Proceedings of the 1999 IEEE
International Symposium on Assembly and Task Planning
Porto, Portugal - July 1999

DYNAMIC SCHEDULING OF MANUFACTURING ORDERS: A DECISION SUPPORT SYSTEM APPROACH

Ana Almeida①, Carlos Ramos①, Sílvio do Carmo Silva②
email: ana@dei.isep.ipp.pt, csr@dei.isep.ipp.pt, scarmo@ci.uminho.pt
①Instituto Superior de Engenharia do Porto, Dept. Engenharia Informática
Rua São Tomé, 4200 Porto – Portugal, Phone: 351 - 2 – 8340500
②Universidade do Minho, Dept. de Produção e Sistemas
Gualtar, Braga – Portugal, Phone: 351 – 53 – 604100

Abstract

This paper presents a Decision Support System (DSS) based architecture and a new method for the dynamic scheduling of manufacturing systems. The proposed architecture is based on the DSS (Decision Support System) paradigm [Reynolds-88]. The typical scheduling problems, namely the order allocation and order sequencing problems, are to be solved for a given production plan which defines a set of manufacturing orders to be processed. These specify product types to be manufactured, their quantities and due dates. The allocation problem is concerned with the assignment of *n* orders to *m* manufacturing processors, under the given production plan, and the solution to the sequencing problem identifies the input sequence and the instants of processing of the assigned orders on each manufacturing processor

1. Introduction

The scheduling problem [Bowman-59] is a well-known problem for which many contributions have been done. Nevertheless, it is very difficult to find optimum solutions to real world scheduling problems. Moreover most of the existing scheduling methods do not allow interaction with the scheduling operator, usually the decision agent, and do not take advantage of including him or her in the decision cycle. According to Bedworth et al [Bedworth-87] *"...common sense is the best way to scheduling when there is a complex scenery"*. What seems really useful is a tool for supporting decisions to help operators to achieve and contribute for good scheduling. Such a tool must have the capability of leading to good solutions in a short time. For this, scheduling DSS systems with the involvement of the operator can be most adequate. This stems from the fact that once a solution is obtained by some heuristic, it can, most likely, to be improved through a DSS for scheduling. In fact, the information provided by the DSS added to the user knowledge about the behaviour of the manufacturing systems operation and experience in scheduling can be included in the search procedure for good scheduling decisions.

In [Davis-88], [Sprague-89], [Liebowitz-90] and [Chung-94] it is suggested that in areas where the DSS approach has been applied to scheduling, the results are much better when compared with those obtained from other conventional methodologies.

In this work, a Decision Support System architecture, together with a method for scheduling manufacturing orders, considering their deadlines, are proposed to be used in an integrated manner for interactive and iterative scheduling,.

In the resulting DSS system, manufacturing orders are the primary input. They contain information such as the type and number of products to be manufactured and their deadlines. Within a manufacturing order it is also the description of the operations necessary for the manufacturing of the product as well as the type of manufacturing processors necessary to carry them out. To each operation performed in one or more product parts it is associated a processing time. This time may include auxiliary time elements such as set up time and time for parts handling. By performing several operations on parts we will obtain a product (e.g. a chair). These operations may include assembly operations.

2 – Decision Support System approach

2.1 – Definition

A Decision Support System is a comprehensive computer-based system used to help people in reaching decisions about semistructured problems [Reynolds-88]. It is usually used to support decision-making when the problem is too complex to be solved manually or there is too much data or complex calculations to be done. In this case, there is a need for interaction between the decision-maker and the DSS. With the aid of the DSS, which include some methods to enhance decision making, the decision-maker is comfortable in achieving good decisions. The characteristics and underlined reasons for using a DSS totally justify its application to the complex problem of scheduling of orders in a multiple resource manufacturing system environment.

0-7803-5704-3/99/$10.00©1999 IEEE

2.2 – *The Adaptive System*

A DSS can be considered an adaptive system. Simon(1980) describes such a system as one that adapts to changes of several kinds over three time horizons. In the short run, the system allows a search for answers within a relatively narrow scope. In the intermediate time horizon, the system learns by modifying its capabilities and activities, i.e. the scope or domain changes. In the long run, the system evolves to accommodate much different behaviour styles and capabilities.

A DSS gives the decision-maker the capability and flexibility to search, explore and experiment with the problem area, within certain boundaries.

Usually, in scheduling systems of real world problems, the optimal solution cannot be achieved. So one must look for good, not optimal, solutions. One way towards this can be explored through simulation of different scenarios generated by the scheduling method presented in section 4. Over the time it is important that the system learns from the decision maker behaviour, the obtained results, the history and the environment changes.

2.3 – *Easy to use Interface*

One of the most important factors that determine the success, usefulness and flexibility of a DSS it is the user system interface. This should be intelligent and user-friendly. The term user-friendly is usually used to describe a system that is easy for a non-computer-user to use with almost no training. In a DSS based scheduling interaction between DSS and decision-maker must be provided to give the possibility to introduce changes to automatically generated schedules and provide for user responsibility and some user decision making autonomy. An intelligent user-friendly interface shall be used in order to easily present all the necessary information, letting the user to easily access and manipulate it and perform the changes on schedules he or she considers necessary.

3 – Proposed Architecture

The scheduling of orders should be accomplished before and during the manufacturing period. Therefore the release dates of the orders are variable. Several orders arrive to the system at different times, some leave the system because they have been finished and still some others may be cancelled. This means that the scheduling decisions have to be taken constantly. This type of scheduling is identified as *Dynamic Scheduling*, must incorporate information on-line in real time and it must allow the adjustment of data and schedules whenever necessary.

According to Reynolds [Reynolds-88] a DSS consists of four main components. These include an easy to use interface, a model management subsystem that enables the user to specify one or more models to simulate the real world, a data management subsystem that provides access to both internal and external data and the data needed to support the decision-making process. In our system we follow this structure as shown in figure 1.

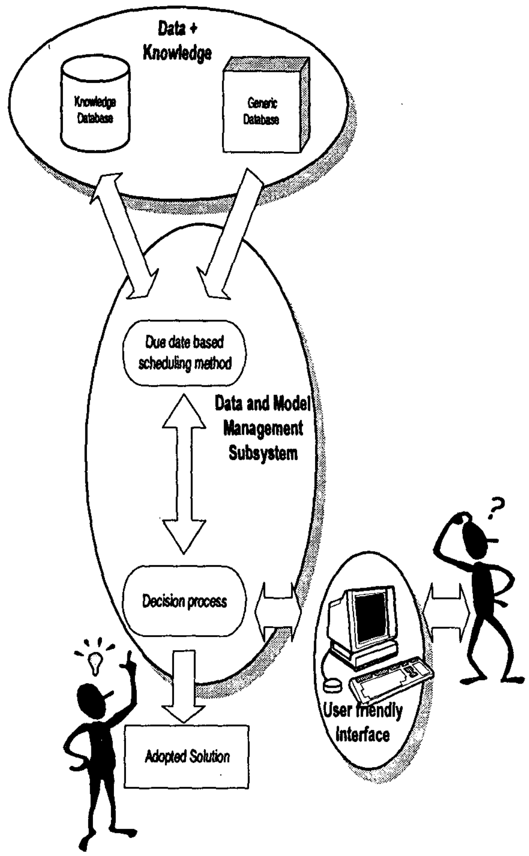

Figure 1 Configuration of the Scheduling Decision Support System

3.1 – *Data + knowledge*

The Knowledge Database supports the decision making in respect to the manufacturing resources choice and the scheduling policy adopted. This may include batching, order aggregation for equal products, JIT approach, batch splitting, batch overlapping and simultaneous manufacturing of identical or different product types.

The Knowledge Database is created over the information collection of previous situations and obtained results enabling the user to study more scheduling alternatives than time would permit if the analysis were done manually. It contributes for better and faster decision making in many situations.

A Generic Database provides the needed information for scheduling. It is a big collection of data, internal and external to the shop floor. There is information related with products, processes, manufacturing resources, which may be production lines, machines and tools, and also related with environment variables, changing factors, etc. A wide variety of data can be accessed and manipulated in order to make the necessary calculations to generate the scheduling results.

3.2 – Data and model management subsystem - decision making process

The Data and Model Management Subsystem contributes directly to the decision making process. Making a decision means to choose among several alternative solutions the one to adopt, assuming all the responsibilities of that choice.

There are some important aspects that need to be considered in creating a DSS. The balance between autonomy and user interaction is one of the most critical aspects in the real use of this kind of systems. In respect to production planning and scheduling important knowledge and expertise tend to remain with the decision-maker for a long time. Therefore a complete autonomous system, without any possible interaction with the decision-maker, tend to be useless. In order to achieve the best results the system must interact with the user accepting suggestions for improvement of the automatically generated solutions. The decision-maker uses the DSS as a tool to improve the decision making process, noticing that he or she is the ultimate responsible for the good or bad scheduling decisions.

A dynamic environment is characterised by high variation on working conditions and on working requirements through time. These include variation of available resources, and of available product mix. This is due to, for example, new order releases, unexpected changes in processing time or order cancellation. Variation of resources may be due to machines breakdown, tool breakage or absenteeism.

Under this dynamic environment a due date based scheduling method is used to do a difficult part of the scheduling job. In fact we consider this method to be the core of the scheduling DSS.

The scheduling method picks up information about the work and resources, related with the scheduling problem and suggests possible solutions if they exist. This method verifies the time horizon of the resources associated to each operation and looks for their availability in a relevant time interval. The inclusion of some high level heuristics help in the decision process

3.3 – User friendly interface

The advantages of DSS for scheduling can be enhanced through a good design of an intelligent user-friendly interface for user inter-action. In order to achieve best results the interface should be carefully designed and be the result of a close interaction and co-operation between system developers and its users [Almeida-98].

Such an interface, for a scheduling system, following a DSS approach, must include the following characteristics [Ramos-96]:

a) the input information must be given by means of screens where the definition of operation characteristics and precedence constraints is easy to include,
b) the results obtained should be presented through Gantt charts making easy the visualisation of the work distribution. This offers the user the important function of comparing different scheduling scenarios,
c) a possibility must exist for the user to make the scheduling changes he considers necessary to obtain good solutions.

3.4 – System characteristics

The scheduling system proposed in this paper permits obtaining good solutions in a short time with reduced computational effort.

The system is to be used in a dynamic manner. It allows the continuous updating of data and generates several scheduling solutions from which the user can choose one, found to be the most adequate.

4. Scheduling with deadlines

In this section we will describe the method used for scheduling orders on manufacturing processors, that considers deadlines. This method is based on previous work developed by Ramos et al [Ramos-95]. It involves two phases: *the forward influence phase* and the *backward influence phase*. Algorithms for these phases are presented and explained.

The method allows dealing with situations in which several operations on different parts of the same job or order can be carried out simultaneously, in the manufacturing system. The processing plan of an order is represented in a graph [Rocha-94], where the nodes correspond to operations and the arcs indicate preceding operations. Figure 2

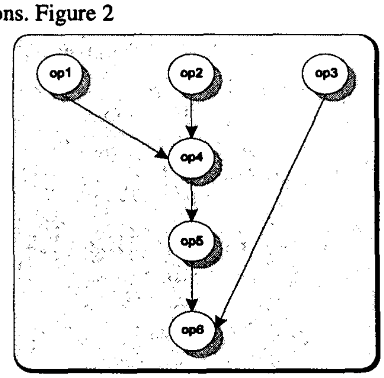

Figure 2 Example of a process planning precedence graph.

4.1. Basic concepts of the method: Behaviours and Agendas

In the scheduling method, the concept of Agenda is important. It represents the time intervals in which each manufacturing processor (MP) is free. Thus we can easily know the time intervals in which a MP is in use. In an Agenda each time interval i is represented in the following form: (tin_i, tfin_i)

A Scheduling Behaviour (SB) is a resource unrestricted late start schedule of all operations of a single part or of a single product made of several parts, depending on what is to be schedule if orders of parts or orders of products.

A SB can be associated with the precedence graph of figure 2. Figure 3 illustrates the SB for the manufacturing of a product whose process plan is represented in figure 2. For this product six types of operations, on several parts, are necessary. The op1 needs 3 *tu* (time unit) in MP1, op2 needs 2 *tu* in MP2, op3 needs 2 *tu* in MP3, op4 needs 4 *tu* in MP4, op5 needs 2 *tu* in MP5 and op6 needs 3 *tu* in MP6, resulting in a total manufacturing time of 12 time units for a single product. However for the manufacturing of N products it is not necessary a manufacturing time of N*12 *tu.* In fact, after the first product has been manufactured it is possible to manufacture a new product every 4 *tu.* For manufacturing 5 products we spend only 28 *tu.* op4 is the critical of operation. It creates a bottleneck in MP4 and sets the production rate.

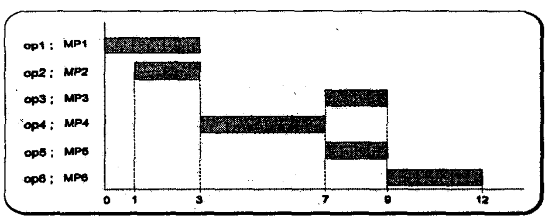

Figure 3 Scheduling Behaviour of the product

Supposing that we intend to manufacture a order of 5 products within a deadline, t_d, of 70 tu, being the beginning of processing time, t_i, at instant 0, we have: $N = 5$, $t_d = 70$ and $t_i = 0$

From the processing times Δtp_i and the waiting times Δtw_i for each operation on the SB, it is possible to determine the scheduling times Δtoc_i, at the manufacturing processors, as well as the time intervals, $\Delta tin_{i,j}$ between the starting of any two operations i and j, and also the time intervals $\Delta tfin_{i,j}$ between the finishing of any two operations i and j.

The scheduling times at the machines, for the manufacture of N products are given by:

$$\Delta toc_i = N x \Delta tp_i + (N-1) x \Delta tw_i$$

Δtoc_i is the minimum free time interval needed to schedule the operations i of an order of N products.

In order to support the description of the scheduling algorithm let us consider the situation shown in figure 4, where the MP free times, in the interval [0, 70], are represented by their agendas:

$Ag1_{(0,70)} = [(0,8), (15,45), (50,65)]$

$Ag2_{(0,70)} = [(0,13),(18,43),(49,60),(65,70)]$

$Ag3_{(0,70)} = [(6,20), (23,45), (56,70)]$

$Ag4_{(0,70)} = [(0,8), (19,41), (50,70)]$

$Ag5_{(0,70)} = [(0,16), (24,50), (60,70)]$

$Ag6_{(0,70)} = [(0,10), (25,53), (62,70)]$

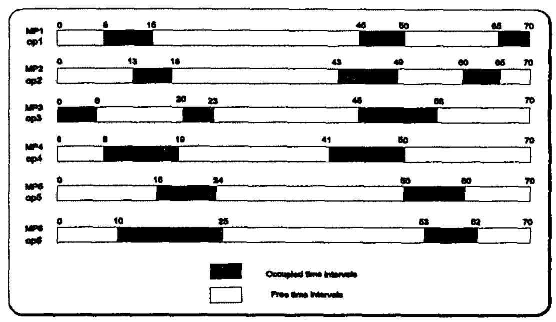

Figure 4 Initial Agendas of Manufacturing Processors.

Before we determine a feasible schedule for the production of the N products in the system let us develop first the due date based scheduling method subject to agendas.

4.2. Due date based scheduling algorithm

Considering Scheduling Behaviours (SB) and Agendas we will explain the basic concepts of the scheduling method.

As referred previously (section 4), the method involves two phases, the forward influence phase and the backward influence phase. The first is applied to all the operations and, after this, walking in the inverse direction, the second is applied to all the operations, except to the last one, as described below.

4.2.1. Algorithm steps for the forward influence phase

Step1) Determine the scheduling interval for each set of operations i: (tc_i, td_i)

The scheduling interval for each set of operations is a time interval where the operations should be scheduled.

tc_i is the time where the scheduling interval begins and td_i the time where the scheduling interval ends. The time tc_i is, in the case of the first set of operations, the lower bound of the whole scheduling interval, for all sets of operations, and is designated, in this case, as tinf, i. e: tc_i = tinf, for i = first (first set of operations of the manufacturing order chain)

The tc_i value for other than the first set of operations is given by: $tc_i = tinf + \Delta tin_{first,i}$, for $i \neq first$

where $\Delta tin_{first,i}$ is the time interval between the starting of the first set of operations and the starting of the set of operations i in the MP i. This is obtained from the SB above referred (section 4.1).

The time td_i is, in the case of the last set of operations, the upper bound of the of the whole scheduling interval, for all sets of operations and is designated, in this case, as td, meaning also time of the deadline of an order, i. e:

$td_i = td$, for i = last (last set of operations of the chain; td = deadline for the order)

The value of td_i for other than the last set of operations is given by: $td_i = td - \Delta tfin_{i,last}$, for $i \neq last$

where $\Delta tfin_{i,last}$ is the time interval between the finishing of the i^{th} set of operations and the finishing of the last set of operations. This is obtained from the batch behaviour above referred (section 4.1).

Step 2) Take initial Agenda Ag_i *in the scheduling interval* (tc_i, td_i)

Ag_i becomes the list of free time intervals of the MP i, within the scheduling interval (tc_i, td_i)

Step 3) Compute the Agenda Ag_i' *in the scheduling interval*

If there is any influence of a previous set of operations k in the set i under consideration then the new Agenda Ag_i' is obtained from the intersection of Ag_i and Ag_{ki}: $Ag_i' = Ag_i \cap Ag_{ki}$

where Ag_{ki} is the forward influence of operation k on operation i (see step 5).

So Ag_i' is the list of free time intervals in MP i, where the set of operations i may be possible to schedule taking only into consideration the influence of preceding operations.

Notice that we may have more than one set of operations k preceding the set i, in this case the new agenda will be:

$$g_i' = Ag_i \cap Ag_{k_1 i} \cap Ag_{k_2 i} \cap .... \cap Ag_{k_n i}$$

Step 4) Determine Agenda Ag_i'' or $Ag_{forward\_i}$

This is done by deleting from Ag_i' the time intervals with duration smaller than the scheduling time toc_i.(section 4.1). If this were not done, batch pre-emption would be necessary, i.e., not all operations i, on equal number of parts, of the N products, would be made uninterruptedly. Thus:

$$Ag_i'' = Ag_i' \big| \{(ti,tf),\{tf-ti<toc_i\},(\,)\}$$

So Ag_i'' is the agenda with time intervals potentially adequate and capable of accommodating the scheduling time toc_i for the order, i. e., for the N products.

Step 5) Determine forward influence

If a operation i precedes operation j, the forward influence of i on j is obtained by adding Δtin_{ij} to the beginning of the time intervals and $\Delta tfin_{ij}$ to the end of the time intervals of Ag_i''. Thus:

$$Ag_{ij} = Ag_i'' \big| \{(ti,tf),\{\,\},(ti + \Delta tin_{ij},\ tf+\Delta tfin_{ij}\,)\}$$

This Agenda Ag_{ij} is necessary for applying step 3) to operations which follow i.

Step 6) Repeat steps 1 to 5 until the last set of operations has been reached

If the last set of operations has not been reached then return to step 1 to the next operation of the chain

The application of the forward influence phase permits, for the example presented in figure 4 and for the production of 5 products, to obtain the following Ag_i'' or $Ag_{forward\_i}$ agendas:

$Ag_{forward\_1}$ =[(15,45)], $Ag_{forward\_2}$ =[(18,43)],
$Ag_{forward\_3}$ =[(23,45)], $Ag_{forward\_4}$ =[(20,41)],
$Ag_{forward\_5}$ =[(24,43)], $Ag_{forward\_6}$ =[(26,46)],

From the initially available free time intervals the intervals (19,20) in MP4 (43,50) in MP5 and (46,53) in MP6 were eliminated by the forward influence phase

In step 4 we suggested that the time intervals of the agenda $Ag_{forward\_i}$, or Ag_i'' could only potentially be used for scheduling. This is because we do not know, at this stage, the influence caused on scheduling an operation i due to agenda restrictions of resources carrying out succeeding operations. Due to these restrictions, some apparently valid intervals for previous operations may not be feasible. To clearly determine this we must apply the backward influence phase of the method.

4.2.2. Algorithm steps for the backward influence phase

To find which intervals within each Agenda $Ag_{forward\_i}$, can really be used we must determine the backward influence of operations onto Agendas.

For this we can apply the following steps:

Step 1) *Determine backward influence*

If operation j succeeds operations i then subtract Δtin_{ij} from the beginning of the time intervals of agenda Ag_j and subtract $\Delta tfin_{ij}$ from the end of the same intervals.

The backward influence of j on i is expressed by the following agenda:

$$Ag_{ji} = Ag_j \,|\, \{(ti,tf),\{\},(ti-\Delta tin_{ij},tf-\Delta tfin_{ij})\}$$

Step 2) *Settle the feasible intervals of agenda* Ag_i

The agenda of operation i, Ag_i, can now, finally, be settled and is obtained from the intersection of $Ag_{forward\_i}$ with Ag_{ji}

$$Ag_i = Ag_{forward\_i} \cap Ag_{ji}$$

Notice that we may have more than one operation j succeeding operation i. In this case, the backward influence would originate the agenda:

$$Ag_i = Ag_{forward\_i} \cap Ag_{j_1 i} \cap Ag_{j_2 i} \cap \dots \cap Ag_{j_n i}$$

Step 3) *Repeat steps 1 and 2 until backward influence for all operations has been taken into account.*

The application of the backward influence phase permits, for the example presented in figure 4 and for the production of 5 products, to settle Agendas and the feasible intervals for the scheduling of the order. The following Agendas, relevant to scheduling the order, are finally settled:

Ag1 = [(17,37)], Ag2 = [(18,37],
Ag3 = [(24,43)], Ag4 = [(20,41)],
Ag5 = [(24,43)] Ag6 = [(26,46)].

As can be seen only a single feasible interval exists, in each agenda, for scheduling the order within the time interval [0,70]. Therefore, only a possibility exists for scheduling the sets of operations 1, 2, 3, 4, 5 and 6 by the method presented in this paper. This is shown in Figure 5.

As can be realised, the intervals (15,17) and (37,45) of the MP1, (37,43) of MP2 and (43,45) of MP3 were eliminated by the backward influence phase.

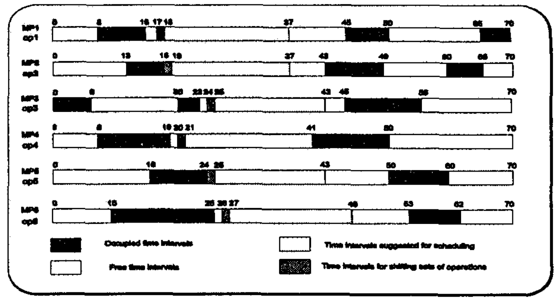

Figure 5 Possibility of Scheduling

It exists at the beginning of each useful time interval, a slack of 1 *tu* in each MP. These slack times can be used to adjust forward or backward the schedule.

The only solution obtained stems from some imposed restrictions on batching. By relaxing batch size, trough batch splitting, for example, it might be possible to take advantage of other free time intervals of manufacturing resources. In this case the split batches of the order could be taken separately for scheduling by the same method and, possibly provide further alternatives to scheduling. The split batch sizes could be chosen by the user, through various attempts or simulations within the framework of the scheduling DSS above referred.

If more than a single set of intervals exist for scheduling, further thinking is necessary to choose among alternative schedules. So, some additional criteria and heuristics can be put on top of the scheduling method just described to arrive to good solutions. It is here, that the user may have the most important role to play in the DSS proposed. However, as in the example, if only a solution exists the decision to accept or reject it still remains with the user of the DSS scheduling system.

If there is no solution, the system can suggest possible actions based on heuristic rules or strategies to help solving the problem This helps the decision-maker to look for an acceptable solution.

4.3- Heuristic rules

To select good solutions some heuristic rules can be used [Almeida-98], such as Minimum Slack Time, Maximum Slack Time, Least Free Time Interval, etc.

When it is not possible to schedule the manufacturing orders within the agendas established by the method previously described, then some strategies must be adopted which can explore the possibility of better use of the free time intervals of the manufacturing resources.

Some of such strategies include:

a) *Batch Splitting.* Through this we can break an order in batches whose sizes can fit the free time intervals of manufacturing resources applying the schedule method based on resource agendas described in this paper

b) *Minimum Delay*. Through this strategy we determine when it is possible to schedule the manufacturing orders with minimum delay in relation to the deadline

c) *Free Time Extension*. This strategy aims at joining together free time intervals of manufacturing resources in order to obtain lager free intervals, where manufacturing orders can be scheduled.

These strategies, and certainly others, give large scope to users for interactive scheduling under a DSS.

5 – Conclusions

In this paper we propose a decision support system based architecture for dynamic scheduling of manufacturing orders. The referred architecture is constituted by three main components: data and knowledge, data and model management subsystem and an easy to use interface.

We have shown that, most probably, the best approach to solve real world dynamic scheduling problems in a multi-order multi-resource environment is to use a scheduling DSS system with balanced distribution of functions between user and DSS. This balancing can be achieved by providing the DSS with good mechanisms for automatic generation of scheduling solutions which, if necessary can be improved through close interaction with the user. Such interaction is important for dealing with complex scheduling situations which can be better tackled taking in consideration user ability and knowledge and the possibility of trying out different heuristics operating strategies or scheduling rules.

The automatic mechanism for generating solutions must be based on effective and efficient methods which can provide answers to user if questions towards obtaining a good scheduling solution.

One such method has been thoroughly described in this paper. This method involves a due date based algorithm that considers manufacturing processors agendas, i. e. their available times.

We can argue that, not only the proposed architecture for a scheduling DSS but also the method for generating solutions through user interaction, can be very successful in complex and dynamic scheduling situations.

One of the strong points of the algorithm used is its capability of, within the free time intervals of the manufacturing processors, pin pointing those intervals where manufacturing orders can be successful and efficiently scheduled.

The inclusion of some high level heuristics and the fact that the system learns form the environment changes, the history, the obtained results and the decision maker behaviour makes the decision process faster and easier.

REFERENCES

[Almeida-98] Almeida, Ana M. N., Praça, Isabel C., Ramos, C., Bragança, A., and Marinho, A. SAPBP:System for Planning and Line Balancing Support 1st IMS, Switzerland, 1998

[Bedworth-87] Bedworth, David D., and Bailey, James E., Integrated Production Control Systems, Management, Analysis, Design 2/E, John Willey & Sons, 1987

[Bowman-59] Bowman, Edward H, The Schedule-Sequencing Problem Operations Research, Vol.1959, 621-624

[Chung-94] Chung, W. C., Tam, M. M. C., Saxena, K. B. C. and Yung, K. L. Evaluatuion of DSS use in Hong Kong Manufacturing Industries Computers in Industry, 21 (3), 307-324, 1994

[Davis-88] Davis, M. W., Applied Decision Support Englewood Clifs, Prentice Hall, 1988

[Liebowitz-90] Lieboiwtz, J. The Dynamics of Decision Support Systems and Expert Systems London Dryden Press, 1990

[Ramos-93] Ramos C. and Oliveira E. CIARC: a multi-agent community for intelligent robotics assembly systems with real-time capabilities, IEEE International Conference on Systems, Man and Cybernetics, pp. 617-622, 1993

[Ramos-95] Ramos C., Almeida, A. and Vale, Z. Scheduling Manufacturing Tasks Considering Due Dates: a new method based in behaviours and agendas. Eighth International Conference on Industrial and Engeneering Applications of Artificial Intelligence and Expert Systems, Australia, 1995.

[Ramos-96] Ramos, C., Bragança, A., Rodrigues, M. F., Marinho, J. A., Almeida, A., and Vale, Z. An Intelligent User Interface for a Decision Support System in Operational Planning of Manufacturing Systems. Advanced Summer Institute of Network Excellence in intelligent Control and Integrated Manufacturing- ASI'96, 1996.

[Reynolds-88] Reynolds, George W., Information Systems for Managers West Publishing Company, 1988

[Rocha-94] Rocha J. and Ramos C. Task planning for flexible and agile manufacturing systems IEEE International Conference on Intelligent Robots and Systems, pp. 105-112, 1994

[Simon-80] Simon, H. Cognitive Science: The Newest Science of Artificial Cognitive Science, 4, 33-36, 1980

[Sprague-89] Sprague, Ralph H., Watson, Hugh J. Decision Support Systems, Putting Theory into Pratice Prentice-hall International, 198