

An approach to data-mining for product driven systems

Philippe Thomas, André Thomas

▶ **To cite this version:**

Philippe Thomas, André Thomas. An approach to data-mining for product driven systems. Theodor Borangiu, André Thomas, Damien Trentesaux. Service Orientation in Holonic and Multi-Agent Manufacturing Control, Springer-Verlag, pp.181-194, 2013, Studies in Computational Intelligence, vol. 472, 978-3-642-35851-7. 10.1007/978-3-642-35852-4_12 . hal-00913300

HAL Id: hal-00913300
https://hal.science/hal-00913300

Submitted on 3 Dec 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

One Approach of Data-Mining for Product Driven Systems

Philippe Thomas. André Thomas.

Centre de Recherche en Automatique de Nancy (CRAN-UMR 7039),
Université de Lorraine, CNRS, Campus Sciences,
B.P. 70239, 54506 Vandœuvre lès Nancy cedex France
philippe.thomas@univ-lorraine.fr

Abstract. The first objective of this chapter is to highlight some new Product Driven Systems (PDS) issues. Effectively, several possibilities have been proposed to give to products or objects capacities to react to environment modifications (especially in manufacturing and logistics context here). In particular, bio-inspired approaches are now promising. All these new perspectives lead putting products in action according to collected information. That's why all technics leading to exploit and organize data are necessary. The main objective of the chapter is addressed in a second part, where we highlight why learning machines could be seen as a new way to transform data in useful knowledge.

Keywords: Product Driven Systems, Viable systems, Learning machines, Neural network, Data mining.

1 INTRODUCTION

After the second industrial revolution the main companies' objective has been productivity. Ford Motor Company has introduced the concept of mass production. Since then, many techniques have been introduced leading to process automation and optimization of planning and production control activities. Among Manufacturing Planning and Control Systems, MRP Systems emerged during the seventies in order to solve problems such as those related to delays of orders, to intermittent stock consumption or to forecasting of raw materials consumption. However, inertia facing unexpected events occurring on the shop-floor has been seen as a residual issue. In order to compensate for this drawback, a new MRP² generation has been proposed with a closed loop approach [27]. The main characteristic of these systems is its multi decision level horizons structuration. These horizons may be classified into long, medium, short and very short term. Considering them, four decision levels have been identified: strategic, tactical, operational and execution one. At the beginning of eighties, new management philosophies appeared which implied drastic changes in production management area. The main goals of these changes have been to improve the

system reactivity and flexibility, on the one hand, and the service quality, on the other. These challenges are still valid today and have been mainly implemented by Just in Time (JiT) and Theory of Constraints (ToC) philosophies.

The main idea of JiT philosophy is based on the efficient use of productive resources. Various approaches have been proposed such as Lean manufacturing [13], Demand Flow Technology [3] or Six Sigma [18]. On manufacturing shop floors, the main used tool to implement this philosophy and to control physical/material flows is the kanban system. Lean manufacturing and, in particular, kanban system have led a great revolution in production system management. The management functions, which are in centralized and hierarchical MRP² systems, become, in lean philosophy, completely or partialy distributed. In spite of the results obtained with these decentralized approaches, in the middle of the 80', Theory of Constraints (ToC) introduces a new point of view, different and complementary, based on a global optimum attainment which brings back to a centralized approach [9]. ToC induces that organization has to be evaluated and controlled by using three indicators: profits generated by sales, operating costs and inventories. In ToC, a good bottleneck management is the key of the success. Finally, the existence of these three concurrent philosophies of production management have led to the proposition of many hybrid systems using techniques coming from MRP², JiT and distributed approaches or ToC which were implemented in software such as ERP (Enterprise Resources Planning), APS (Advanced Planning Systems), SCM (Supply Chain Management Systems)...

In the next section, a brief overview of intelligent manufacturing systems is recalled with their strengths and weaknesses. Section 3 presents a viable system model for product driven system. Section 4 focuses on the need of knowledge in these systems and the learning approaches used to design knowledge. An illustration of this approach is presented in section 5 before to conclude.

2 INTELLIGENT MANUFACTURING SYSTEMS

The development of production management systems has led to the "Computer Integrated Manufacturing" concept [29]. The main goal of these systems is to interconnect all the information systems included in the production system. CIM systems have to supervise and control all company operations. At that time, the paradigm generally accepted was that the CIM system would be able to lead a great flexibility when changes occur and would give the best solution to the problems encountered in production system. Nevertheless, first implementations led to centralized and too much rigid structures unable to adapt quickly to changes; except some flexible manufacturing systems initiatives which were very productive. So, at the beginning of nineties, CIM systems were no longer considered as the solution of all problems of production companies [1].

Considering the bad results of integrated systems in terms of flexibility and reactivity, collaborations between research centers, universities and companies have been initiated in order to design and develop the production systems for the future. The most important of them was the "Intelligent Manufacturing Systems" Project (IMS)

[30]. The basic idea of IMS was the design and implementation of decentralized systems. Its main goal was system flexibility in order to deal quickly with disturbances inherent to the production processes. We can note that this function was before allocated to men who, for example, monitored and changed priorities of the shop orders on the shop floor. The idea was to automate all or a part of this function by using new communications technologies (Auto-ID, Multi-Agent systems…). These new systems have to be robust, reconfigurable and reusable [14]. These systems must allow:

- a global integration of heterogenous softwares and hardwares in a factory, a virtual factory or through a supply chain,
- an open architecture allowing to introduce or remove sub-systems,
- an efficient communication and cooperation between the different departments of a company or between companies,
- to take into account the human factors,
- to adapt itself quickly to changes or perturbations,
- to be tolerant to faults at the system level as well as to the sub-systems levels in order to collect the causes of faults and to minimize their impacts on the production.

Considering the centralization criterion, the production systems have been divided into four types [1]: centralized systems, hierarchical ones, modified hierarchical ones and heterarchical systems. Around the design of decentralized systems appeared different types of systems and concepts. The main decentralized production systems are bionic, fractal and holonic ones (HMS). This chapter focuses on the last one.

HMS consortium has proposed holonic production systems based on the holon concept. A holon is an entity which may be included in other holons [25] and are organized in holarchies. Holons have the abilities of autonomy and cooperation. Nevertheless, concepts of agent and holon are often confused. Although the holon may be viewed as an agent, the main difference is that the control part is associated to the physical part in a holon. In an agent (which is an abstract entity), a physical entity may be merged or modeled by an abstract entity. Product Driven Systems is an evolution of holonic system where interoperability and intelligence are improved. In PDS, products become the company resources controllers [16]. This leads to the intelligent product concept. This one has been defined as an entity equipped with physical and informational representations, able to affect decisions which concern the global system and the intelligent product itself [15]. In practice, the Radio Frequency identification (RFID) is a technology able to link information and physical environment [28]. The central idea is to move from a classical hierarchical and centraized control to a distributed decision making one where a part of the decision is made locally, all along the products life cycle. So, the needed information is reduced and locally processed. The PDS have been generally designed as a particular class of holonic systems. The main advantages of IMS approaches are feasibility, robustness, flexibility, reconfigurability and reusability.

Up to now, many methodologies have been proposed in order to model distributed approaches (PROSA, ADACOR, METAMORPH…). Despite this, no standardized criterion exists allowing model design. The modeling step may be performed by fo-

cusing on functional, physical or abstract aspects [4]. And yet, the tools choice, the criterions choices and the models choices remain linked to the abilities and preferences of designer. This lack of uniformity makes the evaluation and the comparison of different applications in the literature, difficult.

One of the most critical points for the heterarchical approaches (decentralized) comparatively to traditional ones (centralized) is the global decision optimization. Heterarchical systems are not able to formally guarantee their performances in terms of quantifiable variables, and more particularly, of costs. Heterarchical systems are interested in classical criterions of cost, time or efficiency, but also to goals relating to flexibility, reconfigurability, reactivity, interoperability… These goals are not easily quantifiable, and so, the comparison and evaluation of benefits of such systems are difficult.

In conclusion, it can be noticed that the two great approaches of production planning and control have strengths and weaknesses. Actually, the conventional approaches (centralized) insure the efficiency of the global system but it is hard for such systems to attain flexibility and reactivity objectives. At the opposite, the IMS approaches (distributed) insure flexibility and reactivity, but are not able to insure performance and consistency between decisions taken to different levels. No matter how complex could be supply chain or manufacturing systems, global consistency of them remains a key issue leading to hybrid (centralized and distributed) systems implementations [19].

In order to associate the advantages of each approach and to avoid their limitations, several initiatives have proposed to associate a HMS to a hierarchical planning system [26]. The function of HMS is to determine an alternative planning when disturbance occurs. As previously mentioned, another way would be to design hybrid systems (centralized/distributed). Herrera [12] highlights that to allow acceptable efficiency and consistency between different decision levels and to improve the flexibility and reactivity capabilities of such systems; “Viable Model” could be a good way to structure their architecture. Moreover these systems have to include a data acquisition system in order to collect data from the physical system to be controlled. These data must be filtered, analyzed, possibly aggregated… in order to become exploitable knowledge. The following part of this chapter focuses now on this point.

3 VIABLE SYSTEM MODEL FOR PDS

3.1 Viable System Model (VSM)

The origins of VSM arise from the works of Beer [2] applied to the steel industry in the fifties. This research can be placed in the line of works of Norbert Wiener, Warran McCulloch and Ross Ashby. The main objective of these works was to identify and to explain how systems are viable. Although, VSM is a general model for the study of any viable system, the most concerned application areas has been human activity organizations, i.e., corporations, firms or governments. In this domain, VSM changes the view of the traditional management model based on command and control, in which a control system is designed as a pyramid and such decisions are disaggregated

in a top-down manner at different structural levels. The main difference, inspired by the biological organization, consists in mapping this hierarchy into a structural recursion [12]. The premise of this change of perspective was inspired from the living beings composition (cells, organs, systems, etc.). Indeed, they have properties of autonomy, self-organization and self-regulation, allowing them to have an independent existence. The differentiation of their functions and the relationships between these elementary components produce more complex systems, without that subsystem essential properties would be lost. However, one of the most important properties of a viable system is their intrinsic recursion. In fact, any viable system contains and is contained by another viable system. Every subsystem maintains its autonomy towards its environment, but it also contributes to generate the viable system in which it is included. In that way, a viable system and its different subsystems have the same structural requirements. A viable system supports its objectives thanks to an overall cohesion and adapts itself by the autonomy of its subsystems. VSM was developed looking for invariances in organic systems. These invariances allow defining a homomorphism of their functions, organization and structure. Beer defines five elementary functions that any viable system must have: implementation, coordination, control, intelligence and policy.

3.2 VSM model of Manufacturing Planning and Control system

The model describes here (Fig. 1) has been proposed by Herrera *et al.* [12]. It is consistent with the five functions of the Manufacturing Planning and Control Systems which can be described as: Strategic Planning, Sales and Operations Planning (S&OP), Master Production Planning (MPS), scheduling and execution. Each of these functions corresponds to a level in the decision making process regarding to different horizons going from a longer to shorter one. In practice, these decisions are taken using a rolling horizon to take into account the frequent changes that occur in the data (demand, capacity, etc.). Thus and for example, the strategic planning is revised once a year, the S&OP is computed monthly, the MPS weely, and schedules are performed daily or more frequently depending on disturbances. Each function deals with a corresponding aggregation level of products, respectively families, finished products and items (components). In this context, one of the major issues is to adapt decisions at each level when disturbances (internal or external) happen. The frequently resulting modifications in the decision making process lead to the so-called system nervousness which deteriorates the global system performance (productivity and efficiency). One should notice that the shorter the horizon is, the more frequent are the changes. Thus, the performance is more deteriorated at the lower level (scheduling level). More precisely, this model is a generic model based on VSM dealing with production planning considering both MPS level and scheduling (lot-streaming).

In a PDS, the basic unit is the intelligent product, which is able to:

- acquiring and archiving data,
- communicating with its environment,
- interacting with and on it.

So, intelligent products should have necessary autonomy, auto-organization and auto-regulation properties to become the basic subsystem of a VSM model which is able to model all levels of a MRP² system.

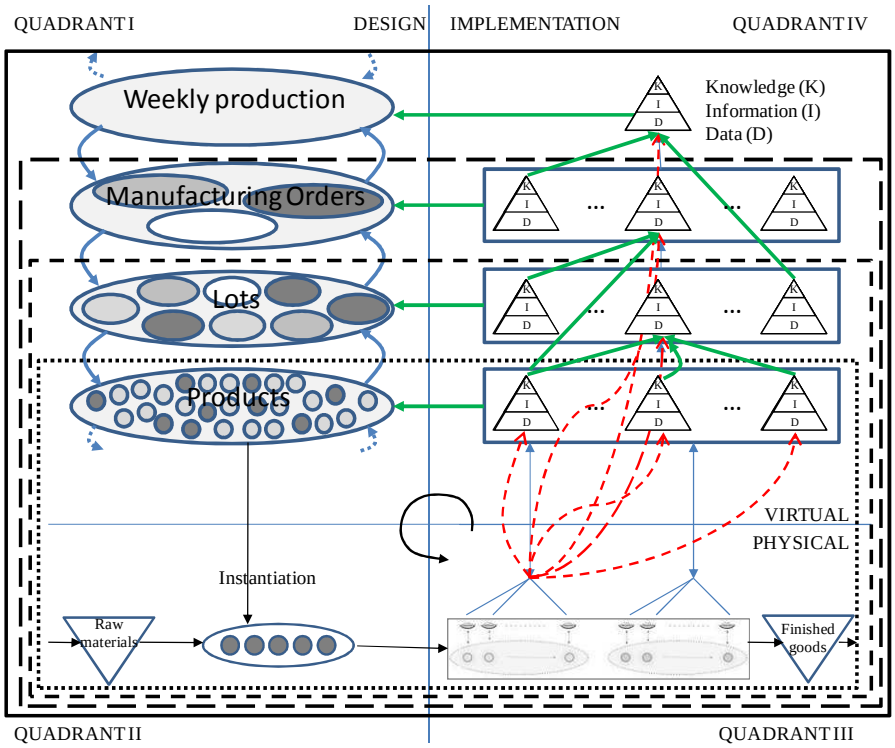


Fig. 1. VSM based product driven control system [11].

Concerning the figure 1 above, we have made the hypothesis that products are instrumented with RFID technology which allows acquiring, archiving data, and communicating and interacting with their environment. The products holarchy is designed in order to represent the decision levels of a MRP² system. This resulting metamodel uses the holon concept. This figure is subdivided into four quadrants (I, II, III, and IV) in order to simplify the explanations. The horizontal axis distinguishes the physical world to the virtual one. The vertical axis distinguishes the design phase which is a representation of the system to the implementation phase which performs the decision making and the knowledge management. Red arrows (dashed) represent data flowing from the shop floor to informational system, i.e. the quadrant IV (Data Management System – DMS), when the green (bold) arrows represent informational or knowledge flows into the DMS. To avoid overloading the figure, only some examples of these flows are shown.

The quadrant I shows the planning system which may be centralized or distributed. Its decomposition is based on four levels of aggregation of product entities (weekly

production, manufacturing orders, lots and products). To each level, entities are modeled as agents. The product entities are agents with a specific control/autonomy level which allow to represent all the hybridization levels of the system from a pure centralized system (product agents transmit information to upper level where decision is made) to a pure heterachical system (agents communicate among themselves in order to make decision). The quadrant II is a conceptual representation of the instantiation phase. It shows the physical implementation and corresponds to the instanciation of products into holons, which lead them to become intelligent and possess functions allowing them to interact with environment and to acquire the desired level of autonomy. The quadrant III shows the physical system in which product holons are able to react according to events impacting their own evolution. The quadrant IV shows the virtual implementation and corresponds to the transformation process of data (coming from the shopfloor) to information allowing knowledge to emerge. It is this knowledge which must be loaded by agents in quadrant I in order to improve their adaptation abilities to events with the principle of experience feedback. The question that needs to be answered is "How perform this experience feedback?"

4 DATA MINING AND PDS

As previously said, in the concept of product driven system (PDS), product must take decisions and interact with its environment thanks to acquired knowledge and information. The synchronization of physical and informational flows inherent to PDS implies that many data may be exploited in order to create this knowledge and this information. These data may be related to product himself, or to the production process or resources. So, the question becomes: how to exploit these data?

Considering figure 1, green arrows (thick) connect the elements of quadrant IV to elements of quadrant I. These connections represent the knowledge loaded in agents. So, the first task is to determine which knowledge is needed by different agent levels (obviously, a product agent doesn't need the same knowledge than a manufacturing order agent). So it is necessary to define precisely these needs in order to be able to answer them. When this is done, it remains to determine how to build this knowledge. For this, the recursion property of the VSM model may be useful. A first level may be defined which includes the two physical quadrants (II and III) and the product layer of the two virtual quadrants (I and IV). This level is surrounded by dotted line. The second level (lots) includes the first level and adds the lots layer of quadrants I and IV. It is surrounded by short dashed line. The third level (manufacturing order level) includes lots level and adds the manufacturing order one of the quadrants I and IV. It is surrounded by long dashed line. At last, the fourth and last level (weekly production level) includes manufacturing order level and adds the weekly production layer of the two virtual quadrants (I and IV). It is surrounded by solid line. This decomposition is comparable to the concepts of systems and sub-systems of the system engineering which may be used in order to define interfaces between levels.

Two approaches may be exploited in order to extract knowledge. The first one is to extract knowledge from expert interviews. The second one is to extract knowledge

directly from data. These two approaches may be mixed. We focus here on the second approach. To each level, the knowledge building process is performed in several steps:

1. Determination of which knowledge must be loaded in agent. As explained previously, this need is different according to agent level.
2. Determination and collect of available information in order to produce knowledge. This point will be detailed afterwards.
3. The structure of the model (here multilayer perceptron) has to be determined.
4. At last, learning and validation phase must be performed. The failure of the learning process implies a feedback on the second or third points.

Let us focus on the product level. In quadrant IV, the knowledge to load in the agent must to be built. The main particularity of this level is that the entries of knowledge building process are only data collected on the workshop. This process is then a classical process of knowledge extraction from data. The main difficulty is to know which data is necessary. It can be noticed that these data may be of continuous or discrete nature, determinist or stochastic one, and the knowledge design process must take into account the hybrid nature of this data.

When the next level is considered, the entries of knowledge building process may be data (aggregated or not) collected on the workshop, but also information and knowledge built at the product level. So the double challenge is:

- To determine which data, but also which information and which knowledge at the product level are necessary in order to built the knowledge to be loaded in agent at the lot level,
- To define a tool allowing to aggregate entities of different nature (data, information, knowledge) in order to built the desired knowledge.

The desired knowledge to be loaded in agents of upper levels will be built with a recursive approach by using the preceding procedure. This paper highlights knowledge building process at the product level in order to point out the main difficulties encountered and to propose solutions.

5 ILLUSTRATION

Let us consider a simple production process constituted of sequential work centers presented by figure 2. One of these work centers is a bottleneck. To simplify the model and the intelligent product behavior, the only knowledge which must be loaded in product agents is the lead time between the manufacturing order release and the products arrival into input queue of bottleneck.

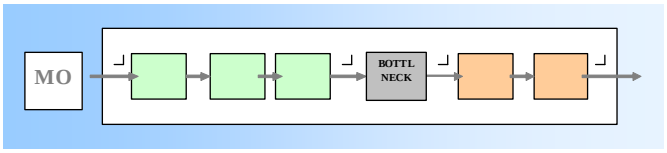


Fig. 2. Considered production system.

The knowledge building process must be highly automated. So, the data loaded by holons (quadrants II and III) are collected and exploited by using multilayer perceptron which uses supervised learning. Its structure is given by:

$$z = \sum_{i=1}^{n_1} w_i^2 . g\left(\sum_{h=1}^{n_0} w_{ih}^1 . x_h^0 + b_i^1 \right) + b \quad (1)$$

where x_h^0 are the n_0 inputs of the network, w_{ih}^1 are the weights connecting the input layer to the hidden layer, b_i^1 are the biases of the hidden neurons, $g(.)$ is the activation function of the hidden neurons (here, the hyperbolic tangent), w_i^2 are the weights connecting the hidden neurons to the output one, b is the bias of the output neuron and z is the network output.

The weights and biases are determined by using a supervised learning which can be performed in two steps [24]:

1. *initialization step*; this initialization may be randomly performed or by using more complex algorithms. This step is crucial in order to avoid local optimum trapping [21].
2. *learning step*; many learning algorithms exist. One of them is the Levenberg-Marquardt algorithm. It works as a hessian algorithm when solution is distant and as a gradient algorithm when solution is near [20].

This neural network must model the lead time between the manufacturing order release and the products arrival into input queue of bottleneck. This lead time is a continuous notion. A first step is to fetch the lead time and all the explanatory variables collected by each product holons. These explanatory variables will become the neural network inputs. They may be continuous, as utilization rates, queues size… or discrete, as routing choice, machine choice… The question that needs to be answered is "How to take into account these discrete data"? Previous works have shown that some discrete variables may be used without particular precautions but other variables may not [24]. In order to solve this problem, two approaches may be used.

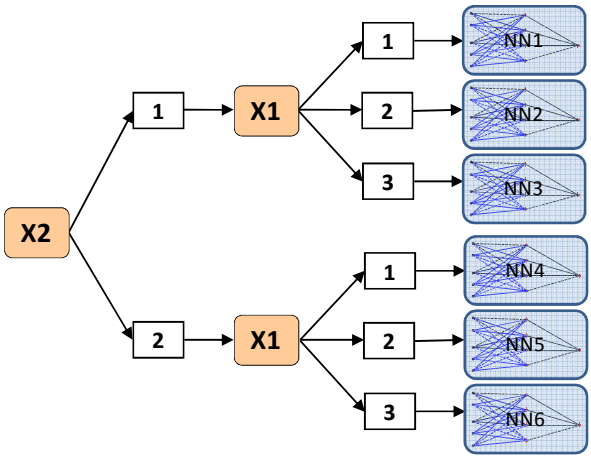


Fig. 3. Taken into account discrete variables – multi-model approach.

The first approach is similar to the multi model philosophy. In fact, if learning cannot take into account some discrete variables, this is due to the system behavior changes when these variables change of state. So, these discrete variables define different operating areas of the system, and so, it is necessary to design one neural model for each operating area. As example, if two discrete variables X1 and X2 may take 3 and 2 states respectively, the considered system can be found into 2*3=6 operating areas and so, it needs 6 neural models to learn (figure 3). The advantages of this approach are that the neural networks have only continuous inputs and so, the learning is much simpler. Moreover, the structures of the networks include less input and hidden neurons and so the computational time decreases during the learning and exploitation steps. However, the main drawbacks are the number of neural networks to learn and the need to design a models selection system in function of states of discrete variables [24].

The second approach consists to transform the discrete variables into binary ones and to use these binary variables as inputs of the network. With the same example as above, five binary input variables must be created, each of them may take the states 0 and 1: “X2=1”; “X2=2”; “X1=1”; “X1=2”; “X1=3”. The main advantage of this approach is that only one neural network models the entire system. However, this model includes more inputs and hidden neurons and so the computational times increase during the learning and exploitation steps [23]. These two approaches are not paradoxical. An optimal solution may be to mix these two approaches in order to limit both, the growing number of neural models to learn and the size of each of these models.

Following the explicative variables are collected and defined. In a second step, the structure of the neural network must be designed. The works of Cybenko [5] and Funahashi [8] have proved that a multilayer perceptron with only one hidden layer (using a sigmoïdal activation function) and an output layer (using a linear activation

function) can approximate all nonlinear function with any desired accuracy. However, nothing is said about the number of hidden neurons.

The simplest approach would be to choose a very great number of hidden neurons which permits to obtain the best accuracy. However, we are not in front of a regular approximation problem but in front of a function adjustment to a finite number of points [6]. The risk is to learn the noise and not the function. This risk is called *overfitting*. In order to avoid it, different techniques have been proposed as regularization methods, early stopping or penalty methods. However, the determination of the optimal structure of the network allows to avoid the overfitting and to optimize the calculation times. For this, two approaches exist. The first one is a constructive one where the hidden layer is iteratively built. Another way is to start from a structure including too many hidden neurons and to remove the spurious neurons. The main advantage of this approach is to allow to some algorithms to determine simultaneously the hidden neurons number and the feature selection [7, 10, 17].

Three algorithms have been tested and compared on the lead time model in a sawmill [22]. These three algorithms are OBS [10], N2PFA [17] and the one proposed by Engelbrecht [7].

In the considering example, ten explicative variables are collected in order to model the lead time. These variables are related to:

- product information as:
  - dimensions (diaPB, diaGB, diaMoy and Lg),
  - refererence (ref)
  - production range (RQM, T_piece),
- process variables:
  - input queues (Q_trim, Q_RQM)
  - utilization rates (U-trim)

Ten variables are collected and three of them are discrete ones (RQM, ref and T_piece). On these three variables only RQM needs to be binarised in “RQM=4” and “RQM=5” as explained previously. The two others can be exploided directly as the continuous variables.

Finally, the network model includes eleven inputs. The N2PFA algorithm allows pruning two of them (ref and T_piece) which have no impact on the lead time. The algorithme proposed by Engelbrecht allows finding the good number of hidden neurons very quickly but don’t succeed to prune spurious inputs. OBS is not able to find a good structure.

The obtained results have shown that the association of two algorithms (Engelbrecht algorithm and N2PFA) gives the optimal structure of the network quickly. The computational time is divided by two comparing when just N2PFA is used. In fact, a first step of pruning with the fastest algorithm (Engelbrecht) allows finding the number of hidden neurons. In a second step, the N2PFA algorithm works with a smaller structure than the initial one (less hidden neurons) allows to design the network structure by pruning the spurious inputs.

6 CONCLUSIONS

In this chapter, in a first step, we have summarized the new advances in product driven system approach and a VSM model has been presented. In a second step, we have investigated the knowledge building process. In summary, we can say that data are essential sources of knowledge but they are often unclaimed treasure because their exploitation may become time consuming. However, some tools exist, as learning overall and neural network in particular, allowing automating this exploitation. Nevertheless, we need to go further and to make sure to respond positively to the questions:

i) Is all the necessary information included in the data?
ii) Is all the collected data well necessary?
iii) How is it possible to associate various data (continuous or discrete) in the same model?

If the learning fails, this implies that the answer to the first question is negative. In this case, in order to supplement database, the product holons must collect other variables which may require improving the instrumentation. For the two other questions, some tools and methods have been presented succinctly here.

The learning phase of a neural model is a complex task which must be designed off line. However, the resulting model is a simple equation which may be loaded in an agent.

Moreover, a question remains an open issue: How to perform the knowledge building process for upper levels? This point represents a double challenge:

i) how to determine which data, but also which information and which knowledge at the lower level are necessary?
ii) which tool allows to aggregate entities of different natures (data, information, knowledge)?

These challenges will be addressed in our future works.

7 ACKNOWLEDGEMENTS

This paper is based on many works that have been achieved and supervised by authors. Many thanks to researchers that have worked with us on this subject. Particular thanks to Mr. C. Herrera for its contribution during its works of Ph.D. Moreover, the authors gratefully acknowledge the financial support of the CPER 2007-2013 Competitiveness Fibre Cluster (Structuration du Pôle Compétitivité Fibres Grand'Est), local (Conseil Général des Vosges), regional (Région Lorraine), national (DRRT and FNADT) and European (FEDER) funds.

REFERENCES

1. Babiceanu R. and Chen F. (2006). Developpement and applications of holonic manufacturing systems: a survey. *J. of Int. Manufacturing*, 111-131.

2. Beer S., (1984). The viable system model: Its provenance, development, methodology and pathology. *J. of the Op. Research Society*, 7-25.
3. Costanza J. (1996). *The quantum leap in speed market.* Institut of Technology, Inc.
4. Créput J.C. (2008). *Hybridation de métaheuristiques pour la résolution distribuée de problèmes d'optimisation spatialisés.* Habilitation à diriger les Recherches, Université de Bourgogne.
5. Cybenko G. (1989) Approximation by superposition of a sigmoïdal function. *Mathematics of Control Signals and Systems*, 303-314.
6. Dreyfus G., Martinez J.M., Samuelides M., Gordon M.B., Badran F., Thiria S., Hérault L. (2002). *Réseaux de neurones : Méthodologies et applications.*, Paris, France, Editions Eyrolles.
7. Engelbrecht A.P. (2001) A new pruning heurisitc based on variance analysis of sensitivity information. *IEEE trans. on Neural Networks*, 1386-1399.
8. Funahashi K. (1989) On the approximate realization of continuous mapping by neural networks. *Neural Networks*, 183-192.
9. Goldratt E., Cox J. (1984). *The goal*, Great Barrington, North River Press.
10. Hassibi B., Stork D.G., Wolff G.J. (1993). Optimal brain surgeon and general network pruning. *IEEE Int. Conf. on Neural Networks.* San Francisco, USA, 293-299.
11. Herrera, C. (2011). Proposition d'un cadrage générique de modélisation et de simulation de planifications logistiques dans un contexte de décisions partiellement distribuées. PhD thesis, Nancy Université, 2011.
12. Herrera C., Belmokhtar S., Thomas A. (2011). Viable system model approach for holonic product-driven manufacturing systems. *1st workshop on Service Orientation in Holonic and Multi Agent Manufacturing Control SOHOMA'11*. Paris, France.
13. Krafcik J. (1988). Triumph of the lean production system. *Sloan Management Review*, 41-52.
14. Leitao P. (2009). Agent-based distributed manufacturing control: a state-of-the-art survey. *Eng. Application of Artificial Intelligence*, 979-991.
15. McFarlane D., Sarma S., Chirn J., Wong C., Ashton K. (2003). Auto ID systems and intelligent manufacturing control. *Eng. App. of Artificial Intelligence*, 365-376.
16. Morel G., Panetto H., Zaremba M., Mayer F. (2003). Manufacturing enterprise control and management system engineering: paradigms and open issues. *Annual Reviews in Control*, 199-209.
17. Setiono R., Leow W.K. (2000). Pruned neural networks for regression. *6th Pacific RIM Int. Conf. on Artificial Intelligence PRICAI'00*, Melbourne, Australie, 500-509.
18. Tennant G. (2001) *SIX SIGMA : SPC and TQM in manufacturing and services*, Gower Publishing Ltd.
19. Thomas A. (2004) *De la planification au pilotage pour les chaines logistiques*, Habilitation à diriger les recherches, Nancy Université, 2004.
20. Thomas P., Bloch G. (1996). From batch to recursive outlier-robust identification of non-linear dynamic systems with neural networks. *IEEE International Conference on Neural Networks ICNN'96*, Washington, USA, 178-183.
21. Thomas P., Bloch G. (1997). Initialization of one hidden layer feed-forward neural networks for non-linear system identification. *15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics WC'97*, Berlin, Germany, 295-300.
22. Thomas P., Thomas A. (2008). Sélection de la structure d'un perceptron multicouches pour la réduction d'un modèle de simulation d'une scierie. *5ème Conf. Int. Francophone d'Automatique CIFA'08*, Bucarest, Roumanie.

23. Thomas P., Thomas A. (2009). How deals with discrete data for the reduction of simulation models using neural network. *13th IFAC Symp. on Information Control Problems in Manufacturing INCOM'09*, Moscou, Russie.
24. Thomas P., Thomas A., Suhner M.C. (2011). A neural network for the reduction of a product driven system emulation model. *Prod. Planning and Control*, 767-781.
25. Van Brussel H., Wyns J., Valckenaers P., Bongareerts L., Peeters P. (1998). Reference architecture for holonic manufacturing systems: PROSA. *Computers in Industry*, 255-274.
26. Verstraete P., Valckenaers P., Van Brussel H., Saint Germain B., Hadeli K., Van Belle J. (2008). Towards robust and effcient planning execution. *Engineering Applications of Artificial Intelligence*, 304-314.
27. Vollman T., Berry W., Whybark D. (1997). *Manufacturing planning and control systems.* New-York: McGraw-Hill.
28. Vrba P., Macurek F., Marik V. (2008). Using radio frequency identification in agent-based control systems for industrial applications. *Engineering Applications of Artificial Intelligence*, 331-342.
29. Waldner J.B. (1992), Principles of Computer-Integrated Manufacturing, John Wiley & Sons, ISBN 0-471-93450-X
30. Yoshikawa H. (1995). Manufacturing and the 21st century-intelligent manufacturing systems and the renaissance of the manufacturing industry. *Technological Forecasting and Social Change*, 195-213.