration \leftarrow 0$.

(3) A neighbourhood of x is constructed, and two alternatives are considered:

- If x does not permit the manufacture of all products, respecting all constraints, a module is added to x, and we obtain x'.
- If x permits the manufacture of all products, a random process decides (with equal probability) whether to add or remove a module from x, and we obtain x'.

The module to be added or removed is randomly selected, the random process being weighted with the use case number of each module. Modules with a large (small) weight are more likely to be selected to be added (removed).

(4) $C(x')$ is evaluated in a similar way to that in Step 2. $Iteration \leftarrow Iteration + 1$.

(5) If $C(x') \leq C(x)$ (with respective penalties), then the neighbour is accepted: $x \leftarrow x'$ and $Level \leftarrow 0$, otherwise go to Step 7; if $C(x') \leq Best(x)$, then the best solution is recorded: $Best(x) \leftarrow C(x')$ and $x^* \leftarrow x'$.

(6) A random number α is compared to $p(Level)$, if $\alpha \leq p(Level)$, $x \leftarrow x'$, otherwise x' is rejected.

(7) If $Level \geq Max\_Level$, $p(Level)$ is updated; if $Iteration \geq Max\_Iteration$, the optimisation process is stopped.

(8) All modules in x^* that do not appear in any product $P(k)$ are removed, $C(x^*)$ is updated. x^*, $C(x^*)$ is the list of non-feasible products, and the evaluations of all Pk are given.

4. Case study

This case study is structured around the modular design of headlamp devices. Today, this device is used in applications ranging from extreme situations, like caving, mountaineering, military operations, and professional usage (such as in a gaseous environment) to casual situations, like outdoor and do-it-yourself activities. Cavers, miners, and rescue personnel require a robust, light, reliable, and water-resistant headlamp. In many other cases, users require a simpler, less costly product. The power of light is expressed in lumen. Low cost headlamps typically produce less than 10 lumens in output, and are priced below \$10. Mid-priced lamps cost between \$10 and \$150. These lamps produce from 50 to 150 lumens. High performance headlamps produce between 350 and 1200 lumens for a price varying between \$400 and \$800.

Without providing all the details of the parts and specifications of modern headlamps, we present some of the major characteristics of their core technology. The energy used for the headlamps used in mining, caving, and rescue operations has shifted almost completely from acetylene gas (carbide lamp) to electricity (battery-powered lamp). Modern headlamps use LEDs instead of classical light bulbs. The mechanical parts required to host LEDs and batteries are mostly made of aluminium or plastic. These parts must form a tight, cool container, which is lightweight and strong. They often require lithium-ion (Li-ion) batteries, which can neither be shocked nor be wet, because of the risk of fire or explosion, and must not be exposed to moisture during the manufacturing process. A current and voltage regulator is placed between the batteries and the LEDs to allow light emission.

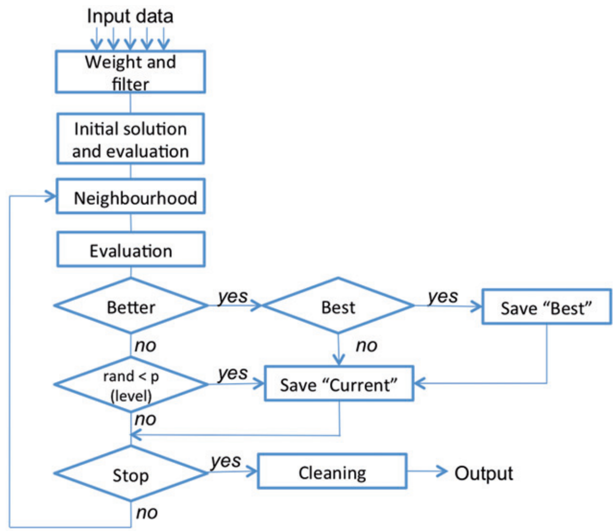

Figure 6. General scheme of the algorithm.

This microcontroller furnishes several grades of power and light, and switches and cables connect the modules. An example is illustrated in Figure 7.

The device presented in Figure 7 produces a maximum of 700 lumens to keep the cost of the components low (under \$70). It weighs less than 250 g. It has 10 functional parts, among them a lamp, a battery pack, a microcontroller, and a switch. It is made up of 56 components, which are presented in detail in Table 7. However, the reliability test, especially for the microcontroller device, has not been provided and thus cannot be warranted for the lamp. Many options can be accommodated on this device. A short list of some of these options is given in Table 8, which generates over 10 billion possible configurations.

Case study: eight options, 15 functions, seven constraints, 11 products, one supplier per function, one quality rating per function

A subset of eight options is tested first. These options, which are listed in Table 1, generate 15 functions for the product.

Every component performs a specific function, and is defined with a cost and a quality rating (evaluated based on its failure rate).

From these components, it is possible to put together preassemblies (called modules). The failure rate and the cost of a module both depend on the components it contains. The following has been adapted for computation purposes:

Failure rate

The failure rate of a module or is the sum of the failure rate of the components it contains minus d. It is a positive value.

$$\rho(M_j) = \mathrm{Max}\left(\sum_{i \in Mj} \rho(C_i) - d; 0\right) \tag{5a}$$

The failure rate of a product is the sum of the failure rate of the modules it contains minus d. It is a positive value.

$$\rho(P_k) = \mathrm{Max}\left(\sum_{i \in Pk} \rho(M_i) - d; 0\right) \tag{5b}$$

It is assumed by this choice that chosen modules act as key elements of the product. By the way the failure of one of them generates a failure of the product and impacts its quality. This assumption is acceptable for core parts

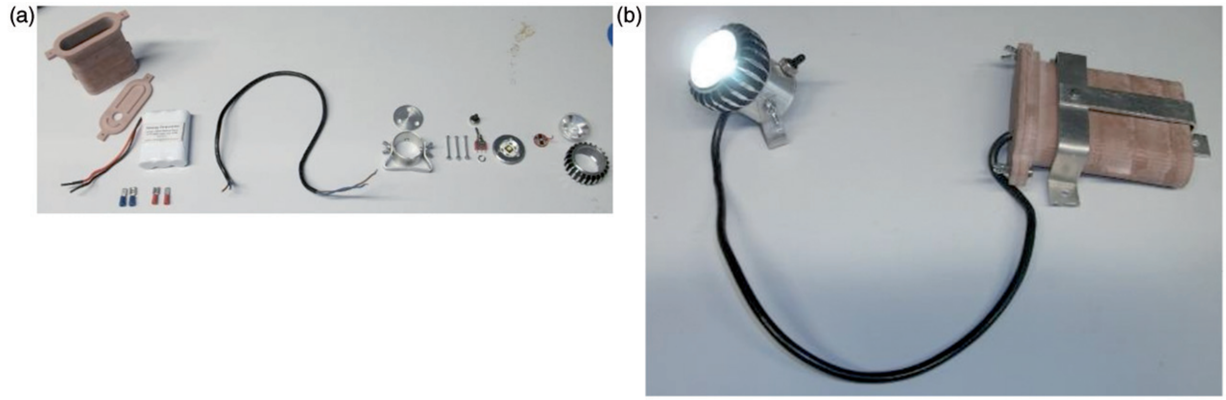

Figure 7. Illustration of a headlamp.

Table 1. Details of the 15 functions selected for the test.

Function number	Name	Detail
F1	LED (mounted on a star pcb, with 2 connections per pole)	Warm
F2		White
F3	Batteries with PCB voltage, driver and charger	PCB, 3,7V, 5Levels Driver (on, off, low, middle, High)
F4		PCB, 3,7V 4Levels Driver (on, off, Low, High)
F5	Switcher	2 positions and water-proof
F6		2 positions and strong water proof (100 m and gas resistant)
F7	Battery case	Case for helmets with small cables and water-proof
F8		Case to be carried-on manually with long cables and water-proof
F9	Battery case Fixtures for helmets	For camp
F10		For Petzl écrin-roc
F11		For Petzl Elios
F12	Battery output charger	
F13	Battery water-proof reinforcement	
F14	Colour	Blue marine
F15		Militarian

of a product. Even if inside such core components redundancy is organised, overall the component will be perceived as an entity with improved quality characteristics. The action on quality is modelled by the quantity d. It is also assumed that d will not change over time. This assumption is a strong limitation, as every enterprise has continuous improvement programs. Nevertheless we decide to keep this variable as constant to manage a tractable model.

- If $d < 0$, the module is of poorer overall quality than the components it contains. The assembly operation increases the risk of failure.
- If $d > 0$, a sort operation is performed after the module has been assembled. The failure rate of the module is reduced.

Cost

To calculate cost, we suppose that the cost of a module depends on that of its components.

$$Cost(M_j) = (1 - a)\left(\sum_{i \in Mj} Cost(C_i)\right) \tag{6}$$

- If $a > 0$, the module is less expensive than the sum of its components (this could change, if the contractor profits from the effect of volume sales),
- If $a < 0$, the module is more expensive than the sum of its components.

So, the cost and failure rate of a finished product are directly linked to the modules and the components selected for its manufacture.

Technical constraints of this scenario

Not all combinations of components result in a technically or commercially feasible product. For the purposes of the study, the following constraints are observed:

F1 or F2: contain only one type of LED	If F8, then not F9, F10, F11: if the batteries are carried manually, then there are no helmet straps
F3 or F4: a four-level or a five-level PCB device	F9, F10, or F11: helmet straps depend on the helmet
F5 or F6: switch is reinforced or non-reinforced	F14 or F15: one of two colours available in each product
F7 or F8: battery cases made for helmets or to be carried manually	

F1 or F2 means that a feasible product must contain either F1 or F2, but not both. Also, if a final product contains F8, it will not contain F9, F10, or F11, and so on.

This set of functions and the related constraints return a possible 2930 different modules.

This leads to the following questions: for a given level of lamp reliability, how can a set of modules be selected that ensures quality at the lowest possible price? How can a set of modules be selected that achieves the lowest cost for each product, while maximising product reliability?

Our study proposes to select the set of modules that will permit the manufacture of the following set of final products. The full description of each module is provided in Table 9.

Table 2 contains different models of headlamps for manufacture. For example, P1 is a lamp for cavers. It must be reliable (an expected failure rate of 15.10^{-6}), and we would like to provide it at the lowest possible cost. This lamp contains functions F2, F3, F5, F7, F10, and F12. Based on a simple assembly of raw components, the resulting product, P1, will have failure rate of 15.10^{-6}, and a final cost of \$131.50.

In order to test the algorithm, we decided to consider a function (F11) that is not necessary in any product. Table 3 presents the quality (in number of failures per 0.106 products) and cost (in \$) expected for each product to be manufactured, as well as the quantities of products to manufacture (in thousands). For instance, for product P1, the constraint is to obtain an overall failure rate lower than 16.10^{-6} at the lowest possible cost. The quantity to be produced is 50. Comparing these numbers with those in the last two lines of Table 2: for product P1, if each component is assembled individually, the final cost will be \$131.50; however, the overall quality will not meet the failure rate requirement ($17 > 16$). In some cases, raw material assembly is an acceptable solution for the market, but the product could still be improved upon from a cost (or quality) perspective. For instance, for product P11, the required level for the failure rate is 20.10^{-6}, and, with raw material assembly, it is possible to achieve 17.10^{-6}. In Table 3 the products that can be made from raw material assembly and meet the market demand are identified in italics (seven products cannot).

In our case study, we noted the quantities produced by a lamp manufacturer: 790 products were ordered and split into 11 product types. The results from this case study are presented in the following section.

5. Results and discussion

5.1 *Computational evolution*

We consider here the above-defined problem. Modules are manufactured and assembled under cheaper conditions than raw materials, and a sorting operation makes it possible to discard a few of the problematic modules. For computational purposes, $a = 0.05$ and $d = 1$ for Equations (5) and (6).

As explained previously, G (the management cost of a module) has a major impact on the number of modules. In the following, $G = 300$. The penalty cost for each non feasible product is \$1000.

For the simulated annealing procedure, the parameters are the following (values obtained after several tests were performed):

- Max_Iteration = 500
- Number of levels = 3, with a $p(Level)$ of 0.4, 0.2, and 0.1 respectively; also $Max\_Level = 100$ iterations.

Figure 8 shows the number of modules (current solution) and the number of non-feasible products (current and best solutions).

Table 2. Functions and products.

Function	Cost (in $)	Failure rate (.10-6)	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
F1	23	1	0	1	0	0	0	0	1	1	0	0	0
F2	30	1	1	0	1	1	1	1	0	0	1	1	1
F3	60	1	1	1	1	1	1	1	0	0	1	1	1
F4	27	30	0	0	0	0	0	0	1	1	0	0	0
F5	3	1	1	1	1	1	0	0	1	0	0	0	0
F6	4	2	0	0	0	0	1	1	0	1	1	1	1
F7	35	1	1	1	1	1	1	0	0	0	1	1	1
F8	30	20	0	0	0	0	0	1	1	1	0	0	0
F9	4	3	0	0	0	1	0	0	0	0	0	0	1
F10	1	10	1	1	1	0	1	0	0	0	1	1	0
F11	4	5	0	0	0	0	0	0	0	0	0	0	0
F12	2.5	3	1	1	1	1	1	1	0	0	1	1	1
F13	20	5	0	0	0	0	1	1	0	1	1	1	1
F14	2	1	0	0	0	0	0	0	0	1	1	1	1
F15	3	1	0	0	0	0	1	1	0	0	0	0	0
Resulting of failure rates with the simple assembly of row components			17	17	17	10	24	33	52	59	24	24	17
Resulting of costs (in $) with the simple assembly of row components			131.5	124.5	131.5	134.5	155.5	149.5	83	106	154.5	154.5	157.5

Table 3. Constraints, objectives, and quantity per product.

Product	P1	P2	*P3*	P4	P5	P6	P7	*P8*	*P9*	P10	*P11*
Target cost per product	×	120	×	×	×	×	80	×	**160**	150	×
Target failure rate per product	16	×	***20***	9	20	30	×	**60**	×	×	**20**
Quantity per product	50	50	50	50	100	50	70	70	100	100	100

Note: Product in bold values are feasible with raw component assembly.

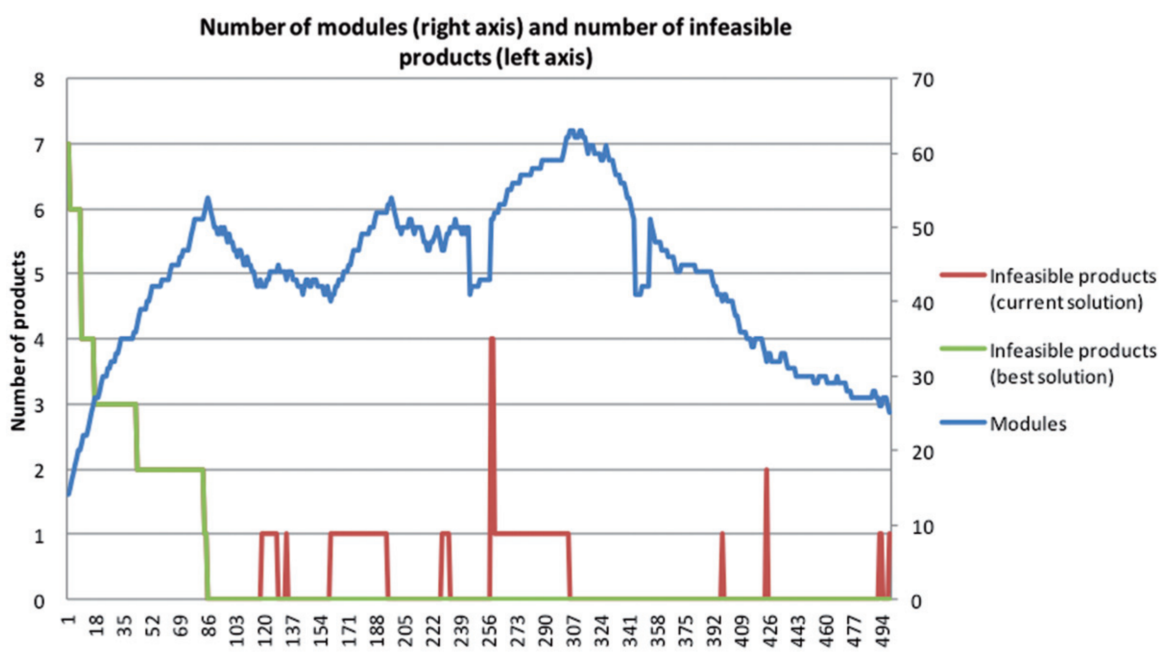

Figure 8. Number of modules (left axis) and non-feasible products (right axis).

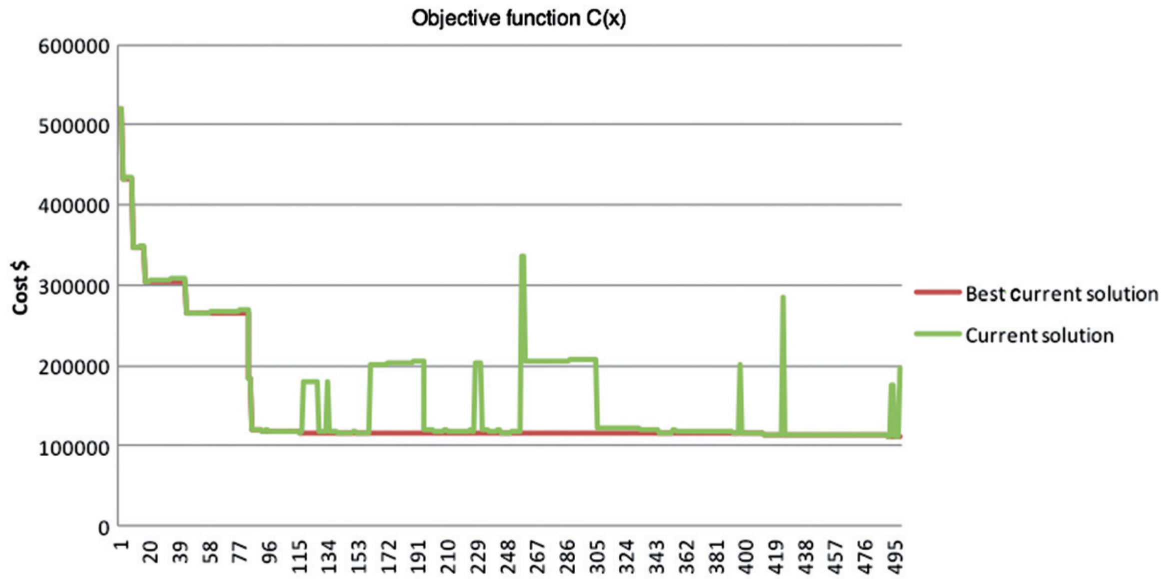

Figure 9. Objective function.

Figure 8 shows that, starting with these seven non-feasible products, modules are added to the current solution until all the products are feasible. This point is reached after 85 iterations. The algorithm seeks to improve the objective function by removing modules until some products become non-feasible, adding to modules for feasibility and removing them for improvement. By the end of the process, 28 modules had been selected.

Figure 9 shows the current and best solution for $C(x)$. In this figure, we see a typical evolution of a simulated annealing algorithm with stepped improvements and penalties.

The final solution is five times cheaper that the solution that requires raw material assembly.

The 28 modules selected make it possible to produce all the required products (respecting both constraints), and it is the best solution found up to now in terms of cost $C(x)$. Applying the cleaning procedure (Step 8 in Subsection 3.3), some modules are removed from the best solution and 18 are proved to be sufficient to solve the problem.

The final solution is described in the following tables.

Table 4 shows the solution made up of 18 modules (M1 to M18), obtained from the assembly of several functions. For instance, Module 5 has a cost of \$61.50 and a failure rate of 1.10^{-6}. It is a package made up of F_2 and F_7. Operationally, this package contains a white LED and a helmet case with small cables and a waterproof reinforcement kit. M12 is made up of five raw components. With these modules, the 11 products become feasible.

As expected, F11 does not appear in any module.

The use case of every module is presented Table 5. For instance, Product 4 contains M2, M3, and M10.

It also appears from Table 5 that M2, M9, and M12 are used in three or more products. Major modules are used in one or two products.

Table 6 displays the final characteristics of each product.

For instance, product P1, ordered in a quantity of 50,000, will have a cost of \$126.425 and a failure rate of 16.10^{-6}. Note that the raw assembly solution (noted in the remainder, C^0) can be made up of two parts (quality and cost): for example, 17.10^{-6} and \$131.50. This method retrieves a better solution for each parameter (quality and cost). Overall, the algorithm outperforms C^0. When required, improvements can always be made. While not requested, secondary objectives are also systematically improved upon.

These results are very encouraging, as they constitute the initial solution for an industrial team wishing to reduce the discrepancy between marketing needs and manufacturing system abilities. This gap is filled by joint action on the modules, that is, action in terms of quality and costs. With their partners' quality management and cost management skills, this team can try to achieve better perfomance in terms of market coverage.

This example proposes modular design as a solution to cope jointly with quality and cost constraints. Instead of performing a raw component assembly, the modules have to be defined. During preassemby operations, a quality

Table 4. Module composition matrix.

	Cost	Failure rate	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15
M1	23	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	30	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	4	3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
M4	20	5	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
M5	61.75	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
M6	65.55	3	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0
M7	57	50	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0
M8	32.3	21	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
M9	59.375	3	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0
M10	95.475	5	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0
M11	3.325	12	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0
M12	96.425	15	0	0	1	0	0	0	1	0	1	1	0	1	0	0	0
M13	22.8	6	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
M14	49.4	31	1	0	0	1	1	0	0	1	0	0	0	0	0	1	0
M15	21.85	15	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0
M16	59.85	1	0	0	1	0	0	0	3	0	0	0	0	0	0	0	0
M17	59.85	21	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1
M18	9.5	5	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0

Table 5. Product x module matrix.

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18
P1		1										1						
P2	1											1						
P3		1										1						
P4		1	1							1								
P5					1						1		1			1		
P6									1				1				1	
P7	1						1											
P8				1				1										
P9						1			1					1				
P10						1			1						1			
P11				1	1				1						1			1

Table 6. Results.

		Raw results		After the algo	
	Quantity (10ˆ3)	Quality	Cost	Quality	Cost
P1	50	17	131.5	*16**	126.425
P2	50	17	124.5	16	*120**
P3	50	17	131.5	*20*	126.425
P4	50	10	134.4	*9**	129.475
P5	100	24	155.5	*20**	147.725
P6	50	33	149.5	*30**	142.025
P7	70	52	83	51	*80**
P8	70	59	106	*60*	101.7
P9	100	24	154.5	21	*160*
P10	100	24	154.5	21	*150**
P11	100	17	157.5	*20*	150.625
Legend:					
Requirement					
OBTAINED					
Improved					
Non feasible with raw component assembly:*					

Table 7. Device description.

Major parts	Raw components	Quantity of row components
Lamp-mounting device	Core	1
	Lens	1
	Back	1
	Front	1
Driver	Microcontrolller, capacity, résistance, platine.	13
Water-Proof Switch	Switch	1
	Lock washer	2
	Toric seal	1
	The water-resistant end	1
Lamp-circuit case fixture	Screw Æ 3	3
	Nuts Æ 3	3
Cables	Cable	2
	Master key cable	1
Circuit case	Case body	1
	Base back	1
	Butterfly nuts Æ 3	2
	Screw Æ 3	2
	Aluminum folded support	1
Battery case	Lid	1
	Pass-cable	1
	Case	1
	Screw Æ 3	2
	Butterfly nuts Æ 3	2
Battery	Li-Ion 3,7V, 6600mA, with protection device	1
	Charger	1
Junctions	Junction	4
Lamp	LED – CREE MCE 430Lumens@350mA	1
Fixtures & sealings	Thermal fix	1
	Silicon joint	1
	Adhesive tube	1
	Welding device	1
Total number of components		56

assessment can be carried out, so that the modular design, combined with quality and cost control, becomes part of the continuous improvement cycle of the manufacturing system.

The example proves that it is possible to address a particular market by simultaneously considering modularity, quality, and cost control. In order to appreciate the extensive improvements achieved by this algorithm, Table 3.

The assembly of raw components does not meet the marketing requirements for P1, P2, P4, P5, P6, P7, and P10 (see Table 6).

For P1, assembly of the raw components achieves a quality of 17.10^{-6}, while the required quality is 16.10^{-6}. The associated cost is \$131.50. The modular design helps, as the constraint had been set to 16 and the cost improved to \$126.425. In fact, the quality constraint is fulfilled, and the \$5.075 saved for each product can be shared among customers, manufacturers, and suppliers.

For P2, raw component assembly achieves a cost of \$124.50. However, the modular design, as conceived in this paper, makes it possible to limit the cost to \$120, and also to reach a better quality solution, of 16.10^{-6}. Incidentally, the modular design reinforces the overall quality performance of the brand. From a manufacturer's and a marketer's point of view, the quality provided cannot be considered excessive, as its level remains reasonable.

The requirements of P4, P5, P6, P7, and P10 are not met with raw component assembly, based on the following figures: a quality constraint of 9.10^{-6}, 20.10^{-6}, and 30.10^{-6} for P4, P5, and P6 respectively, and a cost constraint of \$80 and \$150 for P7 and P10 respectively. The modular design, with its effect on cost and quality, allows these constraints to be respected, although modestly, and also improves the pending parameters, compared to raw component assembly.

Table 8. Options for each module.

Module	Options	Number of variants
LED colour	White/Warm	2
Let platine	Star/Round	2
LED Connection per pole choice	A 2 connections (/4 connections	2
Batteries protection	PCB/No PCB	2
Batteries water-proofed	Yes/No	2
Batteries Volatge (constraint)	3,7v/11,6v	2
Driver (the type depend of the voltage of the batteries)	5 levels, 4 levels, 3 levels, 2 levels	4
Cable	Small/middle/long	3
Batteries case	(Helmet (10 models) or hand (3 models)) * water proof (30 m)/ not water-proof * (ability to charge or not directly from the case)	52
Case colour	16 Colours (including balck and white)	16
Lamp carrier	Water-proof (30 m)/Not water proof * Colour (16 possibilities)	32
Driver & switcher container	Colour (16 possibilities)	16
Switcher	2, 3, 4, 5 positions	4
Quality level	Non checked/warrant	2
Over-all sealing requirements	Non water-proof/water-proof/Diving (100 m)/Gas Environement	4
Total of variants		10 468 982 784

Table 9. Engineering rules.

P1: Lamp for cavers, reliable (rate of failure at 16.10-6), at the lowest possible cost. Lamp with white output, 5 levels of output, batteries mounted on the ‘‘Écrin-roc’’ helmet, with a battery charger, without water-proof reinforcement, not coloured. (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 1,f6 = 0, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 0, f12 = 1, f13 = 0, f14 = 0, f15 = 0)

P2: Lamp for cavers, reliable (at the lowest failure rate), at the cost $120. Lamp with warm output, batteries mounted on ‘‘Écrin-roc’’ helmet, with a battery charger, with no waterproof reinforcement, nor colour. (f1 = 1, f2 = 0, f3 = 1, f4 = 0, f5 = 1,f6 = 0, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 0, f12 = 1, f13 = 0, f14 = 0, f15 = 0)

P3: Lamp for cavers, reliable (rate of failure at 20.10-6), at the lowest possible cost. P1 but for ‘‘Elios’’ helmets, (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 1,f6 = 0, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 1, f12 = 0, f13 = 0, f14 = 0, f15 = 0)

P4: Lamp for cavers, very reliable (failure rate 9.10-6 at the lowest possible cost. P1 but for the ‘‘Camp’’ helmet, (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 1,f6 = 0, f7 = 1, f8 = 0, f9 = 1, f10 = 0, f11 = 0, f12 = 1, f13 = 0, f14 = 0, f15 = 0)

P5: P1, Military lamp, (rate of failure at 20.10-6), at the lowest possible cost, lamp with white output, batteries mounted on the helmet ‘‘écrin-roc’’, with a battery charger, with wateproof reinforcement on switch and batteries, coloured in‘‘Militarian’’ camouflage, (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 0, f6 = 1, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 0, f12 = 1, f13 = 1, f14 = 0, f15 = 1)

P6: P5 with a rate of failure = 30.10-6, but batteries are hand-held, (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 0, f6 = 1, f7 = 0, f8 = 1, f9 = 0, f10 = 0, f11 = 0, f12 = 1, f13 = 1, f14 = 0, f15 = 1)

P7: Outdoor activities, best quality available for 80$, warm white lightening, 4 levels of lightening, no waterproof reinforcement, batteries hand-held, no output charger, not colour. (to keep cost down), (f1 = 1, f2 = 0, f3 = 0, f4 = 1, f5 = 1, f6 = 0, f7 = 0, f8 = 1, f9 = 0, f10 = 0, f11 = 0, f12 = 0, f13 = 0, f14 = 0, f15 = 0)

P8: P7, Outdoor activities, lowest cost, medium quality of 60,10-6, with waterproof reinforcement and blue (to suggest itswater resistance), (f1 = 1, f2 = 0, f3 = 0, f4 = 1, f5 = 0, f6 = 1, f7 = 0, f8 = 1, f9 = 0, f10 = 0, f11 = 0, f12 = 0, f13 = 1, f14 = 1, f15 = 0)

P9: P1 Lamp for cavers, most reliable possible at the cost of $160 with waterproof reinforcement (batteries and switch) and blue coloured in blue (to suggest its water resistance) (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 0,f6 = 1, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 0, f12 = 1, f13 = 1, f14 = 1, f15 = 0)

P10: P3 Lamp for cavers, most reliable possble, at a cost of $150, with waterproof reinforcement (batteries and switch) and colour in blue (to suggest its water resistance) (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 0,f6 = 1, f7 = 1, f8 = 0, f9 = 0, f10 = 1, f11 = 1, f12 = 0, f13 = 1, f14 = 1, f15 = 0)

P11: P4 Lamp for cavers, reliable (rate of failure at 20.10-6), with waterproof reinforcement (batteries and switchr) and colour in blue (to suggest its water resistance) (f1 = 0, f2 = 1, f3 = 1, f4 = 0, f5 = 0,f6 = 1, f7 = 1, f8 = 0, f9 = 1, f10 = 0, f11 = 0, f12 = 1, f13 = 1, f14 = 1, f15 = 0)

Others products (P3, P8, P9, and P11) are feasible with raw component assembly, although modular assembly obtains better results on the non-constrained parameter. For instance, for P11, raw component assembly achieves a quality of 17.10^{-6} and a cost of \$157.50. Thus, this product exceeds the quality constraint, which is set to 20.10^{-6}, and its cost is improved, reaching \$150.625. Again, these savings can be shared among customers, suppliers, and manufacturers.

Overall, this algorithm improves market coverage by ~177%: from ~36.4% to 100%. Although the example does not prove conclusively that this process is the definitive solution to addressing cost and quality issues, it does open the way to optimising these issues in a multisectorial application of a product.

5.2 *Experimental plan to simulate several supply chain configurations*

In this section, we evaluate the influence of a (cost) and d (quality) on the feasibility of the required products.

For computation purposes, the parameters are fixed as follows:

- $Max\_Level = 200$
- $P(Level) = [0.4, 0.2, 0.1]$
- $Max\_Iteration = 1000$
- $G = 300$ (management cost of each module)
- $P = 1000$ (penalty for each non feasible product)
- a (Equation 1) varies from -0.1 to 0.1, with steps of 0.02
- d (Equation 2) varies from -4 to 4, with steps of 2

Behind the ability to design a product family with modules is the ability of suppliers to sort and reduce costs every time a module is created. Variations on a and d simulate the different scenarios that increase or decrease cost and quality. Every scenario is solved three times, and the mean results are provided in this section.

In order to retrieve useful results from the experiments, we have decided to synthesise them in the following way: the average results of positive, negative, and null values of a and d.

When a is positive, the overall cost is below the sum of the cost of a module's components every time it is used. A positive a means cost improvement. In contrast, a negative a means cost reduction. When d is positive, a sort operation is performed, and when d is negative, module creation adds defects. When a or d are null, the influence of the remaining non-null parameters can be seen on the number of modules and the number of non-feasible products.

The number of non-feasible products is noted *#inf_prod*, and the number of modules is noted *#modules*. Results are presented in Figure 10.

Influence of d:

When d evolves from a negative to positive value, there is a decrease in the number of non-feasible products, whatever the value of a. When $a = 0$ (central column), the relative effect of d can be seen when the cost does not vary from raw components to modules, and the decrease in the number of non-feasible products with the variation in d is $(3\text{–}5.5)/3 \approx -83.33\%$. When $a > 0$, the effect of $d > 0$ compared to that of $d < 0$ on the variation in the number of non-feasible products is stronger by $(0.64\text{–}2.7)/0.64 \approx 321\%$. When $a < 0$, the effect of $d > 0$ and that of $d < 0$ on the variation in the number of non feasible products are the same as when $a = 0$, the influence equals $(3\text{–}5.5)/3 \approx -83.33\%$.

When d evolves from a negative value to a positive value, this has an impact on the number of modules necessary to generate the solution. For $a = 0$, this number drops from 19.15 to 15.8. For $a > 0$, the variation is greater, the number increasing from 15.67 to 22.17. For $a < 0$, the reduction, although not as large (from 16.74 to 17.68), is still notable.

So, the relative level of quality of the modules, compared to that of the raw components, has a major influence, both on the number of feasible products and on the number of modules necessary to satisfy the constraints of all the products.

Influence of a:

When a evolves from a negative to positive value, this has a strong influence on the number of non-feasible products and on the number of modules. When $d = 0$, an increase in a generates a reduction in the number of non-feasible products, from 6.2 to 1.72. This represents a drop of $(1.72\text{–}6)/1.72 \approx -248\%$, a figure which has to be

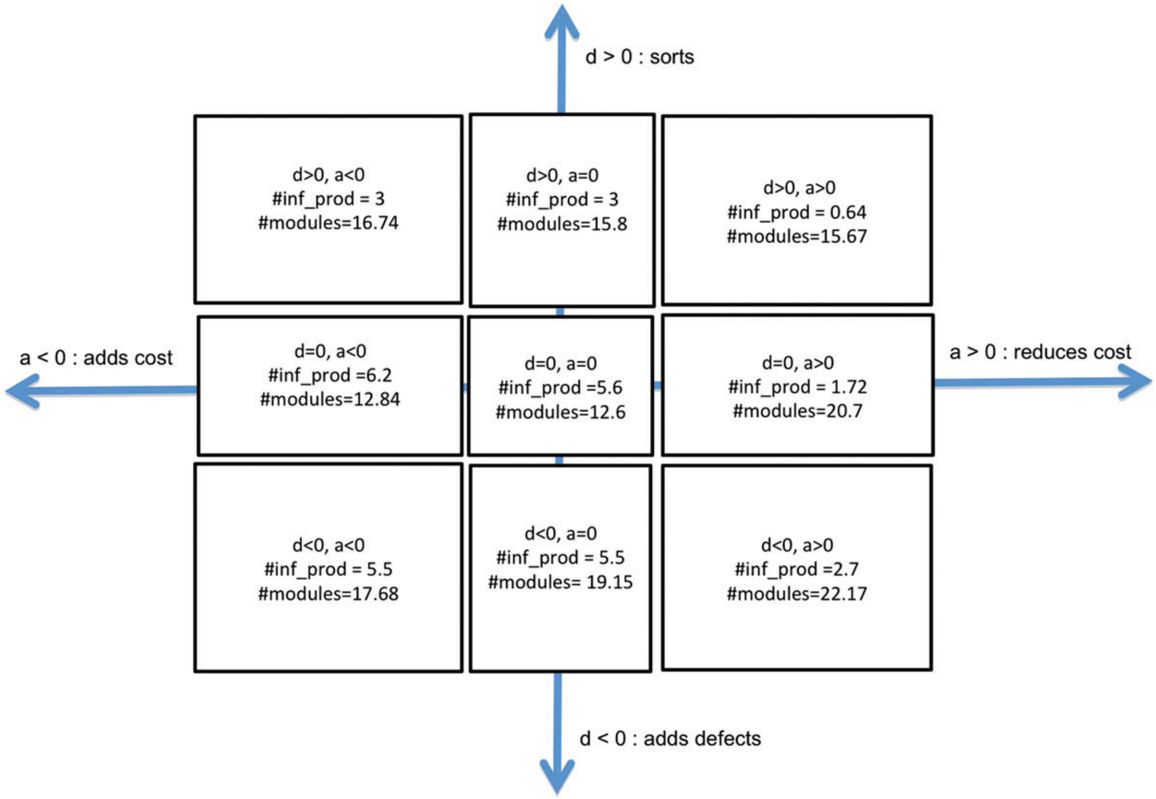

Figure 10. Synthetic result of the experiment design.

compared with the -83.3% drop in the influence of d under the same conditions. The influence of a, all other parameters being equal, is then three times stronger than that of d. When $d > 0$, the number of non-feasible products decreases from 3 to 0.64 when a increases. When $d < 0$, the number of non-feasible products continues to decrease, from 5.5 to 2.7, when a increases.

The influence of a on the number of non-feasible products depends on d. When d is positive, it influences the performances of a by $(368–103)/103 \approx 257\%$, in terms of the reduction in the number of non-feasible products.

When a evolves from a negative to positive value, this has a strong influence on the number of modules. When $d = 0$, the increase in a increases the number of modules by $(20.7 - 12.84)/20.7 \approx 37.97\%$. When d is positive, the increase in a has less impact on the number of modules generated, and that number is reduced, on average, by $(15.67 - 16.74)/15.67 \approx -6.8\%$. In contrast, when $d < 0$, the increase in a increases the number of modules at a rate of $(22.17 - 17.68)/22.17 = 20.25\%$.

Here a has a direct impact on the cost of the final product, and so as a decreases, the cost of the modules naturally leads to an increase in the number of modules.

When $d < 0$ or $d = 0$, the way to improve the objective function is to add modules, as costs decrease with an increasing number of modules, although it must be remembered that this strategy is limited by G (management cost of a module).

The case where $d > 0$ and $a > 0$ is the best scenario of all. Every time a module is used, there is a sort operation and also a cost reduction. The number of non-feasible products is the lowest (0.64), and the number of modules is reasonable: not 100 and not two, but 15.67, on average.

The case where $d < 0$ and $a < 0$ is the worst possible scenario from an industrial perspective. Every time a module is used, defects are added and the cost increases. The number of non-feasible products does not decrease (because of quality constraints), while at the same time the search operation generates a large number of modules.

This experiment design explores the performances of module providers. The ideal case that managers are looking for is when suppliers are able to improve quality ($d > 0$) and at the same time reduce their costs when they use a module ($a > 0$). With this scenario, the number of non-feasible products decreases dramatically, while the number of intermediate modules remains manageable. This is a win–win situation.

From this experiment, it can be seen that the results remain robust on variations of supplier performance (low $d > 0$ and/or low $a > 0$). This helps manufacturers deliver promising results during ramp-up, even though the supply chain has just been set.

This research will help manufacturers negotiate global module management terms with their suppliers: reasonable volumes, with better quality at lower cost for each part. From the supplier's point of view, this can be the time to upgrade to a global contract with manufacturers and an opportunity to enter into mutually agreed continuous improvement actions.

6. Conclusion

This paper is about modular design. It takes into account the actions taken on cost and quality every time a module is used, which enables the production of a particular product family, each product of which is constrained by limits on one of these two parameters.

The paper presents a model designed to minimise the total production cost. A real case study and an experimental design have been examined in order to analyse the behaviour of this model in depth. The test stems from the authors' experience in manufacturing headlamp devices for environments which are constrained in terms of cost and quality).

The model makes it possible to select a set of modules that allows the constraints of each product to be satisfied, while minimising the total production cost for the product family. The model is useful for practical applications, since it yields the set of modules and its composition, as well as the bill of materials for each product.

More extensive tests yield a comparative analysis of the influence of cost and quality parameters, revealing that these two parameters affect the number of non-feasible products. A cost savings ($a > 0$) on every module produced reinforces the effect of quality improvement. The same applies to quality. Every time a module of improved quality ($d > 0$) is produced, the effect of cost savings is reinforced (in terms of the number of non-feasible products that become feasible). This is the most notable result.

Operationally, this result could be used directly in the following method:

(1) The marketing department issues a document for a product family specifying the cost and quality constraints over each declination of the product.
(2) The purchasing department has to find a supplier who can decrease the cost of each module, and add quality to it.
(3) The manufacturing department must identify the relevant modules, and iterate with the purchasing department in order to reach to an acceptable solution.

This research opens up opportunities for further study. The first concerns the influence of quantity on the stability of the module. In the test, several hundred lamp sets were ordered. In a manufacturing environment, there is always a ramp-up period to fine-tune the manufacturing system and to enable learning. A major characteristic of this period is volume increases. The overall volume can increase (for all products), but can also evolve in a differentiated manner (from one product to another). This can heavily impact the choice of modules and their stability, and is a research avenue that needs to be investigated.

The second concerns the introduction of a list of suppliers, along with their relative performances. Taking this information into account could lead to an optimum manufacturing solution, or to a robust one, which might be more costly but more resilient to disruptions.

The third concerns the effect of learning on quality and module stability. This effect occurs with a volume increase on a particular product. It is another effect that is characteristic of the ramp-up period.

Finally, a one-level module design has been proposed here, and a multi-level modelling should be considered as well.

References

Agard, B., da Cunha, C., and Cheung, B., 2009. Composition of module stock for final assembly using an enhanced genetic algorithm. *International Journal of Production Research*, 47 (20), 5829–5842.

Agard, B. and Kusiak, A., 2004a. Data-mining-based methodology for the design of product families. *International Journal of Production Research*, 42 (15), 2955–2969.

Agard, B. and Kusiak, A., 2004b. Data mining for subassembly selection. *Journal of Manufacturing Science and Engineering*, 126 (3), 627–631.

Agard, B. and Penz, B., 2009. A simulated annealing method based on a clustering approach to determine bills of materials for a large product family. *International Journal of Production Economics*, 117 (2), 389–401.

Baud-Lavigne, B., Bassetto, S., and Penz, B., 2010. A broader view of the economic design of the X-bar chart in the semiconductor industry. *International Journal of Production Research*, 48 (19), 5843–5857.

Bertsche, B., 2008. Reliability in automotive and mechanical engineering. Series: VDI Buch. Berlin: Springer-Verlag, 98–159.

Carmignani, G., 2009. An integrated structural framework to cost-based FMECA: the priority-cost FMECA. *Reliability Engineering & System Safety*, 94 (4), 861–871.

Cheung, K.L. and Leung, K.F., 2000. Coordinating replenishments in a supply chain with quality control considerations. *Production Planning & Control*, 11 (7), 697–705.

Colledani, M. and Tolio, T., 2011. Integrated analysis of quality and production logistics performance in manufacturing lines. *International Journal of Production Research*, 48 (2), 485–518.

da Cunha, C., Agard, B., and Kusiak, A., 2007. Design for cost: Module-based mass Customization. *IEEE Transactions on Automation Science and Engineering*, 4 (3), 350–359.

da Cunha, C., Agard, B., and Kusiak, A., 2010. Selection of modules for mass customisation. *International Journal of Production Research*, 48 (5), 1439–1454.

Das, K. and Chowdhury, A.H., 2012. Designing a reverse logistics network for optimal collection, recovery and quality-based product-mix planning. *International Journal of Production Economics*, 135 (1), 209–221.

Department of defense, 1980. Procedures for performing a failure mode, effects, and criticality analysis, MILSTD 1629-A.

Ebrahimipour, V., Rezaie, K., and Shokravi, S., 2010. An ontology approach to support FMEA studies. *Expert Systems with Applications*, 37 (1), 671–677.

Eubanks, C. and Ishii, K., 1997. Advances FMEA using behavior modeling. *In*: *Proceedings of the 1997 ASME design engineering technical conferences and design theory and methodology conference*, 14–17 September 1997, Sacramento, California.

Fujita, K., 2002. Product variety optimization under modular architecture. *Computer-Aided Design*, 34 (12), 953–965.

Garey, M.R. and Johnson, D.S., 1979. *Computers and intractability: a guide to the theory of NP-completeness*. 1st ed. New York: W.H. Freeman & Co.

Ghemraoui, R., Mathieu, L., and Tricot, N., 2009. Design method for systematic safety integration. *CIRP Annals – Manufacturing Technology*, 58 (1), 161–164.

Hassan, A., *et al*., 2010. Conceptual process planning – An improvement approach using QFD, FMEA and ABC methods. *Robotics and Computer-Integrated Manufacturing*, 26 (4), 392–401.

Hollnagel, E., 2008. Risk + barriers = safety? *Safety Science*, 46 (2), 221–229.

Jiao, J., Simpson, T.W., and Siddique, Z., 2007. Product family design and platform-based product development: a state-of-the-art review. *Journal of Intelligent Manufacturing*, 18 (1), 5–29.

Kmenta, S., Fitch, P., and Ishii, K., 1999. Advanced failure modes and effects analysis of complex processes. *In*: *Proceedings of the 1999 ASME design engineering technical conferences*, 12–15 September 1999, Las Vegas, Nevada.

Kusiak, A., 1999. *Engineering design: Products, processes, and systems*. 1st ed. San Diego, California: Academic Press, Inc.

Kusiak, A. and Huang, C. C., 1996. Development of modular products, *IEEE Trans. Compon., Packag., Manuf. Technol. A*, 19 (4), 523–538.

Larson, N. and Kusiak, A., 1996. Managing design processes: A risk assessment approach. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 26 (6), 749–759.

Lee, H.L. and Tang, C.S., 1997. Modeling the costs and benefits of delayed product differentiation. *Management Science*, 43 (1), 40–53.

Li, S. and Lin, B., 2006. Accessing information sharing and information quality in supply chain management. *Decision Support Systems*, 42 (3), 1641–1656.

Lough, K.G., Stone, R., and Tumer, I., 2006a. Prescribing and implementing the risk in early design (RED) method. *In*: *ASME international design engineering technical conferences*. *In*: *ASME 2006 international design engineering technical conferences and computers and information in engineering conference (IDETC/CIE2006)*, 10–13 September 2006, Philadelphia, Pennsylvania, USA.

Lough, K.G., Stone, R., and Tumer, I., 2006b. The risk in early design (RED) method: likelihood and consequence formulations. *In*: *ASME international design engineering technical conferences*. *In*: *ASME 2006 international design engineering technical conferences and computers and information in engineering conference (IDETC/CIE2006)*, 10–13 September 2006, Philadelphia, Pennsylvania, USA.

MacDuffie, J.P., Sethuraman, K., and Fisher, M.L., 1996. Product variety and manufacturing performance: evidence from the international automotive assembly plant study. *Management Science*, 42 (3), 350–369.

Robert, R.A., Tumer I.Y., and Stone R.B., 2002. Deriving function failure similarity information for failore free rotorcraft component design. *In*: *Proceedings of 2002 ASME design enginnering technical conference and computers and information in engineering conference*, 29 September – 2 October 2002, Montreal, Canada.

Shahrokhi, M. and Bernard, A., 2004. Risk assessment/prevention in industrial design processes. *In*: *System, man and cybernetics, IEEE international conference*, Vol. 3, 2592–2598.

Stamatis, D.H., 2003. *Failure mode and effect analysis: Fmea from theory to execution*. 2nd ed. Milwaukee, Wisconsin: ASQ Quality Press.

Stone, R.B., Tumer, I., and Roberts, R., 2002. Deriving function failure similarity information for failore free rotorcraft component design. *In*: *ASME Design Engineering Technical Conference*, Montreal, Canada.

Stone, R.B., Tumer, I.Y., and Van Wie, M., 2005. The function–failure design method. *Journal of Mechanical Design*, 127 (3), 397–407.

Stump, B. and Badurdeen, F., 2012. Integrating lean and other strategies for mass customization manufacturing: a case study. *Journal of Intelligent Manufacturing*, 23 (1), 109–124.

Swaminathan, J.M. and Tayur, S.R., 1998. Managing broader product lines through delayed differentiation using vanilla boxes. *Management Science*, 44 (12, part 2), 161–172.

Tapiero, C.S., 1987. Production learning and quality control. *IIE Transactions*, 19 (4), 362–270.

Thornton, A.C., 1999. A mathematical framework for the key characteristic process. *Research in Engineering Design*, 11 (3), 145–157.

Tixier, J., *et al.*, 2002. Review of 62 risk analysis methodologies of industrial plants. *Journal of Loss Prevention in the Process Industries*, 15 (4), 291–303.

Tumer, I.Y. and Stone, R.B., 2003. Mapping function to failure mode during component development. *Research in Engineering Design*, 14 (1), 25–33.

Tuncel, G. and Alpan, G., 2010. Risk assessment and management for supply chain networks: a case study. *Computers in Industry*, 61 (3), 250–259.

Twede, D., Clarke, R.H., and Tait, J.A., 2000. Packaging postponement: a global packaging strategy. *Packaging Technology and Science*, 13 (3), 105–115.

Wang, W. and Wang, J., 2003. Research on approach of supply chain planning with quality control in an OKP environment. *In*: *Proceedings of the 42nd IEEE Conference on Decision and Control*, December 2003, Maui, Hawaii, USA.

Wu, L., De Matta, R., and Lowe, T.J., 2009. Updating a modular product: how to set time to market and component quality. *IEEE Transactions on Engineering Management*, 56 (2), 298–311.

Yigit, A.S., Ulsoy, A.G., and Allahverdi, A., 2002. Optimizing modular product design for reconfigurable manufacturing. *Journal of Intelligent Manufacturing*, 13 (4), 309–316.