Algorithm for complete enumeration based on a stroke graph to solve the supply network configuration and operations scheduling problem

Julien Maheut, Jose P. Garcia-Sabater

ROGLE-Depto. de Organización de Empresas. Universitat Politécnica de Valéncia (SPAIN)

juma2@upv.es, jpgarcia@upv.es

Abstract:

Purpose: The purpose of this paper is to present an algorithm that solves the supply network configuration and operations scheduling problem in a mass customization company that faces alternative operations for one specific tool machine order in a multiplant context.

Design/methodology/approach: To achieve this objective, the supply chain network configuration and operations scheduling problem is presented. A model based on stroke graphs allows the design of an algorithm that enumerates all the feasible solutions. The algorithm considers the arrival of a new customized order proposal which has to be inserted into a scheduled program. A selection function is then used to choose the solutions to be simulated in a specific simulation tool implemented in a Decision Support System.

Findings: The algorithm itself proves efficient to find all feasible solutions when alternative operations must be considered. The stroke structure is successfully used to schedule operations when considering more than one manufacturing and supply option in each step.

Research limitations/implications: This paper includes only the algorithm structure for a one-by-one, sequenced introduction of new products into the list of units to be manufactured. Therefore, the lotsizing process is done on a lot-per-lot basis. Moreover, the validation analysis is done through a case study and no generalization can be done without risk.

Practical implications: The result of this research would help stakeholders to determine all the feasible and practical solutions for their problem. It would also allow to assessing the total costs and delivery times of each solution. Moreover, the Decision Support System proves useful to assess alternative solutions.

Originality/value: This research offers a simple algorithm that helps solve the supply network configuration problem and, simultaneously, the scheduling problem by considering alternative operations. The proposed system allows an easier generation of many different alternatives for the supply network configuration.

Keywords: stroke graph, supply network configuration, alternative operations, complete enumeration

1. Introduction

A supply network (SN) is a network of organizations involved through upstream and downstream relations in which several processes and activities are carried out to produce value in the form of products and/or services for the end customer. To face increasing demand in customized products, SNs must offer a product or service with a minimum cost and a short lead time by considering global constraints and future market opportunities.

To do so, supply chain management must be based on two pillars: supply chain integration and coordination (Stadtler, 2005). Integration, in turn, is based on three factors: partners' selection, the network's inter-organizational and organizational collaboration, and leadership. Coordination is based on the use of information and technologies, and addresses processes and advanced planning. For SNs to be able to coordinate efficiently, the literature contemplates two phases at the strategic level: supply chain design (Mohammadi Bidhandi, Yusuff, Megat Ahmad & Abu Bakar, 2009) or supply chain redesign (Nagurney, 2010), and supply network configuration (SNC) (Salvador, Rungtusanatham & Forza, 2004).

Graves and Willems (2005) were the first to introduce the SNC problem, whose objective is to determine the suppliers, products, processes and forms of transport that must be selected to minimize the costs involved. In general, this problem contemplates different possible configurations because, for instance, raw material can be purchased from different suppliers (Wang, Huang & Dismukes, 2004), products can be produced or assembled on different machines, or products can be delivered by different forms of transport (Li & Womer, 2008). Selecting a configuration implies reaching a compromise between the costs involved and the service levels to be offered to the customer. The literature includes a large number of mathematical models which address the SNC problem. We refer readers to the following reviews (Goetschalckx, Vidal & Dogan, 2002; Mula, Maheut & Garcia-Sabater, 2011). The

literature includes some case studies such as the work of (Li & Womer, 2008) which not only deals with the configuration problem, but also includes considerations at the tactical and/or operational level.

In relation to scheduling problems, lots of works are available in the literature: capacitated resources, sequence-dependent setup times (Xiaoyan & Wilhelm, 2006), lead time concepts (Sahling, Buschkühl, Tempelmeier, & Helber, 2009), multi-stage production (Seeanner & Meyr, 2012), products substitution (Chern & Yang, 2011; Lang, 2009), multi-site scheduling (Alvarez, 2007), which are just some of the characteristics that might be considered.

However to the best of our knowledge, the single product, multi-site, multi-stage, supply network configuration and operations scheduling problem considering alternative operations has not yet been resolved by contemplating complete enumeration by a stroke graph.

This article proposes the use of a stroke graph structure to enumerate all the feasible solutions for the SNC and operations scheduling when a new customized firm order arrives. The stroke graph proposed is based on the stroke concept (Garcia-Sabater, Maheut & Marin-Garcia, 2013). Complete enumeration needs different transformations of the stroke graph to then determine the total costs and delivery times of each feasible solution. A selection mechanism, that selects a set of feasible solutions to be simulated, is introduced and the specific simulation tool to solve the problem is briefly described.

The structure of the paper is as follows: Section 2 describes the case study. Section 3 proposes the complete enumeration procedure. Section 4 describes the Decision Support System that supports the algorithm and the simulation tool that assesses finding solutions. Finally, Section 5 draws conclusions and provides future research lines.

2.Case study description

The case study proposed in this article is based on a multinational company that designs, assembles and transports milling machines. The environment in which the company works might be classified as engineer-to-order, where unique products are designed to customer specifications. This company has several plants around Europe that are capable of producing parts and assembling subsystems to make the product ordered by the customer and to then transport it to the customer's plant. The products delivered to the customer are milling machines customized according to customer requirements, comprising more than 300 components and subassemblies.

Unfortunately, this company does not serve a constant and regular demand throughout the year, but generally receives sporadic unitary orders. Such discrete demand affects its operations management. This company works according to the "mass customization" philosophy. Given its sporadic demand, and with a view to being able to quickly respond to

customer requirements, it must keep a stock of those components commonly used in the majority of the products with possibilities to be ordered. Given the short delivery time expected by the market, the company is moving to a switch-to-order (Saiz & Castellano, 2008) environment for a high-demand product. When the product is not in stock and reconfiguration is expensive, the company has to be supplied for all the products. The company has dozens of suppliers for each plant and some can supply the same product with different lead times and delivery costs. Moreover, suppliers offer to deliver the product with different due dates (using various and alternative forms of transportation, and charging an extra amount given the urgency of the order).

According to its supplier's different offers, the company has to decide where the required subsystems will be assembled in order to obtain the final product. These assembly stages can be carried out in a single unique plant, or the first phase can be done in one plant and the final assembly stage done in another plant. This implies transport operations among plants.

In the case study presented in this paper, and given the large size of the involved products, it is also necessary to consider that limited resources in each plant is available space. Each plant has different areas where the various assembly operations are undertaken. These areas may be occupied for certain periods according to former programming plans.

As a basic working hypothesis, this work assumes that the products already sequenced can neither be amended nor their schedules and due dates modified. Therefore, as resources have been assigned and scheduled with a defined sequence, the available resources capacity considers an assignment prior to these operations. Then, production planning must not only assign operations to the plants that have production capacity, but must also determine when each operation must begin and end. It is worth stressing that all the operations can be done in the same area in the same plant.

This problem consists in scheduling, that is by defining when and where the production of the different operations required to deliver the end product in the customer's plant and to respect the due date actually takes place. If the due date is not met, the firm has to pay penalties. Given the possible purchasing and assembly alternatives (Maheut & Garcia-Sabater, 2011), or the BOM themselves, the problem must consider all the possible alternative operations. It is worth stressing that the firm does not consider operations which generate different products (for example, trim problems (Eisemann, 1957) or co-production problems (Vidal-Carreras, Garcia-Sabater & Coronado-Hernandez, 2012)).

Stakeholders' expectations not only center on seeking a solution at the lowest cost or the solution with the shortest delivery time, but they have determined some key performance indicators (KPIs) that can be assessed only with a simulator. For this purpose, we go on to propose a heuristic procedure based on complete enumeration to determine all the alternative feasible solutions and to assess them.

3.An algorithm based on complete enumeration

The algorithm herein presented helps to determine all the feasible alternative solutions for producing a single end product.

All the operations are represented using the stroke concept (Garcia-Sabater et al., 2013): a stroke represents any localized operation that transforms (or transports) a series of localized products (preferably measured as SKUs) into another series of localized products (also preferably measured as SKUs). This localized operation and, therefore the stroke representing it, has an associated cost and due date, and consumes a certain amount of resources. Products must consider the site where they are stored. Hence a specific nomenclature has been designed; for example, product "P01" stored in plant PA is called P01@PA.

Different strokes types are characterized as indicated below:

- Assembly and transport strokes have a minimum of one stroke input and have a single stroke output (Maheut & Garcia-Sabater, 2011). Reconfiguration operations are considered an assembly stroke and only the main product obtained is considered (co-products are neglected).

- Purchase strokes have a single stroke output, but have no stroke input.

For this problem, we hypothesize that:

- Strokes must be of only the assembly, purchase or transportation types. Strokes with several outputs cannot be considered in a single stroke.

- At least one of the products can be obtained by different strokes (in other cases, there are no alternatives).

- Product inventory levels are not planned. Those with levels high enough for operations must not be considered. Others have to be ordered with a purchase stroke.

- All SKUs must be an output of at least one stroke. This implies that a SKU has to be obtained by a purchase stroke, or by an assembly or transportation stroke.

- All the SKUs must be an output of at least one stroke, except the end product.

- The end product is the only SKU that is not the input of any other stroke.

Solving the MILP model with commercial mathematical programming software is feasible. However given the characteristics of the case study and the stakeholders' expectations, a heuristic procedure is proposed to generate all the feasible solutions. The proposed algorithm consists in five steps:

Step 1: Incorporating selection strokes

Step 2: Transforming the structure with strokes into a direct hypergraph

Step 3: Generating the complete set of arc vectors by enumeration

Step 4: Determining each feasible solution

Step 5: Assessing feasible solutions

3.1. Step 1: Incorporating selection strokes

The standard form of modeling strokes using mathematical programming is to employ a mixed integer linear programming model as in (Maheut, Garcia-Sabater & Mula, 2012). In this case, the mathematical programming is able to use the structure with strokes by deciding how many different strokes can be performed in each period. The conceptual representation of the stroke is presented in Figure 1.
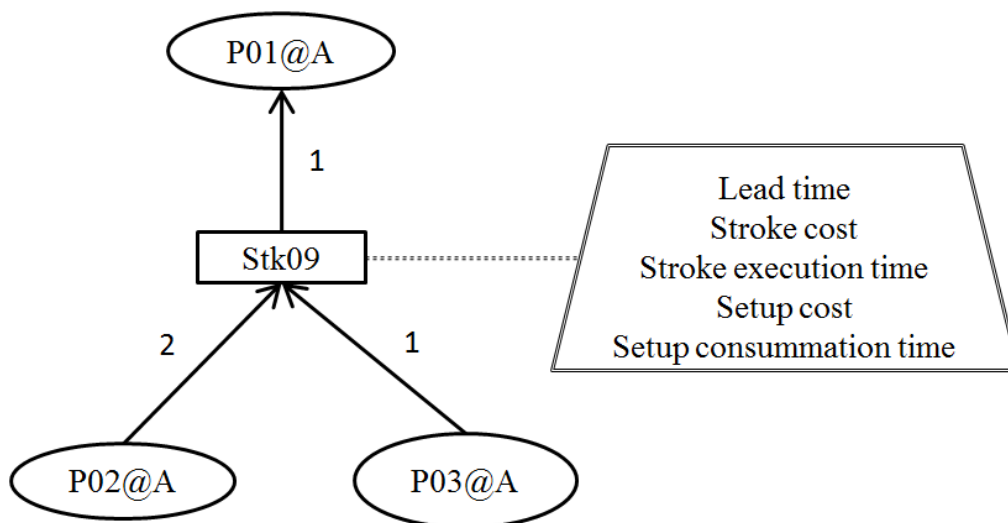

Figure 1. Example of a conceptual representation of one stroke (Maheut & Garcia-Sabater, 2011)

In Figure 1, when one unit of stroke is performed, Stroke Stk09 consumes 2 units of SKU P02@A, one unit of P03@A and generates one unit of P01@A. In our case, only the lead time and the stroke cost need be considered because operations last at least one day (the planning period). The traditional stroke graph is proposed in Figure 2.

In order to transform the stroke graph into a graph structure which enables complete enumeration, selection strokes and phantom SKUs must be incorporated to know where there are alternatives during algorithm execution (Figure 3). Alternatives exist basically when a SKU can be generated by at least two strokes.

When a SKU is the output of "Z" strokes, "Z" selection strokes and "Z" phantom SKUs must be incorporated. By definition, selection strokes are decision strokes and Phantom SKUs are

dummy products, which are considered only for algorithm purposes. Selection strokes have the SKU as output and one phantom SKU as input. These input products are not real and, as seen in Figure 3, they receive a name that reflects that they are phantom SKU from a physical one.
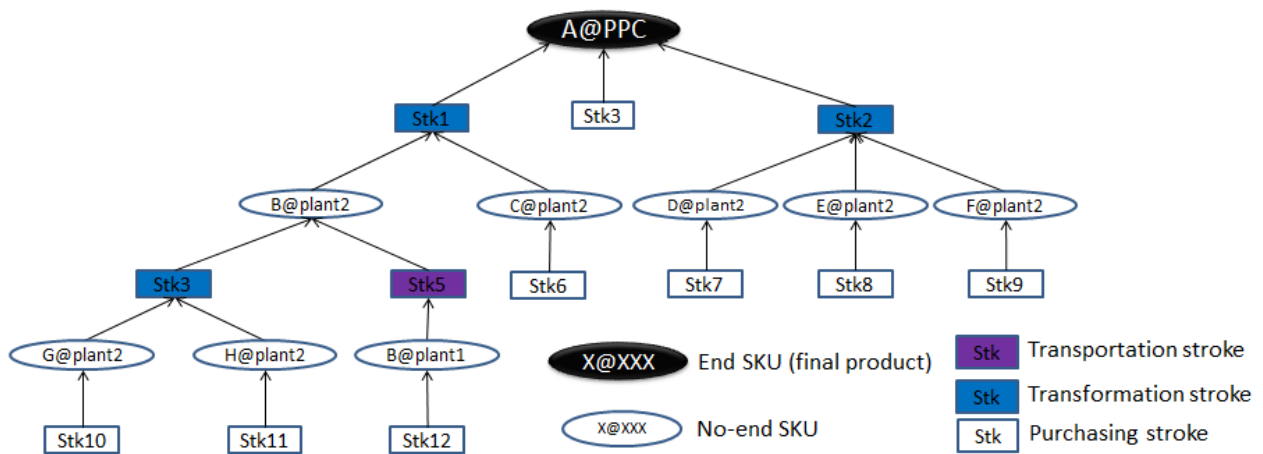

Figure 2. Traditional stroke graph structure

The last phase in this step is to associate the phantom SKU as output for each stroke.
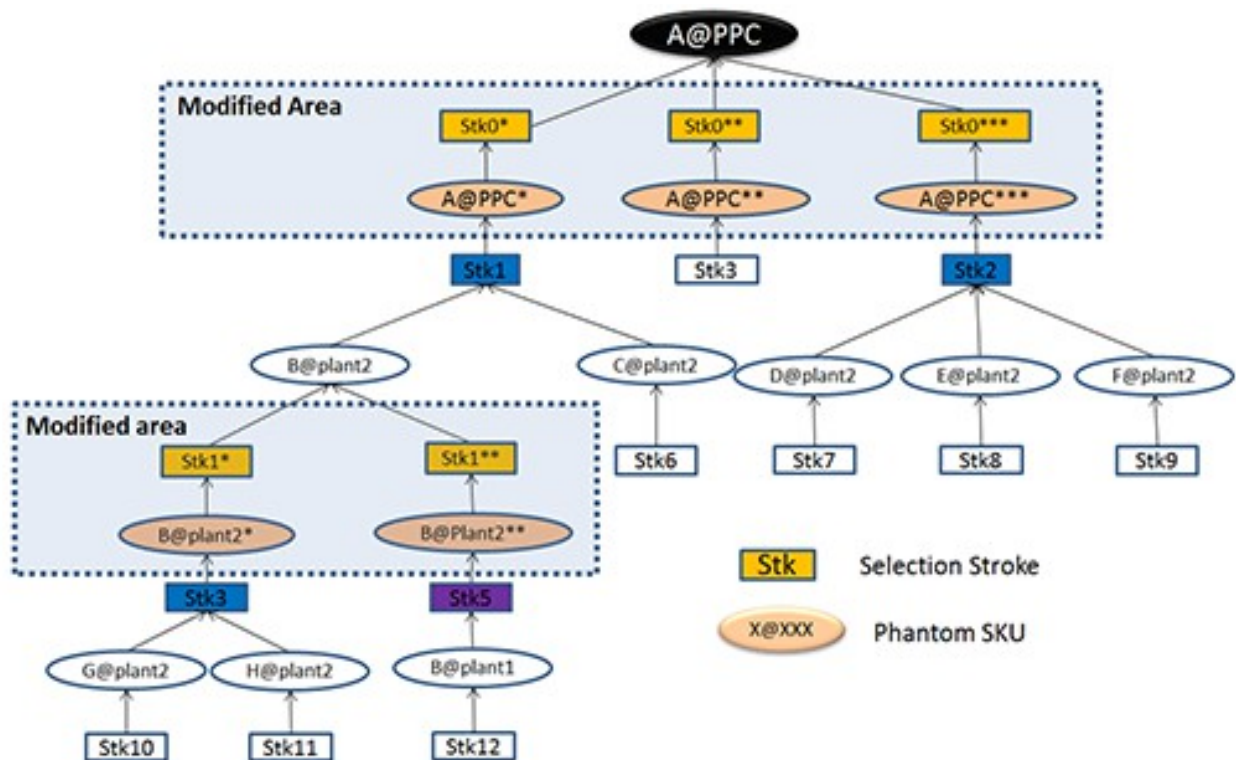

Figure 3. Stroke graph structure incorporating selection strokes and phantom SKUs

3.2. Step 2: Transforming the structure with strokes into a direct hypergraph

The next step consists in transforming the structure with strokes into a direct hypergraph. The hypergraph considers four different types of nodes, each of a different nature and two types of oriented arcs:

- Source nodes: these nodes have no input arcs, but have one output arc.

- Selection nodes: these nodes have at least two selection input arcs.

- Operation nodes: these nodes have at least one input arc, but no selection nodes.

- Terminal node: this node represents the end product. It can be a selection or an operation node, but cannot be a source node. Once again, this node has no output arcs.

- Selection arcs: these arcs have a selection-type destination node.

- Operation arcs: they have an operation-type destination node.

Step 2.1: Transforming each SKU into a node. The first transformation phase fundamentally consists in transforming each SKU (phantom or not) into a node.

Step 2.2: Creating source nodes with purchase strokes. Those SKUs obtained by a purchase stroke are transformed into source nodes.

Step 2.3: Transforming assembly and transportation strokes into operation arcs. The third transformation phase consists in associating each stroke input (which is now a node) with its operation node (stroke output) through a direct operation arc.

Step 2.4: Transforming selection strokes into selection arcs. The fourth transformation phase consists in associating each stroke input (which is now a node) with its selection node (stroke output) through oriented selection arcs.

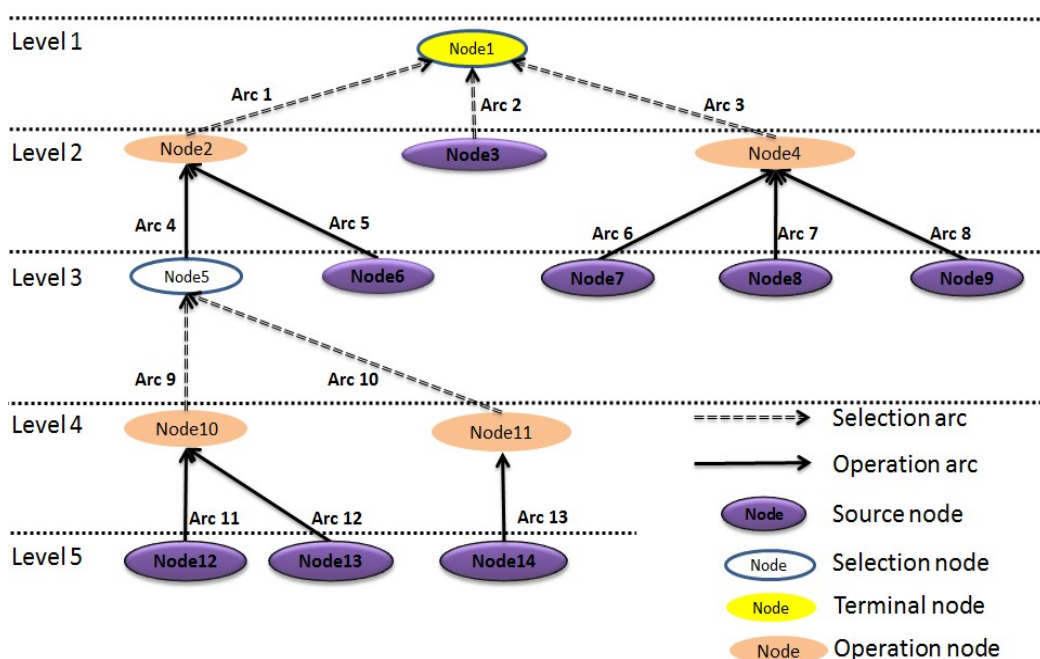The direct hypergraph obtained is the next one (Figure 4):

Figure 4. The nodes-arcs structure

As observed in Figure 4, nodes and arcs belong to different levels and are enumerated in a certain manner that enables an ordered enumeration for algorithm purposes. The transformation phase implies assigning the cost and times of strokes to nodes and arcs. This mechanism is as follows:

- Selection arcs and selection nodes have associated null times and costs.

- Source nodes inherit the resource consumption and costs of the purchase strokes that originate them.

- Operation arcs inherit the lead time and stroke cost of the transportation and assembly strokes that originate them.

Costs and lead times are translated into the new structure (see Figure 5) and the assessment presented in Step 3.5 is described herein.
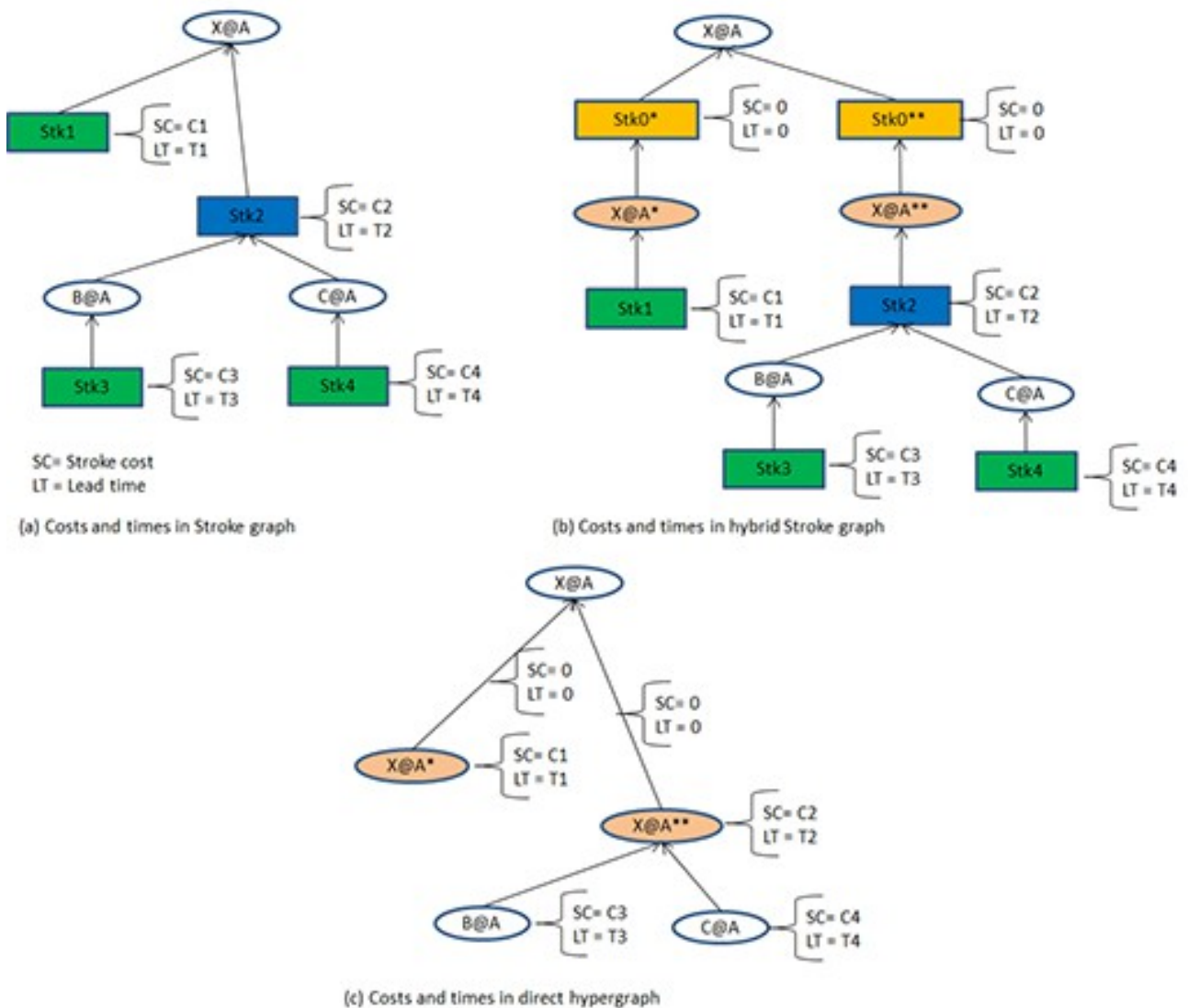

(a) Costs and times in Stroke graph

(b) Costs and times in hybrid Stroke graph

(c) Costs and times in direct hypergraph

Figure 5. Costs and lead times assignment in the different graph structures

3.3. Step 3: Generating the complete set of arc vectors by complete enumeration

Generating the set of feasible solutions by complete enumeration consists in generating all the binary arc vectors. Then, infeasible and redundant solutions are erased. The set of feasible solutions is then obtained as the set of arc vectors and node vectors that activates the terminal node.

The three substeps required are described as follows:

Step 3.1: Generating the binary arc vectors. Since the objective is to generate all the feasible solutions, if the problem consists in N selection nodes and M selection arcs, 2^M different binary vectors can be enumerated and just N vectors as a maximum can be feasible. This first sub-step consists so in generating the two 2^M binary arc vectors where:

- Operation arcs are activated in each arc vector.

- Selection arcs are activated by complete enumeration.

Step 3.2: Eliminating the redundant binary arc vectors. As the set of solutions has been enumerated completely, some of the arc vectors generated are redundant because they have too many selection arcs activated. Consequently, the following binary arc vectors must be eliminated:

- Arc vectors that have more than one activated selection arc for the selection node input.

- Arc vectors that have more than N activated selection arcs.

Step 3.3: Eliminate some infeasible arc vectors. Once the redundant arc vectors have been eliminated, some arc vectors represent infeasible solutions because the combination of the activated selection arcs does not activate the terminal node. Consequently, the following arc vectors must also be eliminated:

- Arc vectors with no selection arcs activated, when they exist.

- Arc vectors with one selection arc activated at one level of the graph and no selection arcs (if they exist) activated at the lower levels.

- Arc vectors with a deactivated selection arc at one level and a minimum of one activated selection arc at a lower level.

Once the arc vectors have been generated by complete enumeration and some of the redundant and infeasible vectors have been eliminated, the next step is to use the arc-node

structure to check the feasibility of each arc vector and to then reduce the feasible vectors to the simplest form in order to assess them.

3.4. Step 4: Generating each feasible solution

In order to check that arc vectors are feasible solutions, each binary node vector has to be used to check that the terminal node can then be activated. Next, unnecessary activated arcs and nodes must be deactivated to obtain the simplest vectors. The steps are described as follows:

Step 4.1: Initializing the binary node vector at 0.

Step 4.2: Activating source nodes in the binary node vector.

Step 4.3: Updating the graph. As arc vectors are not yet generated, the next step is to update the binary node vectors. For node = M to 1,

Step 4.3.1: If the node is of an operation-type, all the node's input arcs are activated and the input nodes of these arcs are activated, then the node is activated.

Step 4.3.2: If the node is of a selection-type, at least one of the node's input arcs is activated and the input node of the activated arcs is activated, then the node is activated.

Step 4.4: Eliminating infeasible solutions. If the terminal node is deactivated, the solution is erased.

Step 4.5: Deactivating unnecessary arcs. For arc = N to 1, all the arcs are deactivated in turn:

Step 4.5.1: If the terminal node is activated, return to step 4.5 (the next arc is deactivated).

Step 4.5.2: If the terminal node is deactivated, then that arc is re-activated; return to step 4.5.

Step 4.6: Deactivating unnecessary source nodes. For node = M to 1, if the node is of the source type, it is deactivated.

Step 4.6.1: If the terminal node is activated, the next arc is deactivated.

Step 4.6.2: If the terminal node is deactivated, that arc is re-activated.

3.5. Step 5: Assessing each feasible solution

In this step, the set of feasible and simplest solutions is generated and the next step is to assess them. Steps 1 and 2 generate the direct hypergraph and, as presented in Figure 5, costs and times are now assigned to nodes and arcs.

Step 5.1: Calculating the cost associated with each solution. The cost associated with the solution is the summation of the cost of each activated arc and node.

Step 5.2: Calculating the earliness and tardiness of beginning and ending each arc and node by considering resources availability. This step is done precisely and directly by the simulation tool during each run. Nevertheless, a procedure based on a traditional earliness and tardiness calculation of each node has been specifically designed to obtain an approximation of these times. The feasible solutions showing the worst time behavior with that approximated method are not considered to assess the exact tardiness of the project. Due to length restrictions and to the limited scope of the paper, the complete procedure has not been introduced into this paper.

This algorithm has been programmed in Java. In the case study, an application case considering 34 purchasing strokes, 8 assembly strokes and about 10 alternative strokes (alternative purchase operation and alternative BOM) has been tested. The algorithm based on complete enumeration has been implemented within a Decision Support System, which includes a simulation that evaluates the different KPIs handled by the supply network for each alternative solution.

4. The Decision Support System (DSS)

The DSS of the case study contains a database based on the stroke concept, a simulation model, which functions to transform data, and also the algorithm described herein.

When a new order arrives, different control mechanisms check that the delivery of the end product can be achieved. One of them proposes to stakeholders the incorporation of transportation strokes like transshipments to consider the transport of goods between plants. Another verifies, for instance, the possibility that all the SKUs that can be in the feasible BOMs are obtained with at least one stroke.

The main difficulty, which is where previous research has not provided results, lies in considering alternative operations. To this end, stakeholders propose alternative operations by introducing new strokes into the database.

4.1. An objective function to select feasible solutions

Having introduced all the strokes into the database, successfully performed the various implemented mechanisms to check the data and performed the algorithm based on complete enumeration, stakeholders receive a set of solutions with their associated cost and time. Next, these solutions have to be simulated in a specific simulator designed to solve the problem.

The different KPIs to be achieved at the end of the simulation execution are: lead time, delivery time, service level, plant workload level, machine cost, etc.

However, with the increased feasible solution number due to alternative operations, prolonged total resolution times given the need to simulate each solution, and as many solutions are identical in the KPIs value, a selection mechanism has been developed.

Stakeholders assumed that the two critical factors to select a better solution were the total cost and lead time associated with each solution. For each solution, a function is used to select a limited set of solutions. The various parameters used in the selection function are presented in Table 1.

$\phi \in [0,1]$	Weight of the cost value in the objective function
$\alpha = 1 - \phi$	Weight of the lead time value in the objective function
V	Value of the function for the solution considered
C^S	Cost of the solution considered
C^{max}	Maximum cost of all the feasible solutions
T^S	Lead time of the solution considered
T^{max}	Maximum lead time of all the feasible solutions

Table 1. Parameters notation

An objective function of selection (1) is used as follows:

$$V = \frac{C^S}{C^{max}}\phi + \frac{T^S}{T^{max}}\alpha \qquad (1)$$

In Figure 6, a screenshot of the experiment setup page to fix the different weights used in the objective function is presented.
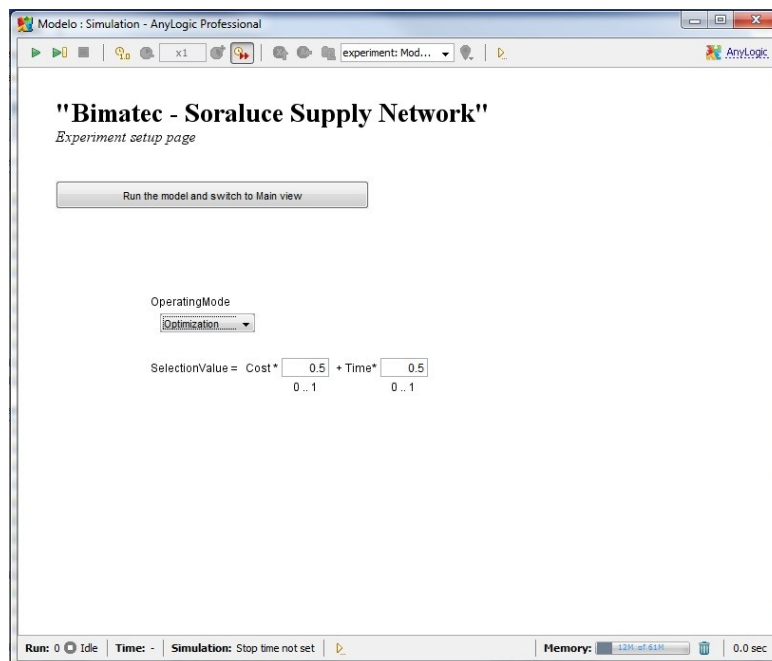

Figure 6. The experiment setup page

Based on the classification, stakeholders decide the set of solutions to be simulated in the simulation model.

In our case, with a set of 30 alternatives, it took about 4 minutes to identify and assess all the feasible solutions. Based on the classification of the objective function values, stakeholders decide the set of solutions to be simulated in the simulation model. As each simulation run lasts about 10 minutes, they obtain the options to select all the solutions or part of them.

4.2. Simulation of the solutions

After selecting the set of solutions to be simulated, the associated strokes performed in each solution are identified. Then an instance of the SN for the order is build. This mechanism to transform solutions is performed to allow stakeholders to physically observe the network.

To do this, the strokes to be performed to complete the order are obtained from the arcs activated in the chosen solution, so it is easy to observe where to execute each operation. With these data, a Supply Network Strategy Customer Service is generated. The SN is configured for each solution, the simulation is run, and a more realistic scheduling plan is generated.

A screenshot of the model designed in the AnyLogic® software is shown in Figure 7.

Figure 7. Simulator interface

In the interface, three main parts can be observed:

- The SN configuration at the top. Suppliers are on the left, three plants considered in this instance are in the centre, and customers are on the right. Each plant is divided into the various main processes. Lines in bold denote that there is a material flow among the different processes, suppliers and customers in the simulated solution.

- The different KPIs of the SN are below the SN representation. Numerically KPIs values are on the left and, on the right, KPIs are graphically represented.

Then, stakeholders make a multicriteria decision based on the value of the different KPIs obtained after each run.

5. Conclusions

In this paper, a complete enumeration algorithm based on a stroke graph is used to generate all the feasible solutions. Each time a new order arrives, the proposed procedure offers stakeholders all the feasible solutions which are needed to be evaluated in a specific simulator. Then, a selection function and the simulation tool are introduced.

As a further research line, algorithms that consider multi-products must be developed in an initial step. Another research line extends the algorithm to consider the strokes that are not only assembling process, but also splitting ones. Further research is required to solve the problem in a distributed manner, and stakeholders can appreciate considering a possible re-scheduling activity as in (Lloret, Garcia-Sabater & Marin-Garcia, 2008) in order to determine if order delivery lead times can be reduced.

Acknowledgements

References

Alvarez, E. (2007). Multi-plant production scheduling in SMEs. *Robotics and Computer-Integrated Manufacturing,* 23, 608-613. http://dx.doi.org/10.1016/j.rcim.2007.02.006

Chern, C.C. & Yang, I.C. (2011). A heuristic master planning algorithm for supply chains that consider substitutions and commonalities. *Expert Systems with Applications,* 38, 14918-14934. http://dx.doi.org/10.1016/j.eswa.2011.05.055

Eisemann, K. (1957). The trim problem. *Management Science,* 3, 279-284. http://dx.doi.org/10.1287/mnsc.3.3.279

Garcia-Sabater, J.P., Maheut, J., & Marin-Garcia, J.A. (2013). A new formulation technique to model Materials and Operations Planning: the Generic Materials and Operations Planning (GMOP) Problem. *European J.Industrial Engineering,* 7, In Press.

Goetschalckx, M., Vidal, C.J., & Dogan, K. (2002). Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms. *European Journal of Operational Research,* 143, 1-18. http://dx.doi.org/10.1016/S0377-2217(02)00142-X

Graves, S.C. & Willems, S.P. (2005). Optimizing the supply chain configuration for new products. *Management Science,* 51, 1165-1180. http://dx.doi.org/10.1287/mnsc.1050.0367

Lang, J.C. (2009). *Production and inventory management with substitutions*. (Lecture Notes in economics and mathematical systems ed.) (Vol. 636) Berlin: Springer.

Li, H., & Womer, K. (2008). Modeling the supply chain configuration problem with resource constraints. *International Journal of Project Management,* 26, 646-654. http://dx.doi.org/10.1016/j.ijproman.2007.08.004

Lloret, J., Garcia-Sabater, J.P., & Marin-Garcia, J. (2008). Cooperative Multisite Production Re-scheduling. In Y. Luo (ed.), *Cooperative design, visualization, and engineering* (5220 ed., 156-163). Springer-Verlag Berlin Heidelberg.

Maheut, J. & Garcia-Sabater, J.P. (2011). La Matriz de Operaciones y Materiales y la Matriz de Operaciones y Recursos, un nuevo enfoque para resolver el problema GMOP basado en el concepto del Stroke. *Dirección y Organización, 45,* 46-57. http://www.revistadyo.com/index.php/dyo/article/viewFile/382/402

Maheut, J., Garcia-Sabater, J.P., & Mula, J. (2012). A supply Chain Operations Lot-Sizing and Scheduling Model with Alternative Operations. In S.P. Sethi, M. Bogataj & L. Ros-McDonnell (eds.), *Industrial Engineering: Innovative Networks, 5th International Conference on Industrial Engineering and Industrial Management "CIO 2011", Cartagena, Spain, September 2011, Proceedings* (pp. 309-316). London: Springer-Verlag London.

Mohammadi Bidhandi, H., Yusuff, R., Megat Ahmad, M.M.H., & Abu Bakar, M.R. (2009). Development of a new approach for deterministic supply chain network design. *European Journal of Operational Research,* 198, 121-128. http://dx.doi.org/10.1016/j.ejor.2008.07.034

Mula, J., Maheut, J., & Garcia-Sabater, J.P. (2011). Mathematical Modelling for Supply Chain Configuration. In *Mathematical Modelling* (pp. 1-17). Nova Science Publishers, Inc.

Nagurney, A. (2010). Optimal supply chain network design and redesign at minimal total cost and with demand satisfaction. *International Journal of Production Economics,* 128, 200-208. http://dx.doi.org/10.1016/j.ijpe.2010.07.020

Sahling, F., Buschkühl, L., Tempelmeier, H., & Helber, S. (2009). Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. *Computers & Operations Research,* 36, 2546-2553. http://dx.doi.org/10.1016/j.cor.2008.10.009

Saiz, E., & Castellano, E. (2008). Driving Mass Customisation In Supply Networks. A Machine Tool Sector Case Study. In *Joint Conference of the International Mass Customization Meeting 2008 (IMCM'08) and the International Conference on Economic, Technical and Organisational Aspects of Product Configuration Systems (PETO'08)*.

Salvador, F., Rungtusanatham, M., & Forza, C. (2004). Supply-chain configurations for mass customization. *Production Planning and Control,* 15, 381-397. http://dx.doi.org/10.1080/0953728042000238818

Seeanner, F., & Meyr, H. (2012). Multi-stage simultaneous lot-sizing and scheduling for flow line production. *OR Spectrum.* In press, 1-41.

Stadtler, H. (2005). Supply chain management and advanced planning--basics, overview and challenges. *European Journal of Operational Research,* 163, 575-588. http://dx.doi.org/10.1016/j.ejor.2004.03.001

Vidal-Carreras, P.I., Garcia-Sabater, J.P., & Coronado-Hernandez, J.R. (2012). Economic lot scheduling with deliberated and controlled coproduction. *European Journal of Operational Research,* 219, 396-404. http://dx.doi.org/10.1016/j.ejor.2011.12.020

Wang, G., Huang, S.H., & Dismukes, J.P. (2004). Product-driven supply chain selection using integrated multi-criteria decision-making methodology. *International Journal of Production Economics,* 91, 1-15. http://dx.doi.org/10.1016/S0925-5273(03)00221-4

Xiaoyan, Z., & Wilhelm, W.E. (2006). Scheduling and lot sizing with sequence-dependent setup: A literature review. *IIE Transactions,* 38, 987-1007. http://dx.doi.org/10.1080/07408170600559706