The problem of corrugator sequencing and its impact on packaging processes

Ayman Hamadeh, University of Aleppo, Faculty of Informatics Engineering, Department of Artificial Intelligence

Daniel Hellström, Lund university, Faculty of Engineering, Department of Design sciences, Division of Packaging Logistics

Foreword

This report presents the result of an initial step towards an interdisciplinary research partnership between the involved departments at Aleppo and Lund University. The initiator and main author, Dr. Ayman Hamadeh, visited Division of Packaging Logistics during the summer 2008. During that time the authors identified corrugator sequencing as a common research field and embarked on a joint explorative study aiming to identify future research projects within the shared problem area.

1. Introduction

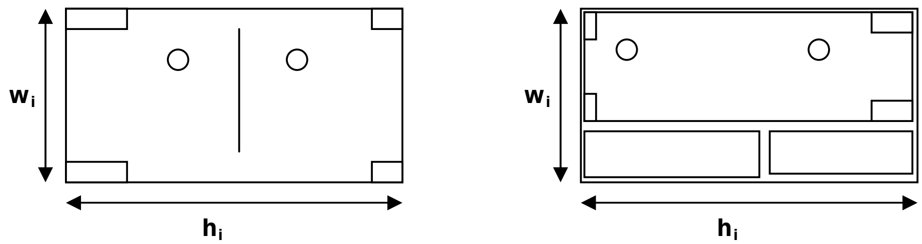
In industry, cutting problems are known complex problems. Generally, the purpose is to produce smaller pieces from larger rectangular rolls or sheets with the aim to optimise the quantity of material used and to satisfy customer orders. Each cutting problem has its own features depending on the material to be cut (wood, plastic, glass, metal, textiles), the shape of pieces to be cut (rectangular, irregular), material specification (rotation, symmetry) and cutting tool specification (guillotine, non-guillotine).

Corrugator machines produce rectangular pieces (in the paper industry they are often called sheets) with specific dimensions for each customer order and for each customer order a fixed number of pieces has to be produced. Rolls of paper (called strips) are glued together and then cut using horizontal cutters and vertical blades. Strips are available in different widths, and their height can be considered infinite. Due to technical limitations of the corrugator machine it cannot produce more than two different types of items from the same strip. Thus, the cutting problem on corrugator machine is reduced to find the best sequence of customer orders on the machine. Usually, the main objective of a cutting problem is to look for a sequence which minimises the length of rolls necessary to satisfy customer orders. However, such a cutting problem does not primarily apply to a corrugator machine since it there are other objectives to optimise, other than the sequencing of orders.

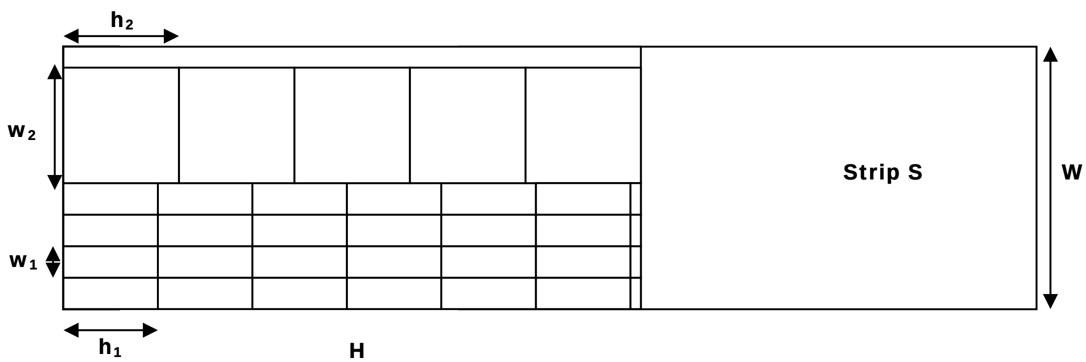
In this report, the authors set out the results of reviewing literature to expand our thoughts about the corrugator sequencing problem. There are three different approaches dealing with this field of research. The first approach uses linear programming to model quantity constraints of customer orders; this approach focuses generally on optimisation of trimmed-off waste i.e. trim waste. The second approach is based on evolutionary algorithms. Due to the complexity of parameters to be taken into account and to the relation between these parameters, the corrugator scheduling problem is a combinatorial optimisation problem and the range of potential solutions is huge. However, evolutionary algorithms can integrate many objectives such as customer services and costs level. The third approach is based on using agent-based modelling or other simulation techniques to run different scenarios.

2. The linear programming model

Let F ={F1 ,...,Fn} set of rectangular items where each Fi contains ri items with dimensions wi × hi and corresponds to a customer order. F has to be cut onto a set S of strips of different widths S ={S1, ... , Sm}.



After being cut, the rectangular pieces are processed in a conversion machine, where the rectangular pieces are converted to the ordered packaging. However, one rectangular piece can be composed of more than one packaging. This feature is not discussed in [1], so for a new order composed of 2 pieces or more there are several choices to form a new item type. These new items have to be cut with their w-edges parallel to the W-edge of the strips where each strip Si is a rectangle of width Wi and infinite height.



[1] supposes that all strips are available for all item types. This is not the case in real-life, manufacturing; each strip has a quality index and each customer order has also a quality index, so each item type has to be cut from a strip of the same quality or higher.

[1] does not take into account delivery due dates for customer orders or urgent orders. In real-life there is a need to define the order of priority between item types according to their due delivery dates or to their urgency and priority.

We call pattern each vector p $\in N^n$ of non-null components; not more than two non-null components for corrugator machine case.

p is a feasible pattern on strip S_k if and only if: $\sum_{1 \le i \le n} w_i p_i \le W_k$

The value $W(P) = W_k - \sum_{1 \le i \le n} w_i p_i$ is called the lateral waste of P.

We call scheme T on strip S_k each pair (P, H) where P is a feasible pattern on strip S_k and H is the height used of strip S_k. If P is a pattern containing items of the type i, T is called an i-scheme, if P is a pattern containing items of the types i and j, T is called an ij-scheme.

The value w(T) = w(P)×H is called the lateral waste of T. We denote by a(T) = wk × H the area of the scheme and by ni(T) = pi × ⌊ H/hi ⌋ the number of items type i contained in scheme T.

T^k set of all schemes on strip Sk ; $T = \bigcup_{1 \le k \le m} T^k$ set of all schemes.

We call 2-configuration any subset C of T. Given F (items), S (strips); Find a 2-configuration of minimum area;

for each item type i: $\forall i \sum_{T \in C} n_i(T) \ge r_i; \quad Min(\sum_{T \in C} a(T))$

2.1 Problem complexity

[1] demonstrates the following theorem: Given an instance of the 2-SSCP and an integer K∈Z+, the problem of determining if there exists a 2-configuration C with area a(C) ≤ K is strongly NP-hard.

[1] has constructed a pseudo-polynomial transformation from a 3m-partition problem which is a strongly NP-complete problem to 2-SSCP.

[1] presents a mixed integer linear programming formulation for the 2-SSCP.

$$\min \sum_{1 \leqslant k \leqslant m} W_k \sum_{p \in \mathscr{P}^k} x^p, \tag{1a}$$

Used height of pattern P

Patterns on S_k

$$\sum_{p \in \mathscr{P}} p_i y^{ip} \geqslant r_i, \quad i = 1, \ldots, n, \tag{1b}$$

$$h_i y^{ip} \leqslant x^p, \quad p \in \mathscr{P}, \ i : p_i > 0, \tag{1c}$$

$$x^p \geqslant 0, \quad p \in \mathscr{P}, \tag{1d}$$

Nb. Item i on pattern P

$$y^{ip} \in \mathbb{Z}_+, \quad p \in \mathscr{P}, \ i : p_i > 0. \tag{1e}$$

Nb. Lines of item i on pattern p: $y^{ip} = \lfloor x^p / h_i \rfloor$

All patterns

The objective function (1a) represents the total area of the configuration. Problem one can be solved by an implicit enumeration procedure based on its linear relaxation. This procedure usually suffers from important time and space requirements.

Problem R is the linear relaxation of the previous problem one.

$$\textbf{Problem R:} \quad \min \sum_{1 \leqslant k \leqslant m} W_k \sum_{p \in \mathscr{P}^k} x^p, \tag{2a}$$

$$\sum_{p \in P} p_i x^p \geqslant r_i h_i, \quad i = 1, \ldots, n, \tag{2b}$$

$$x^p \geqslant 0, \quad p \in \mathscr{P}. \tag{2c}$$

The solution of R presents the height of each pattern, which means that the number of items produced on a pattern is real. Given an optimal solution x* of problem R, one may easily construct a feasible solution of the 2-SSCP using the ROUND procedure proposed by [1]. This procedure simply increases the height of some of the schemes of configuration C(x*) in order to make it feasible with respect to the demand constraints (1b).

2.2 Conditions

Corrugator machines usually have only two (final) buffers to store produced items, one for each cutter. Each buffer can store only items of one type at a time. As a consequence, at any moment, the production of at most two types of items can be started and not completed yet. [1] has formalised this sequence constraint by the following:

(i) If T_h and T_l, $h < l$, contain i-items, then each scheme T_s with $h < s < l$ either contains i-items or is a Ts in a scheme containing one item type;
(ii) If T_h and T_l, $h < l$, are ij-schemes, then each scheme T_s with $h < s < l$ is an ij-scheme;

Due to the high set-up times of mounting the strips, [1] supposes that all the schemes on a same strip have to be processed in sequence.
(iii) If T_h and T_l, $h < l$ belong to T^k, then each scheme T_s with $(h < s < l)$ belongs to T^k.

We think that condition (iii) can be integrated in the objective function instead of considering it as a constraint

These three conditions are called sequence constraints. We call 2-Schemes Strip Cutting Problem with sequence constraint (2-SSCPsc) the restriction of the 2-SSCP to the set of the 2-configurations which satisfy the conditions (i), (ii) and (iii).

2.3 Two heuristics for the 2-SSCPsc [1]

Given a 2-configuration C we can define an undirected graph G(C) = (V(C), E(C)) as in Figure 1. The nodes of G(C) correspond to the item types in F; and E(C) edges correspond to schemes in C. For each scheme T in C; if T is an i-scheme, then T corresponds to a loop {i, i}. If T is an ij-scheme then T corresponds to an edge {i,j}.

The two heuristics are based on a graph characterisation of the feasible solutions of the 2-SSCPsc in terms of particular trees called caterpillars. Both heuristics solve the problem on the condition that all the items of the same type are cut on the same strip, and are based on the following theorem: a feasible 2-configuration C satisfies the sequencing constraint if and only if each connected component of G(C) is a caterpillar (with possible loops).

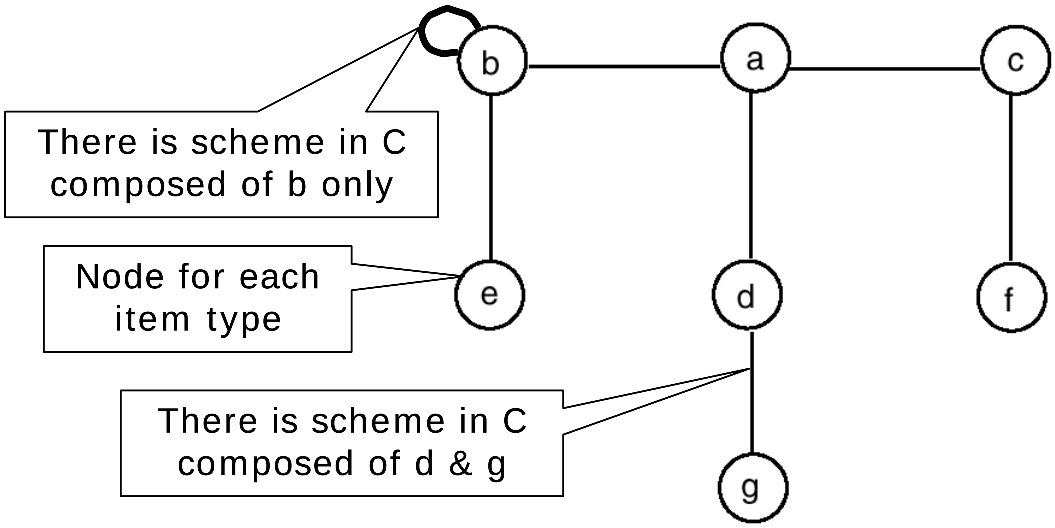


Figure 1. Graph G(C) = (V(C), E(C))

Heuristics have been implemented in C and tested on a SUN ULTRASPARC 450. [1] has used CPLEX 8.0 to solve both the linear relaxation R of problem (1) and the subsequent problems. The tests have been executed on three families, F1, F2 and F3, of instances. All the instances have been generated by randomly sampling the widths, the heights and the demands of the items and, possibly, the widths of the strips in suitable ranges according to a uniform distribution. It is more appropriate to test on real data, and it is preferable to compare the results of these heuristics with other approaches.

2.4 Minimisation of open stack problem [2,3]

There are cutting problems in many production settings, for example, in the paper, steel, textile, glass and furniture industries. In some settings, a pattern sequencing problem associated with the cutting stock problem also arises. It becomes necessary to determine a sequence in which the patterns are to be cut in order to optimise some objective functions. If a stack is opened for each new item type cut, one objective may be to minimise the maximum time during which a stack remains open; MORP, (minimisation of order spread problem), to minimise the number of times which the cutting of an item is re-initiated, that is, to minimise the number of discontinuities; MDP, (minimisation of discontinuities problem), or to minimise the maximum number of open stacks during the cutting process; MOSP,(minimisation of open stack problem). A stack remains open until the last pattern which contains the item of the stack is cut. MOSP is of interest, for instance, when there is limited space around the saw machine or when the number of (stack) mounting compartments of the saw machine is limited.

We can consider our problem 2-SSCPsc on the corrugator machine as a special case of MOSP where the number of opened stacks cannot exceed the two buffers and is not included in the objective function but is declared as a constraint.

[3] presents an integer linear programming formulation for the integrated problem and it proposes a solution procedure for its resolution. A solution to the integrated problem consists of a set of patterns at minimum cost and a sequence in which these patterns are to be processed. By using Lagrangian relaxation, [3] decomposes the problem into two sub-

problems, the cutting stock problem and the pattern sequencing problem. The dual problem is solved by a modified sub-gradient method. The cutting stock problem is of type MSSCSP (multiple stock size cutting stock problem) and is solved using column generation, after relaxing the integrality constraints of the decision variables. The pattern sequencing problem is solved using an enumeration scheme which recursively calls for an exact algorithm for the MOSP. A heuristic is used to generate an upper limit to be used in the sub-gradient method.

Computational tests using unidimensional cutting problem instances were carried out to evaluate the proposed decomposition method. The computational test results were encouraging. The decomposition procedure was able to improve the solution generated by the proposed heuristic in many of the instances tested. Its use, however, was limited to instances with a small number of items. In computational tests, there were difficulties in solving instances with 20 items in an execution time limit of 1 hour. Thus, although the proposed decomposition was general since it can also be applied to cutting problems with two or more relevant dimensions, its applicability is limited due to the increase in execution times.

3. Evolutionary Algorithms [4]

The manufacturing of corrugated cardboard boxes consists of the stages of pattern layout and finishing (i.e. printed, folded and glued) according to specifications which may vary between product styles. Because of its complexity for production management, the most important part of this manufacturing process is the pattern layout stage. In the literature, the procedures which solve the problem of pattern layout optimisation are traditionally based on linear programming models or heuristic algorithms. In real-life practice, however, some plants still schedule corrugator machines manually. The major reason is that analytical methods and good heuristics do not fully capture the problem complexity. In effect, a pattern layout which is optimal or near optimal in terms of trim waste may lead to bottlenecks at the finishing stage or to sub-optimal scheduling solutions for the whole plant. In addition, these troubles on production also concern the delivery of final product, affecting due-date-related performance indexes.

Due to the high complexity of parameters to take into account, [4] has decided to use evolutionary algorithms to provide "good" solutions with respect to all objective functions in "reasonable" time. Evolutionary algorithms represent an intelligent computer-based optimisation technique which has provided very good results when applied to solve other combinatorial and engineering optimisation problems. It therefore seems interesting to apply evolutionary algorithms to solve the corrugator scheduling problem.

Early work on the cutting stock problem is based on linear programming formulation in order to minimise trim waste. Although some of these formulations are still very popular and are used in various commercial computational packages, they lead to a large range of solutions and difficulties in dealing with non-linear problems, which are very common in the real world. As a result, heuristic solution procedures have become increasingly popular in the literature. In practical situations there is a need to have a balance between the waste minimisation, customer service, production costs, machine utilisation, and workforce utilisation. The efficiency and effectiveness of heuristic solutions, however, depend heavily on the heuristic used. Finding effective heuristics is often as difficult as solving the problem itself. When applied to the corrugator scheduling problem many heuristic approaches to the cutting stock problem have been unsuccessful either because they attempted to generate patterns sequentially and had trouble with trim loss at the end of a sequential procedure, or because they used linear programming to minimise trim, and performed poorly with regard to number of pattern changes and order congruity.

Evolutionary algorithms create and manipulate a group of possible solutions referred to as a population. Each possible solution within the population is called an individual. The population undergoes change throughout the run of the evolutionary algorithms thereby evolving the individuals toward the best solution. Within the evolutionary algorithms, the population loops through a series of processes a number of times; each executed loop is known as a generation. These processes include an evaluation process, an alteration process and a selection. These processes may occur in differing order, however, each one is required at each generation. The evaluation process uses an evaluation function which assesses the relative fitness/suitability of each individual of the population in each generation. In addition, in each generation a number of individuals are subjected to some form of change. These alterations are manifested through the use of genetic operators. A loop can be described in the eight following steps:

1) Create initial population
2) Calculate fitness values for each individual
3) Sort individuals according to their fitness values and apply the natural selection
4) Construct parents set (randomly choices)
5) Crossover operations -> sons' set
6) Mutations on sons' set
7) Add sons set to the population
8) If not termination, go to step 2

3.1 Individual solutions

A population is a group of solutions where each solution is an ordered set of patterns and a length of each pattern. So a solution is coded by two chromosomes:

- Vector of numbers identifying patterns
- Vector of pattern lengths

Chromosomes will not be of the same size in each individual.

3.2 Initial population

a) Define minimum and maximum side trim allowed
b) Generate all acceptable patterns (all feasible combinations of item types)
c) Calculate length of each pattern
d) Do the following two steps until the proportion planned of a random number of orders to be planned is at least 100%
    1. Randomly choose an acceptable cutting pattern which has not been chosen before and its length
    2. Calculate the proportion planned of each order to be planned

3.3 Fitness function evaluation

This function defines the trend of the evolution. This function is responsible for the convergence of the algorithm towards a solution satisfying the objectives as optimally as possible. Fitness values represent the power of survival for each individual. Fitness function evaluation can combine all aspects and it can be expressed as the following: Fitness = WPL (weighted planning level) + cost objectives. This multi-objective function seeks to improve the following two objectives: Cost and the WPL index.

WPL (weighted planning level):

The WPL index measures customer service, it is calculated based on the IMP value that is exclusive for each order. The IMP value represents the relevance or urgency for planning a determined order. In real corrugator planning, the urgency for planning a determined order is usually determined in an implicit way by the planner's experience and knowledge. [4] is focusing on a fuzzy inference system to calculate this IMP value. The input variables of the FIS are:

- Measure customer service
- Delivery due dates performance (high penality in the fitness function concerning deliveries that don't respect their due dates)
- Finished machines queue management (minimize queue length in front of conversion machines)
- Client-related importance (assign priority level to each order)

Cost objectives:

- Corrugator running cost (labour, maintenance, power, fuel, etc)
- Change cost (roll preparation and mounting, slitter head setting, etc)
- Side trim waste cost
- Material upgrading cost (produce higher corrugated quality than the quality specified in the customer order because if fits well with another order)
- Pattern change cost

3.4 Crossover operations

A crossover is between two individuals selected randomly to produce a new generation of solutions. There are numerous ways to build the two chromosomes of a new individual (son). [4] have defined the two following recombination operations: one-point crossover and uniform crossover, i.e. in each crossover the offspring is 4 sons. The crossover operators are illustrated in Figure 2 and 3.

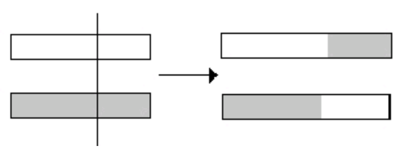


Figure 2. One-point crossover (2 sons).

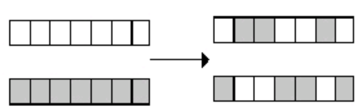


Figure 2. Uniform crossover (2 sons).

3.5 Mutation operations

Mutation is a small modification concerning the genetic code of an individual to produce a new solution (neighbouring search). There are numerous ways to define mutation operators. [4] determined two mutation operators by:

- Cutting pattern addition mutation, which adds a pattern randomly along the chromosome
- Proportion mutation, which generates a new proportion for an existing pattern

Evolutionary algorithm parameters:

- Data structure/DNA
- Creation of initial population
- Population size
- Probability of mutation/crossover
- Evaluation and selection function
- Terminating conditions

Experiences:

- Termination on number of generations
- Natural selection: the best
- Mutation with probability
- Impact of initial population?
- Other types of mutation, or crossover
- Comparison with manual scheduling

The results obtained by [4] were compared with the results provided by the company using the current manual production schedule. 8 different order sets were selected to represent the different schedules for the corrugator machine. The order sets used consist of a variety of grades, dimensions, units ordered, paper widths available, and queue status in converting machines.

[4] proposed a multi-objective evolutionary algorithm for production scheduling in a corrugator manufacturing plant, with the objective of optimising the weighted planning level (WPL). The proposed algorithm also provides some insights into the trim lost and costs incurred when upgrading the corrugator machine. In addition to the improvement in the WPL, the algorithm is enough flexible to incorporate the empirical knowledge of the production manager when they select the most appropriate schedule for the factory. Reduction in upgrading costs is also obtained after the application of the algorithm. These features make this algorithm very attractive for actual implementation.

However, the authors think that there is a lot of research still to do with evolutionary algorithms. How do the initial population impact the results? Moreover, studies are needed to compare different crossover and mutation operators.

4. Agent-based modelling and simulation

The problems arising in the management of a corrugated box factory can be understood in detail using sophisticated simulation technology. The focus in [5] is to be specific rather than general, and ensure the resulting model is not only accurate, but has a fast runtime for the given class of factories. A combination of agent-based modelling, where each interacting part is modelled as an individual entity with certain behavioural rules and decision-making capabilities, and discrete-event simulation, where the simulation of an entire process is represented by single events which might occur concurrently or consecutively over time, has proved to be a reliable approach to modelling such systems.

[5] gives a detailed account of an agent-based model built by Eurobios for SCA Packaging, one of the leading international players in the corrugated cardboard industry. See Figure 4 for an overview of the model developed by [5]. The model has been successfully implemented at multiple production sites in the United Kingdom. While the main purpose of the model is the detailed understanding which possible changes in the customer base or

the production planning have on a number of measures, it has proved itself very useful in showing how to reduce finished goods stocks without compromising on-time-in-full delivery.

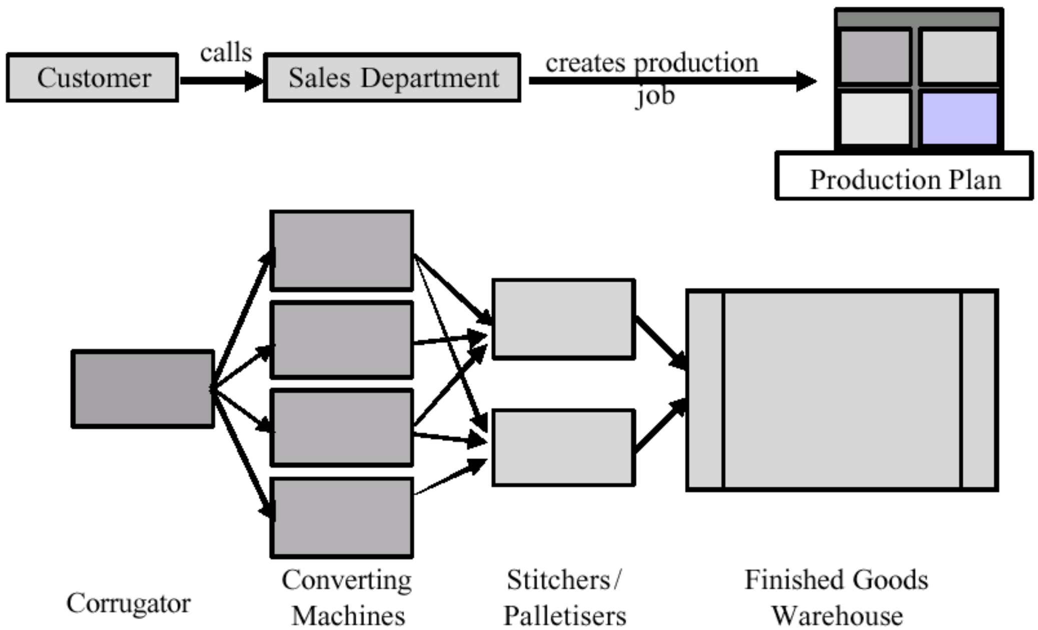


Figure 4. Overview of an agent-based model built by Eurobios for SCA Packaging.

There is an agent for each interacting part of the modelled system, for example, one of the key agent types in the model is the customer. Each customer has a number of unique products from which to order various amounts over the year. The characteristics of these orders are:

- product ordered
- quantity ordered
- delivery location
- delivery date (and time)

The products themselves are covered by a number of different attributes:

- box size
- unique print design/colour scheme
- number of boxes per pallet shipped
- agreement to keep a certain stock
- guaranteeing a certain lead time

Simulation permits its users to run different scenarios and evaluate several strategies and it also facilitates the study the trade-off between efficiency and high sensitivity to disturbances (machine failures, last minute order/order changes) and the trade-off between warehouse level and on-time-in-full delivery. For example, simulation permits its users to evaluate three strategies of insertion for a new order:

- As soon as possible
- As late as possible
- Between these two dates

[5] present an agent-based model of a corrugated box plant. [5] focus on the necessity of a detailed model of all the elements involved in the process. Using this agent-based model, SCA Packaging was given the opportunity to have a clear and reliable understanding of the consequences of possible changes to its customer base or the way its plants are run. This knowledge made it possible for plant management to take certain decisions which led to a reduction in warehouse levels by over 35% without compromising their on-time-in-full delivery commitments.

5. Concluding remarks and further research

To further study the problem of corrugator sequencing problem we aim to proceed with evolutionary algorithms. We think that there are a lot of studies needed to improve evolutionary algorithms, and to look for a new formalisation of chromosomes. It is also possible to develop new ways of crossover and mutation. We can use simulation to validate some parameter values or to test some strategies. However, there is also a need to monitor linear programming if we want to model some sub-problems if necessary.

References

1. Rinaldi, F. and Franz, A. (2007) A two-dimensional strip cutting problem with sequencing constraint. *European Journal of Operational Research* 183: 1371-1384.
2. Yanasse, H.H. and Limeira, M.S. (2004) Refinements on an enumeration scheme for solving a pattern sequencing problem. *International Transactions in Operations Research* 11: 277 –292.
3. Yanasse, H.H. and Lamosa, M.J.P. (2007) An integrated cutting stock and sequencing problem. *European Journal of Operational Research* 183: 1353-1370.
4. Velasquez, G.A., Gisella, D. and Paternina-Arboleda, BC.D (2007) A multi-objective approach based on soft computing techniques for production scheduling in corrugator manufacturing plants. *Ingeniería & DESARROLLO*: Numero 21 Enero-Junio, ISSN 0122-3461.
5. Darley, V., Tessin, P.V. and Sanders, D. (2004) An agent-based model of a corrugated box factory: the tradeoff between finished-goods-stock and on-time-in-full delivery. Proceedings 5th workshop on agent-based simulation, Coelho, H and Espinasse, B., eds. (c) SCS Europe BVBA, ISBN 3-936150-31-1 (book) / 3-936150-32-X (CD).
6. Wascher, G., Haubner, H. and Schumann, H. (2007) An improved typology of cutting and packing problems. *European Journal of Operational Research* 183: 1109-1130.