International Conference
on Industrial Engineering and Systems Management
IESM 2011
May 25 - May 27
METZ - FRANCE

A MILP model for joint product family and supply chain design ⋆

Bertrand BAUD-LAVIGNE [a,b], Bruno AGARD [b], Bernard PENZ [a]

[a] *Université de Grenoble / Grenoble INP / UJF Grenoble 1 / CNRS
G-SCOP UMR5272 Grenoble, F-38031, France*

[b] *CIRRELT, Département de Mathématiques et Génie Industriel
École Polytechnique de Montréal
C.P. 6079, succ. Centre-Ville, Montréal (Québec), H3C 3A7, Canada*

Abstract

This paper tackles the difficult problem of joint family product and supply chain design. This issue attracted more and more attention in the recent years. In this paper, we propose a Mixed Integer Linear Program (MILP) model which integrates product, sub-assembly and component substitution possibilities to a supply chain design model. First, the problem is described, notions of supply chain and product substitution are explained. The concept of substitution is discussed and illustrated. Then we present the MILP which includes joint product and supply chain optimization. Finally, experiments characterize the complexity of the problem through the comparison of the resolution length and efficiency with and without substitution.

Key words: Supply chain design, Product family, Mixed Integer Linear Programming

1 Introduction

In a highly competitive business environment, companies must diversify their offers to meet customer's demands. This diversity affects conception, production and distribution processes in decreasing economies of scale. In this conditions, the question is how to offer a wide variety of products that meets customer's needs, while controlling production and logistical costs? Leverage actions that are considered in this paper are the supply chain design and the product family design through sub-assembly substitution.

Works on product family design usually take into account production and logistical constraints. Mass customization [10] is now a well spread conception technique to achieve large variety of product designs at minimum cost. This allows focusing on the optimization of the product family, considering that assembly constraint are more flexible. Many problematics arise as determining the product family variety [3] or the module optimization [1] to minimize production costs.

However, product and supply chain optimization have not been tackled simultaneously until recent years. The need of joint optimization has been highlight by [2] and [4], showing that both decisions have impacts

⋆ This paper was not presented at any other revue. Corresponding author B. Baud-Lavigne.
Email addresses: bertrand.baud-lavigne@grenoble-inp.fr (Bertrand BAUD-LAVIGNE), bruno.agard@polymtl.ca (Bruno AGARD), bernard.penz@grenoble-inp.fr (Bernard PENZ).

on each other. [2] focuses on a production network and compares sequential design with simultaneous design on a case study and with an analytic analysis. This is partially invalidate by [7] which emphasizes the role of safety stock in the model. Supply chain design models considering bill-of-materials (BOM) are recent and still not much studied. A single-period, multi-product and multi-level models has been proposed by [9], and a multi-period model has been presented by [11]. In these models, BOM is fixed.

A very few studies optimize the product and the supply chain. Two approaches are listed in the literature. The first approach seeks to define the best product family which meets the market needs, by using generic BOM to model the product part of the problem ([8], [12]). In these formulations, BOM are determined so as to respect assembly constraints. The second approach considers the final products as fixed, but the BOM are more-or-less flexible. To model this in an assembly-to-order context where the final assembly time is constrained, [5] considered function and modular design, in which all the assemblies are possible. [6] defined several alternative BOM, one being selected in the optimal solution. This approach needs a complete enumeration of all product configurations. In return, both formulation and solution are facilitated. The only fully integrated model to our knowledge has been proposed by Chen in 2010 [4].

The aim of this paper is to provide a simple integer programming model for the joint product and supply chain optimization problem. Section 2 describes the aim of the model, its mathematical formulation and some limitations. Experiments are led in Section 3 to characterize the complexity of the model. In particular, the impact of substitution on the model resolution is highlighted. Section 4 concludes the paper and give some perspectives.

2 A optimization model for joint product and supply chain

2.1 Model description

In this study, the issue is to define simultaneously the supply chain structure and the bill-of-materials (BOM) of the product family. The supply chain has three layers : suppliers, production units and customers. Product families are represented through the BOM of each product which shares some common sub-assemblies and components. By definition, all assemblies can be produced and transported from a unit to another.

The aim of the optimization is to decide :

(1) the bill-of-materials composition : selection of the relevant components and sub-assemblies,
(2) the suppliers selection / production centers location,
(3) from where each component is supplied,
(4) how many workstations are allocated to each production center,
(5) the allocation of each assembly to each production center.

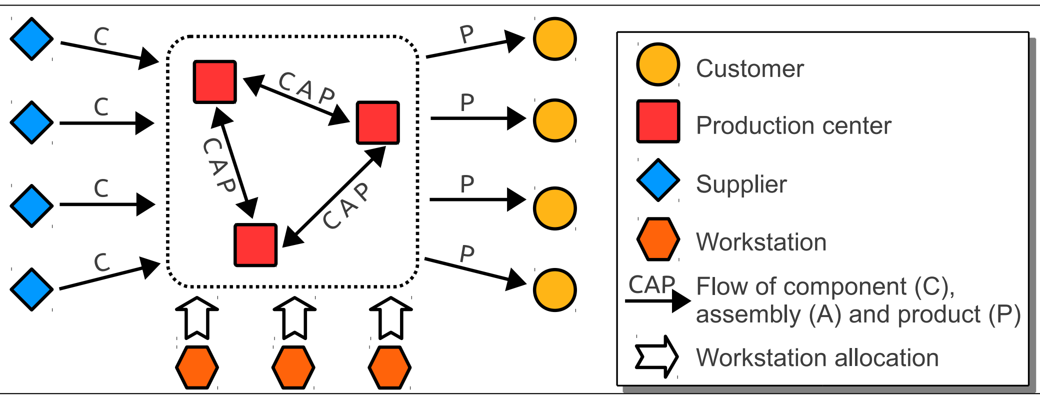
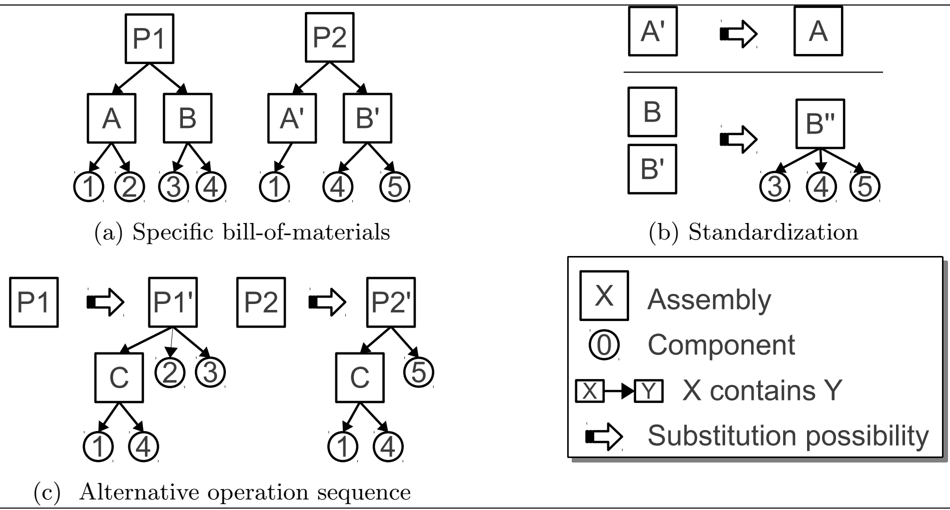


Fig. 1. Schematic supply chain network.

Products and supply chain are illustrated in Figure 1 and 2. Figure 1 describes an example of a supply chain with four suppliers, three production centers and four customers. Flow contents are only components between suppliers and plants, products between plants and customers and both plus assemblies between plants. Three types of workstation can be allocated to each production center.

The example in Figure 2a considers two products: $P1$ and $P2$. Each product is composed of two sub-assemblies. $P1$ is composed of sub-assemblies A and B, while $P2$ contains A' and B'. Sub-assemblies are composed of components.



(a) Specific bill-of-materials (b) Standardization (c) Alternative operation sequence

Fig. 2. Bill-of-materials and substitution illustrations.

Substitution possibilities can be express through explicit equivalences between assemblies. Three types of substitution are considered :

- **standardization** : a component (or sub-assembly) can be upgraded by another one with more functionalities or with a better quality. In one hand, the part is more expensive to buy, to produce or to transport, so variable costs increase. On the other hand, diversity decreases and allows better economy of scale. An illustration is provided in Figure 2b. Sub-assembly A' can be replaced by A, B and B' can be replaced by a new sub-assembly B''. The principle of standardization is used in the BOM generator described in Section 3.2,
- **externalization** : a sub-assembly is bought directly from a subcontractor. Fixed costs are nearly avoid while increasing variable costs. In the example on Figure 2a, the sub-assembly A or B could be replaced by a component bought from a supplier,
- **alternative operating sequence** : other possible assembling order can permit better commonality without necessarily changing costs. Figure 2c presents a sequence where sub-assembly C has a good commonality without adding extra function.

This paper focuses on standardization, although these substitutions options can be expressed by the model described in the following section.

The concept of substitution has been reconsidered in order to model the problem with a reasonable amount of linear constraints. The typical way to express substitution would be to use the BOM as a decision variable. In our understanding, this would conduct to quadratic constraints between BOM and production decision variables in most supply chain design model. Another way proposed in [4] is to use decision variables to express exactly which quantity of each alternative is used for each produced assembly produced. This leads to a considerable amount of decision variables. Our model described below has a simplified approach : product substitution is considered through product transformation. When part X is able to substitute part Y, a virtual process can transform X into Y. Then, quantity of Y on a plant is mixed between actual Y part and alternatives that have been transformed in Y. This view of the problem allows substitution while keeping a light formulation. Indeed, the number of additional variables is exactly the substitution possibilities.

2.2 *Mathematical formulation*

The problem is modelled with flow and fixed cost constraints, as in [9]. The breakthrough of this paper is to include substitution possibilities at each level of the BOM (components, sub-assemblies and final product). The supply chain and the product family are optimized simultaneously.

Sets:

- $\mathcal{P}$: products ; index : $p, q \in \mathcal{P}$
  - $\mathcal{R} \subset \mathcal{P}$: raw materials or components,
  - $\mathcal{M} \subset \mathcal{P}$: manufactured products / sub-assemblies,
  - $\mathcal{F} \subset \mathcal{P}$: finished products.
- $\mathcal{N}$: network nodes ; index : $i, j \in \mathcal{N}$
  - $\mathcal{S} \subset \mathcal{N}$: suppliers,
  - $\mathcal{U} \subset \mathcal{N}$: production centers,
  - $\mathcal{C} \subset \mathcal{N}$: customers.
- $\mathcal{L}$: workstations ; index : $l \in \mathcal{L}$

Parameters:

- g^{pq} : quantity of q in p. q can be a component or a sub-assembly. g represents the bill-of-materials, $p \in \mathcal{M} \cup \mathcal{F}, q \in \mathcal{R} \cup \mathcal{M}$,
- d_i^p : demand of product p by client i, $p \in \mathcal{M}, i \in \mathcal{C}$,
- c^l : workstation l capacity, $l \in \mathcal{L}$,
- t^{pl} : processing time of product p on workstation l, $p \in \mathcal{M} \cup \mathcal{F}, l \in \mathcal{L}$,
- $\mathcal{L}^p \subset \mathcal{L}$: workstations needed by product p, $p \in \mathcal{M} \cup \mathcal{F}$,
- $\mathcal{P}^p \subset \mathcal{P}$: products that can substitute p, $p \in \mathcal{P}$.

The decision variables are presented in table 1. Each variable is associated with its proper cost.

Table 1
Decision variables and their associated costs.

	Domain	Decision variable	Associated cost
Quantity of p produced on i	$\mathbb{N}$	A_i^p	α_i^p
Production of p on i	$\{0,1\}$	B_i^p	β_i^p
Quantity of p that substitute q on i	$\mathbb{N}$	S_i^{pq}	σ_i^{pq}
Flow of p between i and j	$\mathbb{N}$	F_{ij}^p	ϕ_{ij}^p
Use of flow of p between i and j	$\{0,1\}$	T_{ij}^p	τ_{ij}^p
Use of axe between i and j	$\{0,1\}$	O_{ij}	ω_{ij}
Number of workstation l on i	$\mathbb{N}$	L_i^l	λ_i^l
Use of node i	$\{0,1\}$	Z_i	ζ_i

The mathematical model is as follows.

The objective function (1) minimizes procurement, production and transportation fixed and variable costs.

$$\begin{aligned} Z = min & \sum_{i\in\mathcal{N}}\sum_{p\in\mathcal{P}}(A_i^p\alpha_i^p + B_i^p\beta_i^p) \\ + & \sum_{i\in\mathcal{N}}\sum_{p\in\mathcal{P}}\sum_{q\in\mathcal{P}^p} S_i^{qp}\sigma_i^{qp} \\ + & \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}\setminus\{i\}}\sum_{p\in\mathcal{P}}(F_{ij}^p\phi_{ij}^p + T_{ij}^p\tau_{ij}^p) \\ + & \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}\setminus\{i\}} O_{ij}\omega_{ij} \\ + & \sum_{i\in\mathcal{N}}\sum_{l\in\mathcal{L}} L_i^l\lambda_i^l \\ + & \sum_{i\in\mathcal{N}} Z_i\zeta_i \end{aligned} \tag{1}$$

Constraints (2) to (5) are flow constraints. The sources are the component flows from the suppliers to the plants, the sinks are final product flows to customers.

Constraint (2) considers the flow of each manufactured assembly on each production center.

$$A_i^p + \sum_{j\in\mathcal{U}\setminus\{i\}} F_{ji}^p + \sum_{q\in\mathcal{P}^p} S_i^{qp} = \sum_{j\in\mathcal{U}\setminus\{i\}} F_{ij}^p + \sum_{q\in\mathcal{M}\cup\mathcal{F}} g^{qp}A_i^q + \sum_{q/p\in\mathcal{P}^q} S_i^{pq} \qquad \forall i\in\mathcal{U}, \forall p\in\mathcal{M} \tag{2}$$

Constraint (3) considers the flow of each component on each production center.

$$\sum_{j\in(\mathcal{S}\cup\mathcal{U})\setminus\{i\}} F_{ji}^{p} + \sum_{q\in\mathcal{P}^p} S_i^{qp} = \sum_{j\in\mathcal{U}\setminus\{i\}} F_{ij}^{p} + \sum_{q\in\mathcal{M}\cup\mathcal{F}} g^{qp}A_i^q + \sum_{q/p\in\mathcal{P}^q} S_i^{pq} \qquad \forall i\in\mathcal{U},\forall p\in\mathcal{R} \tag{3}$$

Constraint (4) considers the flow of each component on each supplier.

$$A_i^p = \sum_{j\in\mathcal{U}} F_{ij}^p \qquad \forall i\in\mathcal{S},\forall p\in\mathcal{R} \tag{4}$$

Constraint (5) considers the flow of each final product on each production center.

$$A_i^p + \sum_{j\in\mathcal{U}\setminus\{i\}} F_{ji}^p + \sum_{q\in\mathcal{P}^p} S_i^{qp} = \sum_{j\in\mathcal{U}\cup\mathcal{C}\setminus\{i\}} F_{ij}^p + \sum_{q/p\in\mathcal{P}^q} S_i^{pq} \qquad \forall i\in\mathcal{U},\forall p\in\mathcal{F} \tag{5}$$

Constraint (6) assures that customer's demands are satisfied.

$$\sum_{j\in\mathcal{U}} F_{ij}^p = d_i^p \qquad \forall i\in\mathcal{C},\forall p\in\mathcal{F} \tag{6}$$

Constraint (7) assures that fixed costs are paid when a component is provided by a supplier or when an assembly is manufactured on a center. A_{max}^p is the upper bound of $A_i^p \quad \forall i\in U$.

$$A_i^p \le B_i^p A_{max}^p \qquad \forall i\in\mathcal{U}\cup\mathcal{C},\forall p\in\mathcal{P} \tag{7}$$

Constraint (8) assures that fixed costs are paid when a supplier or a center is used.

$$B_i^p \le Z_i \qquad \forall i\in\mathcal{U}\cup\mathcal{C},\forall p\in\mathcal{P} \tag{8}$$

Constraint (9) defines the number of workstation needed on a center.

$$\sum_{p/l\in\mathcal{L}^p} t^{pl}A_i^p \le L_i^l c^l \qquad \forall i\in\mathcal{U},\forall l\in\mathcal{L} \tag{9}$$

Constraint (10) assures that fixed costs are paid when a product is transported between two nodes.

$$F_{ij}^p \le T_{ij}^p A_{max}^p \qquad \forall i\in\mathcal{N},\forall j\in\mathcal{N}\setminus\{i\},\forall p\in\mathcal{P} \tag{10}$$

Constraint (11) assures that fixed costs are paid when a link between two nodes is used.

$$T_{ij}^p \le O_{ij} \qquad \forall i\in\mathcal{N},\forall j\in\mathcal{N}\setminus\{i\},\forall p\in\mathcal{P} \tag{11}$$

Constraint (12) limits substituted products to be used within the center they has been created.

$$\sum_{q\in\mathcal{P}^p} S_i^{qp} \le \sum_{q\in\mathcal{M}\setminus p} g^{qp}A_i^q + \sum_{j\in\mathcal{C}} F_{ij}^p \qquad \forall i\in\mathcal{U},\forall p\in\mathcal{P} \tag{12}$$

The size of the problem is $\mathcal{O}(|\mathcal{P}|^2|\mathcal{N}| + |\mathcal{P}||\mathcal{N}|^2 + |\mathcal{L}||\mathcal{N}|)$ variables and $\mathcal{O}(|\mathcal{P}||\mathcal{N}|^2 + |\mathcal{L}||\mathcal{N}|)$ constraints. The strength of this formulation is to keep linear constraint while limiting the number of additional variables induced by substitution possibilities. Substitution is formulated as a transformation into the desired part. The main drawback is the impossibility to constrain a product to be produced in only one way. Thus, a product can be made from some substituted part in a plant, and in another way elsewhere. The same plants can even use different way to produce assemblies because of capacity restrictions.

3 Experiments

3.1 Design of experiments

Experiments are led by solving the MILP presented in Section 2.2 with ILOG CPLEX 10.2 Java libraries on a server under 64 bits OS with a 2.50 GHz Intel Xeon CPU and 6GB of memory. One processor is used for the resolution. Table 2 summarizes the parameters used in the experiments. Each node of the network has a specific geographic location, which is represented by coordinates between 0 and 1 ; those are used to calculate euclidean distance between units, thus transportation costs. Supplier is defined by a set of component it can supply and a price level. Plant is defined by a labour rate and a fixed cost level. Customer is defined by its demand for each product : demand of a customer for a product is null with a probability *Prob demand* else between 0 and a maximum demand, following a uniform distribution. All the data are generated from the following parameters : number of suppliers, production centers, customers, workstations and products. Table 3 presents the size of the data sets used in Figures 3, 4 (F.3, F.4) and the eight data sets used in Table 4 (T. 4 (x)). BOM is generated as described in the following section.

When several instances are generated from the same parameter set, a different seed is used to generate random numbers. When referring to *Instance x*, the seed x is used. The size of the problem slightly differs from an instance to another because the number of sub-assemblies and components of the BOM is a result of random runs.

Table 2
Common parameters used to generate the instances.

Type	Parameter	Value
Plants	Min. labour rate	30
	Max. labour rate	5
	Fixed cost level	5
Fixed costs	per (axe, product)	100
	per axe	1000
	per (component, supplier)	100
	per (product, plant)	1000
	per suppliers	1000
	per plants	100,000

Type	Parameters	Values
Network node	Logistical cost	100
Suppliers	Max. Price level	10
	Prob product	.5
Customers	Prob demand	.1
Product parts	Max procurement cost	100
	Physical volume	10
Workstation	Prob. ws null	.5
	ws max. cost	10,000
	ws capacity	1000

Table 3
Specific parameters for Figures 3, 4 and Table 4.

Data sets	F.3	F.4	T.4 (1)	T.4 (2)	T.4 (3)	T.4 (4)	T.4 (5)	T.4 (6)	T.4 (7)	T.4 (8)
BOM height	2	2	2	2	2	3	2	3	3	3
Max. sub-assemblies	2	2	2	2	3	3	2	3	4	2
Max. components	2	2	2	2	3	3	2	3	4	5
Max. quality levels	3	3	3	3	3	3	10	1	3	10
Products	5	5	5	5	10	20	40	10	10	50
Customers	50	50	100	100	100	100	100	100	500	1000
Plants	*	4	5	5	10	10	15	20	15	20
Suppliers	*	10	15	50	50	30	30	30	50	50
Workstations	5	5	5	10	5	5	5	5	5	5
Max demand	100	*	1000	1000	500	500	200	500	50	1

* variable parameters

3.2 A BOM generator for a product family

To provide extended experiments, a generator of product family with a high commonality index has been developed. Data sets are often a weak point of the analysis when considering multi-level BOM. In [9] and [2], only one BOM with three levels and nine items is tested ; in [11], BOMs are generated from a three-level structure (products, sub-assemblies, components).

Our BOM generator is based on a generic BOM. The generic product is defined by a set of generic sub-assemblies, a set of components, volume and processing time. Each generic sub-assembly is defined the same way until the maximum height is reached. Each component is unique and defined by a volume, a set of quality level, a cost for each quality level and a set of potential suppliers. Number of sub-assemblies (components) contained in each assembly follows a uniform law between 0 and the parameter *Maximum assemblies (Maximum components)*. Then, this generic product is instantiated once by final product, depending of a quality target q and a level of functionality f randomly chosen. For each assembly, each component is either absent with a probability $(1 - f)$, either its quality level is chosen randomly around the target q ; then each of its generic sub-assemblies are instantiated recursively. An assembly is created if it is new, *i.e.* it does not exist another assembly with the same components and sub-assemblies. Then, all the instantiated assemblies corresponding to the same generic assembly are compared with each other: when an assembly is Pareto-dominant to another assembly *i.e.* all its components and the components of its sub-assembly are worse quality, it is added to the substitution list.

3.3 Results and analysis

The criteria which defines the difficulty of a data set are introduced in [11] mainly as the size of the problem, *i.e.* the size of the network (number of suppliers, plants, customers), the density of the allowed interactions in this network and the size of the BOMs. This statement can be partially verified in Figure 3. Only the number of plants and suppliers varies with respect to the instance structure. The figure shows

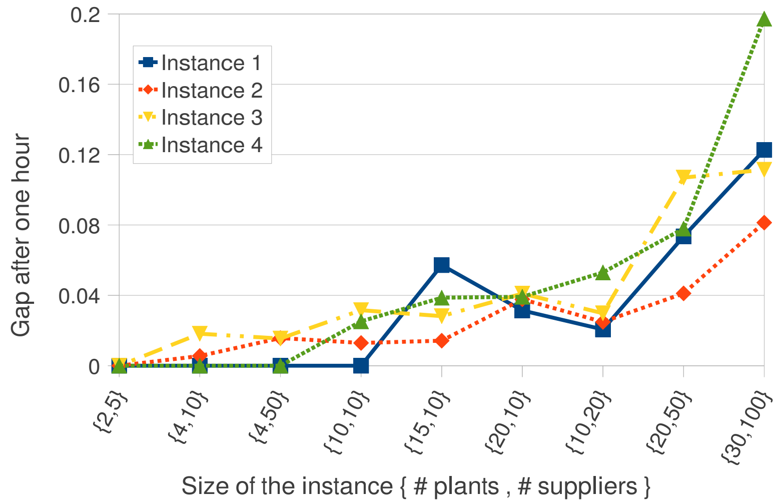


Fig. 3. Gap depending on instance size.

clearly a correlation between the difficulty to solve the problem and its size. However, some contradictory behaviours happened on instance 1, for which resolution is better for 20 plants and 10 suppliers than for 15 plants and 10 suppliers ; this can also be noticed on instance 3 between the second and the third dot. When adding particular nodes as cheap plants or suppliers, the resolution can be facilitated.

Nevertheless, another important characteristic is the equilibrium between fixed and variable costs. When variable costs are dominant on fixed costs, resolution is easier because the linear relaxation in the branch and bound is more accurate. Figure 4 represents for the same instance the resolution time depending on the parameter *Maximum demand*. This parameter is the best suited to play on the balance between fixed and variable costs without changing the structure of the instance. A clear trend is difficult to extract here because of the fluctuation of the results. Nevertheless, some behaviours can be generalized. When the demand is very high, here above 10,000, resolution is extremely easy : under one second on every

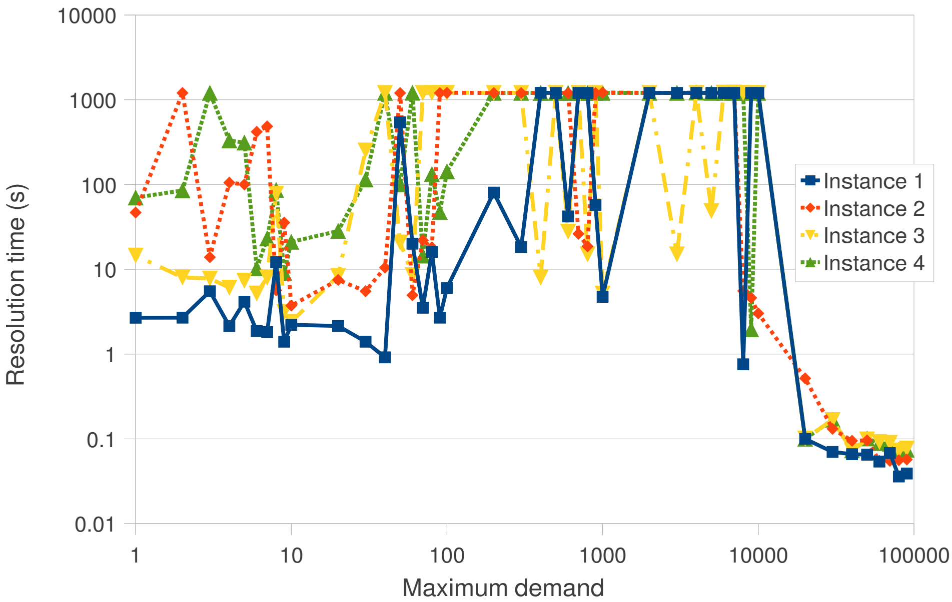


Fig. 4. Resolution time depending on demand variation (more than 1000 seconds represents no optimal solution).

instance. In this configurations, variable costs dominate fixed costs. Also, when demand is low, under 20 or 40 depending on the instance, resolution time is more reasonable. An optimal solution is nearly always found in less than 20 minutes, more often in less than 10 seconds. In this case, fixed costs are dominating. Between those values, resolution is more challenging.

Table 4 presents the results of 8 instances with the initial model and with an equivalent model without the substitution possibilities. Substitution benefits is computed as :

$$\text{Substitution benefits} = \frac{\text{optimal cost without substitutions} - \text{optimal cost with substitution}}{\text{optimal cost without substitution}}$$

Instances are characterized by different substitution benefits, from less than 1% (instances 1 and 6) to nearly 20% for instance 4. Substitution allows important saving when product diversity is higher (instances 4 and 5). When diversity is lower or production volume is high (instances 1, 2, 3 and 7), benefits are lower. Instance 6, which allows only redundancy, is only slightly impacted by substitution possibilities.

Gap is an indicator of the solution quality when no optimal solution has been found. It is computed with the higher inferior bound and the best integer solution. On these instances, gap is not significantly lower without substitution. The model with substitution is even more efficient for instances 2, 3, 5 and 6. On these experiments, resolution is not affected by the substitution possibilities. Cplex has not found any solution for instance 8, which is the bigger instance tested.

Table 4
Comparison of gap with / without substitution.

Instance	1	2	3	4	5	6	7	8
Substitution benefits (%)	.9	2.0	1.4	19.6	5.5	.4	3.3	-
Gap (%)								
with substitution	.06	.11	.29	.07	1.68	.41	1.43	-
without substitution	.04	.13	.32	.07	1.83	.50	1.41	-
Processed nodes								
with substitution	619001	373030	52301	18194	8615	17101	5509	-
without substitution	663114	355273	50215	23413	21733	16807	5530	-

Gap is obtained after 20 minutes - means no solution has been found

4 Conclusion

This paper tackles the difficult problem of joint product family and supply chain design. This issue has been attracted more and more attention in the recent years. However, most study deals with a specific part of the problem, or propose intractable models.

The aim of the paper was to integrate product substitution to supply chain design model in an elegant manner. We considered a problem for which the product family and demand is known and must be satisfied. Decisions include supply chain design – from the supplier selection, the plant location, the product allocation to their distribution – and bill-of-materials reconfiguration through substitution possibilities. The breakthrough of this paper is to consider substitution as product transformation. This allows a simple inclusion of the substitution in most supply chain design models and avoids to add much resolution complexity. Indeed, experiments have shown that substitution does not significantly worsen nor improve the resolution. We also characterize the complexity source from the instance as its size and the trade-off between fixed and variable costs. Finally, the need for complex BOM generator has been highlight, and one methodology is proposed to generate product family BOM as benchmarks.

Perspectives include :

- analysis of real case studies to characterize solution behaviours and to develop a decision-making tool. We need to determine the key variables and study the sensitivity of the solutions. A special attention is needed on the explicit integration of stock costs in the model so as to experiment results obtain in [7] on our model,
- exact and heuristic methods to fasten the resolution. Exact resolution is possible only on small instances. Use of cuts and decomposition methods have to be implemented to solve real-life case studies. Heuristics are also useful to have good results in short time. This can be needed to manually experiment scenarios,
- integration of sustainable design problematics. Products as well as production and distribution processes have an important impact on carbon emission. Preoccupations about sustainable design in product and supply chain design can be a critical success factor.

References

[1] B. Agard and B. Penz. 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, 2009.

[2] B. Baud-Lavigne, B. Agard, and B. Penz. Mutual impacts of product standardization and supply chain design. International Journal of Production Economics, doi : 10.1016/j.ijpe.2010.09.024, 2010.

[3] O. Briant and D. Naddef. The optimal diversity management problem. Operations Research, 52(4):515–526, 2004.

[4] H.-Y.S. Chen. The impact of item substitutions on production-distribution networks for supply chains. Transportation Research Part E: Logistics and Transportation Review, 46(6):803–819, 2010.

[5] R. El Hadj Khalaf, B. Agard, and B. Penz. An experimental study for the selection of modules and facilities in a mass customization context. Journal of Intelligent Manufacturing, 21(6):703–716, 2010.

[6] H.A. ElMaraghy and N. Mahmoudi. Concurrent design of product modules structure and global supply chain configurations. International Journal of Computer Integrated Manufacturing, 22(6):483–493, 2009.

[7] J. Lamothe, A. Bonnafe, M. Gorgas, L. Dupont, and M. Aldanondo. Expérimentation d'un modèle de conception d'une famille de produits et de sa chaine logistique. In Proceedings of 7ème Conférence Francophone de MOdélisation et SIMulation-MOSIM, Paris, France, 2008.

[8] J. Lamothe, K. Hadj-Hamou, and M. Aldanondo. An optimization model for selecting a product family and designing its supply chain. European Journal of Operational Research, 169(3):1030–1047, 2006.

[9] M. Paquet, A. Martel, and G. Desaulniers. Including technology selection decisions in manufacturing network design models. International Journal of Computer Integrated Manufacturing, 17(2):117–125, 2004.

[10] B. J. Pine. Mass customization: The new frontier in business competition. Harvard business school press, 1993.

[11] P. N. Thanh, N. Bostel, and O. Peton. A dynamic model for facility location in the design of complex supply chains. International Journal of Production Economics, 113(2):678–693, 2008.

[12] X. Zhang, G. Q. Huang, and M. J. Rungtusanatham. Simultaneous configuration of platform products and manufacturing supply chains. International Journal of Production Research, 46(21):6137–6162, 2008.