UNIVERSITY OF ZIELONA GÓRA
FACULTY OF MECHANICAL ENGINEERING
in association with the Design Society

THIRD INTERNATIONAL
SEMINAR AND WORKSHOP
10th - 12th October 2002
Zielona Góra - Łagów
POLAND

ENGINEERING DESIGN IN INTEGRATED PRODUCT DEVELOPMENT
Design Methods that Work

ANALYSIS OF CONSISTENCY BETWEEN PRODUCTION ORDER REQUIREMENTS AND RESOURCES AVAILABILITY IN A VIRTUAL ENTERPRISE

Z. A. Banaszak

University of Zielona Góra
Department of Computer Science and Production Management
e-mail: z.banaszak@iiz.uz.zgora.pl

Keywords: workflow, production flow, project-driven production, consistency checking

Abstract: *The paper addresses an issue of consistency checking between a production order requirements and a producer capability aimed at decision-making support for the project-driven virtual enterprises. In order to provide a quick response in a dynamic marketplace the enterprise-spanning workflow and time-restricted resources availability has to be matched with requirements imposed by a given production order. Searching for the feasible solution poses in fact a complex decision making problem that belongs to a class of project scheduling one. Concluding results are summarized within a work order prototyping scheme illustrated on example of a project scheduling.*

1. INTRODUCTION

The objective of this paper is a study of conditions following the consistency between a production order requirements and the given enterprise capabilities. Because of the complex nature of the marked-driven manufacturing systems, a feasibility problem, e.g. of a production order completion within the required time frame, seems to be more frequently faced decision making problem than any kind of optimization one [2]. So, efficient planning and scheduling methods that would balance production tasks and the available manufacturing system capability, and assure quick validation of market demands and react to them through the execution of the production tasks in due time are urgently needed.

Production planning under resources constraints, particularly project planning and scheduling, has received increasing attention that results in the growing rate of publications and widespread commercial packages [4, 8]. The main problem is concerned with the feasible allocation of scare resources to activities over time, i.e. with the schedule in which the capacity of the machines is not exceeded and which meets the overall deadline and the precedence constraints among the operations. In this context The manufacturer's capability can be seen as environment consisting of machines for the given job-dependent speeds, buffers of limited capacity, transportation and material handling means, and other kinds of resource such as energy, manpower, money, and so on. Besides of resources the constraints assumed on their utilization are usually assigned, e.g. throughput, production cost, time-restricted resources availability. In turn, the consumer requirements can be specified in terms of activities involved in a production order. Activities (jobs) usually consist of one or more operations, which do have processing requirements and might be considered to have release dates, due dates and weights. Moreover, they may have precedence relations, and may or may not be split. The relevant constraints regard of the makespan or schedule length, lateness, tardiness, manufacturing overhead, and so on.

In this paper we consider a modeling framework providing a platform for a feasibility problem formulation as well as for a model-checking procedure applied to its solving. An objective is to show how resource-based and causality-based reasoning can be effectively integrated for planning efficiency. Resource reasoning ensures that all the resources needed for the execution of an action are available for allocation [9]. Causal reasoning forces sufficient orderings among actions to achieve the goals and furthermore, determine the extend of concurrency possible in a plan. The proposed model is relevant in many practical situations those could be found in diverse industries such as automobile, home appliance, and software development, as well as in virtual organizations based on small and medium size enterprise networks.

The rest of the paper is organized as follows: Section 2 describes the modelling framework enabling a virtual enterprise prototyping The main problem is formulated in Section 3. In Section 4 an illustrative example of a virtual enterprise prototyping is provided. Some conclusions are presented in Section 5.

2. VIRTUAL ENTERPRISE PROTOTYPING

Virtual enterprises (VEs) are generally defined as a way of organizing business activities, where different and independent partners exploit business opportunity to establish enterprise cooperation. In particular, the process of searching for the appropriate partners is the key to the successful information of the virtual enterprise, which comprises a group of loosely connected firms or service entities possessing certain core competences [5].

The business process in a VE starts with a customer order which in turn has to be decomposed into suborders, which are dispatched to the participants in the VE (that may be treated as subcontractors for a specific part or component). So, the arising problem is how to select a set of partners and how to distribute a customer's order among the partners of a network (e.g., prototype of routes for each order), so as to meet the customer's needs in terms of cost, quality and delivery time as well as enhance support for interoperability between component partners.

In this context an objective of the consistency-checking problem is to determine a feasible schedule of the customer order execution following constraints imposed by a workflow structure and resources availability constraints. In order to solve it any feasible variant (i.e., sub-network of activities) of the enterprise-spanning workflow has to be considered as a potential variant of the order completion. The workflow defines all possible paths through the business process, including the rules that define which paths should be taken and all actions that need to be performed [6]. So, in the case of a VE, the relevant workflow has to consist of many parts (sub-networks) belonging to particular partner-firm-spanning workflows. In that sense a consistency-checking problem can be seen as a kind of multi-project scheduling one [7].

Therefore, the problem of a virtual enterprise prototyping can be seen as two folded, i.e. containing a cross-organizational workflow refinement made on the base of a set of participant enterprise workflows (a feasible, ignoring resources conflicts, causal plan of activities is generated), then post-processing of the feasible workflow to allocate the required resources without altering the causal structure of the plan.

Note an analogy to a project planning. For any customer order (treated as a project) a decision maker has to specify the all project-related information such as the individual tasks, the sequence in which the tasks need to be performed, how much of the tasks can be performed in parallel, and the resources, such as people, vehicles, storages, machines, money, to perform these tasks. The resultant project plan that shows which activity may be accomplished at which resource at which time frame has to match component workflows structure preserving activities and order they have to be executed. Such a sub-network of the set of workflow models can be treated then as a cross-organizational workflow model.

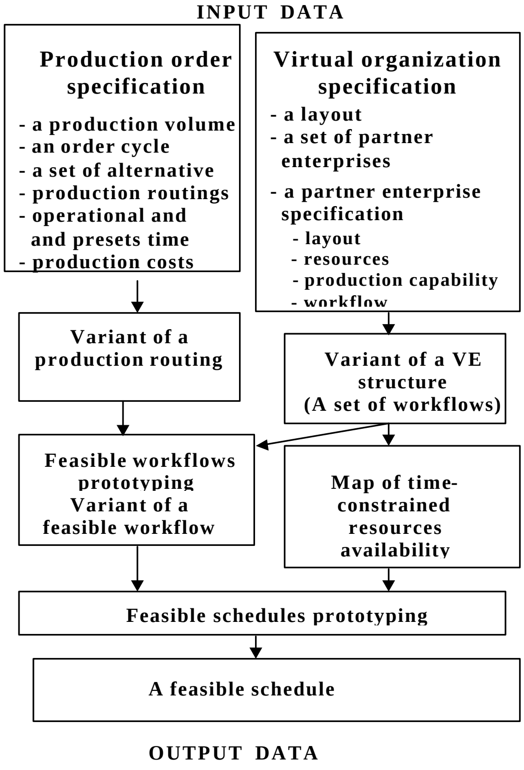


Fig. 1 Scheme of a VE prototyping

The proposed scheme for VEs prototyping is shown in Fig.1. Following this scheme a feasible cross-organizational workflow (a causal plan of activities ordering) is refined for arbitrary selected variants of production routing and VE structure, respectively.

Then the workflow variant obtained is matched with a so-called map of resources availability specifying time periods the particular resources are available as well as amount and inventory cost. In the case a feasible schedule there exists the consistency checking provides a process instance that can be carried out according to a set of values that determines the actual path through the workflow. Such an instance can be treated then as a feasible solution, i.e., a schedule matching the customer order requirements and the VE capability.

3. PROBLEM FORMULATION

Given a virtual organization providing a given production capability while processing some other work orders. So, only a part of the production capability (specified by the resources availability of which is time-restricted) is available for use in the system. It means a set of partner enterprise workflows and a map of resources availability is assumed.

A given production order is specified by an acyclic project network $G = (V,E)$ with fixed processing times assigned to each activity (i,j) ? E, and a given project makespan which is equivalent to a presumed completion time (the work order cycle). Each activity may be executed in one out of the set of $M_{(i,j)}$ modes (system resources). Also, each activity may not be preempted and the mode once selected may not be changed.

The problem considered regards of finding of a makespan-feasible schedule that follows the constraints imposed by the precedence relations and by the time-constrained resources availability.

So, the above mentioned consistency checking problem can be formulated as a kind of the resource-constraint multi-project feasibility problem, and can be mathematically formulated using a model where the objective is to keep one time criterion (mean project delay or multi-project duration) within the required time frame. Constraints ensure that every activity must start once, enforce the precedence relation between activities and that for each resource and each time instant, the resource demand of the activities that are currently in process does not exceed the available capacity of the renewable resources [7]. In general case the problem considered is NP-complete.

4. ILLUSTRATIVE EXAMPLE

The problem considered belongs a class of multi-mode case problems of a project scheduling kind, where finding of a feasible solution is NP-complete [3]. In order to cope with the problem one may consider usage of the model-checking procedure [1]. By model checking we generally mean an algorithmic method by which a desired behavioral property of a system is verified over the model through exhaustive enumeration of all states reachable by the system and the behaviors that traverse through them [1].

Due to this approach some behavioral properties such as activity order preceding, non-overflowing buffers capacity, and makespan including within the required time frame have to be checked comparing with the model representing the production flow. In order to avoid costly exhaustive states enumeration the properties verification are conducted only for states determined by events corresponding to the fork and/or joint type nodes in an activities network.

In order to illustrate the approach proposed let us consider a virtual organization composed of a set of virtual enterprises $\{S_i \mid i=1,\ldots,n\}$ specified by a set of workflows $\{WF_i \mid i=1,\ldots,n\}$. Consider a time horizon t_h and a map of resources availability that specifies the periods the resources are available and their relevant amounts and costs. Assume a variant of the VE structure consists of three VEs and is specified by the workflows shown in Fig.2.

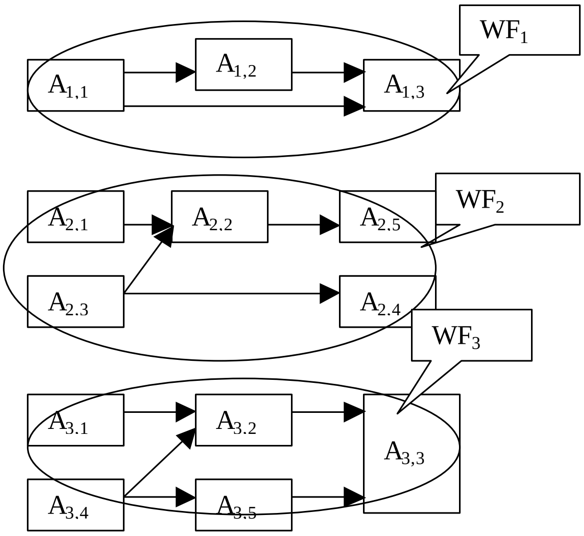


Fig.2 A selected set of workflows

Consider a production order specified by alternative activity networks or production routings (e.g., determining a alternative technologies) shown in Fig.3.

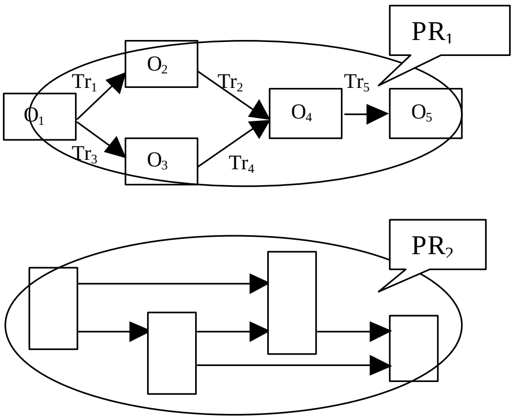


Fig. 3 A set of alternative production routings.

Consider the production routing PR_1 where nodes correspond to activities associated to the fixed enterprises (workplaces), while the arcs encompass activities responsible for transportation of items between workplaces. As a feasible workflow (the way of its refinement is shown in Fig.4) let us assume the activities network shown in Fig.5.

Note that both kind of the PR_1 activities, e.g. manufacturing $\{O_1, O_2.,,,.O_5\}$,and transportation $\{Tr_1, Tr_2,\ldots,Tr_5\}$ are associated with operations responsible for their execution in the particular Ves.

So, in the case considered the workflows WF_1, WF_3 correspond to the goods manufacturing enterprises, and WF_2 encompasses the workflow of a kind of goods transportation company. So, the resultant cross-organizational workflow is composed of subnetworks of the workflows from Fig.1.

.

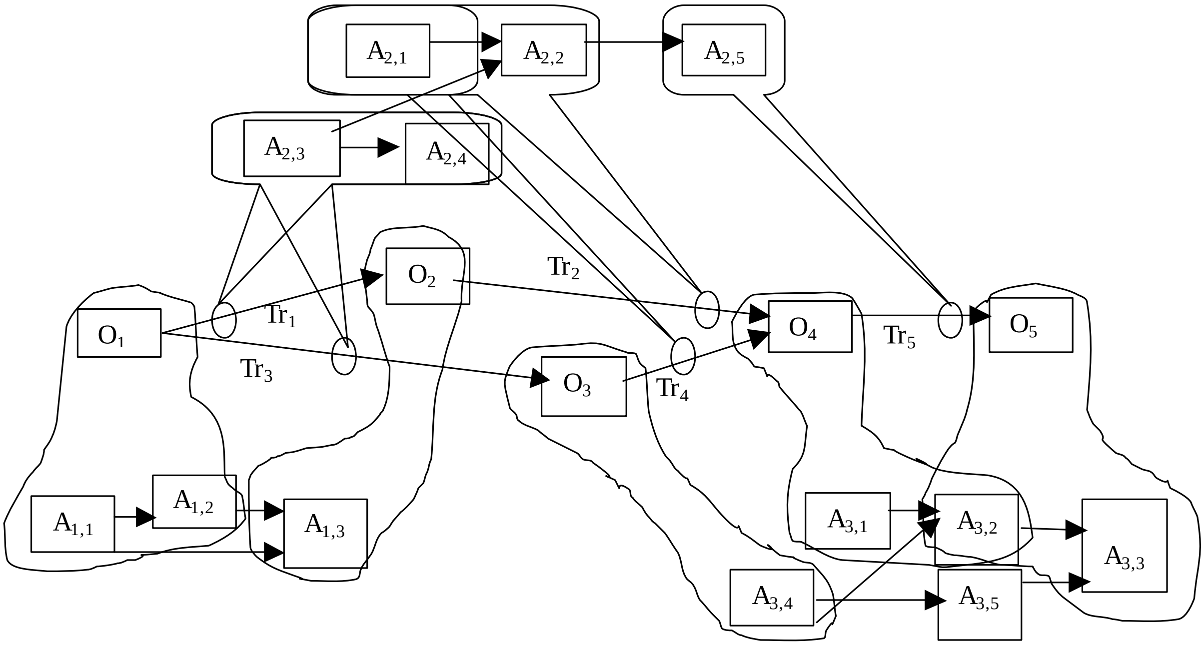


Fig. 4 An example of the cross-organizational workflow refinement

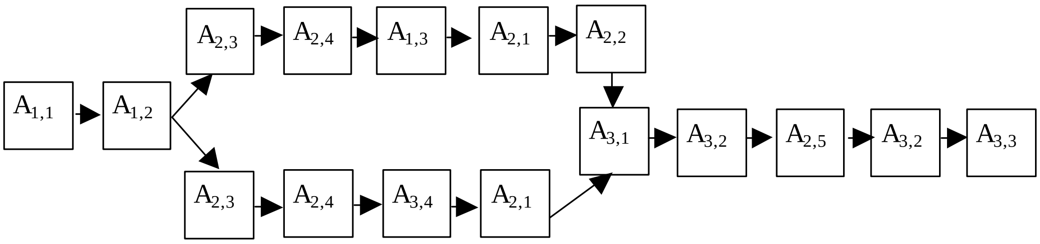


Fig. 5 A variant of the feasible workflow.

Assume the variant of resources allocation to activities of the feasible workflow from Fig. 5 as shown in Table 1. For the sake of simplicity the table does not consist a multi-mode option. Not-blanked cells determine the activity durations.

Due to the Critical Path Method (CPM), the critical path consists of the following activities: $A_{1,1}$- $A_{1,2}$- $A_{2,3}$- $A_{2,4}$- $A_{1,13}$- $A_{2,1}$- $A_{2,2}$- $A_{3,,1}$- $A_{3,,2}$- $A_{2,,5}$- $A_{3,2}$- $A_{3,3}$ and corresponds to the following production routing: R_1- R_2- R_6- R_6- R_1- R_5- R_5- R_3-R_4- R_5- R_4- R_3. The critical path determines the minimum completion time of a project. In the case considered it is equal to 44 units of time.

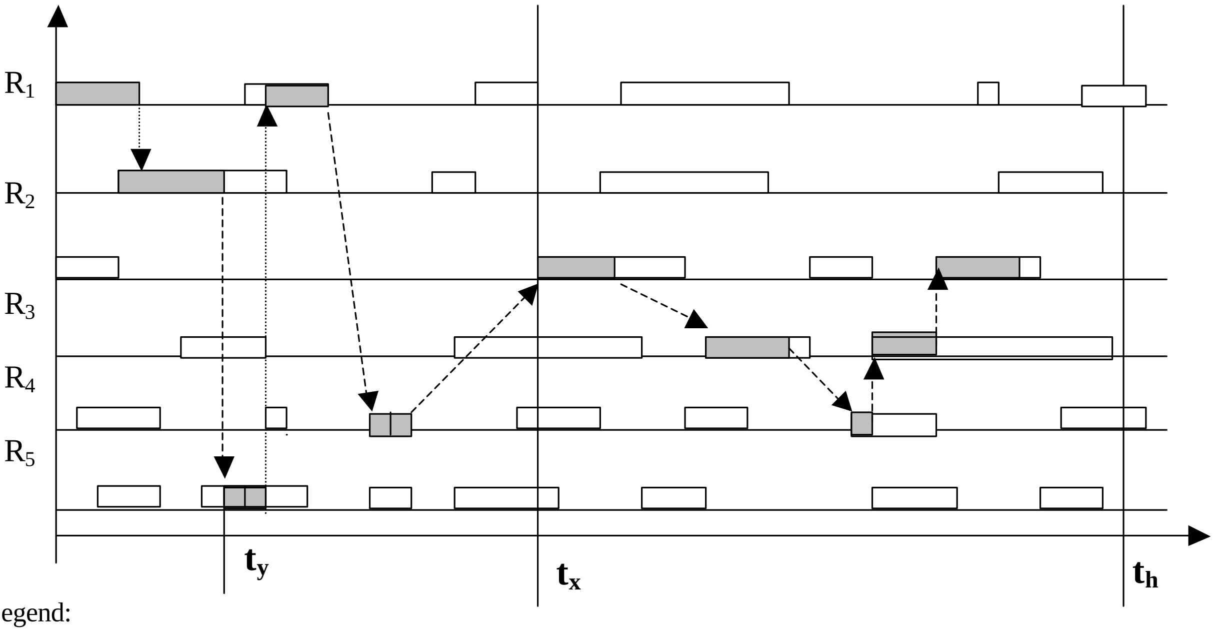
It is assumed that an activity can begin as soon as its predecessor is finished, so the activities on the critical path have no float time, therefore, limited resources must be first assigned to those activities to avoid project delay.

Because each feasible workflow-based solution has to be checked whether the makespan consists within the target time frame while satisfying all the precedence and capacity constraints the critical path has to be checked, too.

In order to take into account the time constrained resources availability let as consider the Gantt's chart from Fig. 6. The shadow bars on the Gantt's chart depict the critical path. The dashed arrows emphasize the feasible solution.

Table 1 *A variant of resources allocation to the network activities.*

Resources	**Activities**											
	$A_{1,1}$	**$A_{1,2}$**	**$A_{1,3}$**	**$A_{3,1}$**	**$A_{3,2}$**	**$A_{3,3}$**	**$A_{3,4}$**	**$A_{2,1}$**	**$A_{2,2}$**	**$A_{2,3}$**	**$A_{2,4}$**	**$A_{2,5}$**
R_1	5		4									
R_2		6										
R_3				5		5	5					
R_4					6							
R_5								1	2			1
R_6										1	2	



Legend:
R_i – the i-th resource
t_h , t_y, t_x – moments determining the end of time horizon, the end of activity $A_{1,2}$, and the beginning of the activity $A'_{3,1}$
▭ - the time slot the i-th resource is available within the time horizon t_h

Fig. 6 Diagram of time-restricted system resources availability.

So, the assumed resources capability of the VE within the given time horizon t_h, is as shown in Fig.6. Checking the time slots each workplace (resource) is available within the time horizon and comparing to the activity times required the model-checking procedure has to start.

Step 1. Determine the critical path and project length T1 without concerning the resource constraint and check whether it does not extend the assumed time t_h, i.e., T_1 ? t_h. Then obtain the feasible (taking into account resources availability constraints) schedule and the relevant project length T_2. Check whether T_2 ? t_h holds.

In the case considered the critical path can be executed due to the assumed resources constraint, see the shadow bars in Fig. 6. So, the same procedure has to repeats for any path linking the joint and fork nodes of the activities network.

Step. 2 Determine the path linking fork and joint nodes (first of all laying on the critical path determined, and then on paths examined), and check the resources availability within the period determined by the moments of the fork activity completion and the beginning of the joint activity. In the case a feasible schedule there exists repeat the Step 2 till the last unchecked path; else feasible schedule does not exist.

In the consider case the state determined by the event corresponding to the path $A_{2,3}$- $A_{2,4}$- $A_{3,4}$- $A_{2,1}$ and to the production routing R_6- R_6- R_3- R_4 has to be scheduled within the period of time determined by moments t_y and t_x. For this state the time slots determining the resources availability from the routing R_6- R_6- R_3- R_4 has to be checked. In the case considered the time slots available on R_3 are not enough to complete the path within the period t_y - t_x. So, the procedure has to repeats for another (if any) variant of the critical path activities allocation. The dashed arrows in Fig. 7 depict the order of the critical path, the another kind of arrows emphasize the feasible solution for the fork-joint path examined.

Note that for each state a consistency (balance condition) between the required and available amount of resources and/or their availability periods is checked. Therefore, the resultant searching procedure may be though as a time-critical resource driven, i.e. focused on a kind of hierarchy of system resources limits.

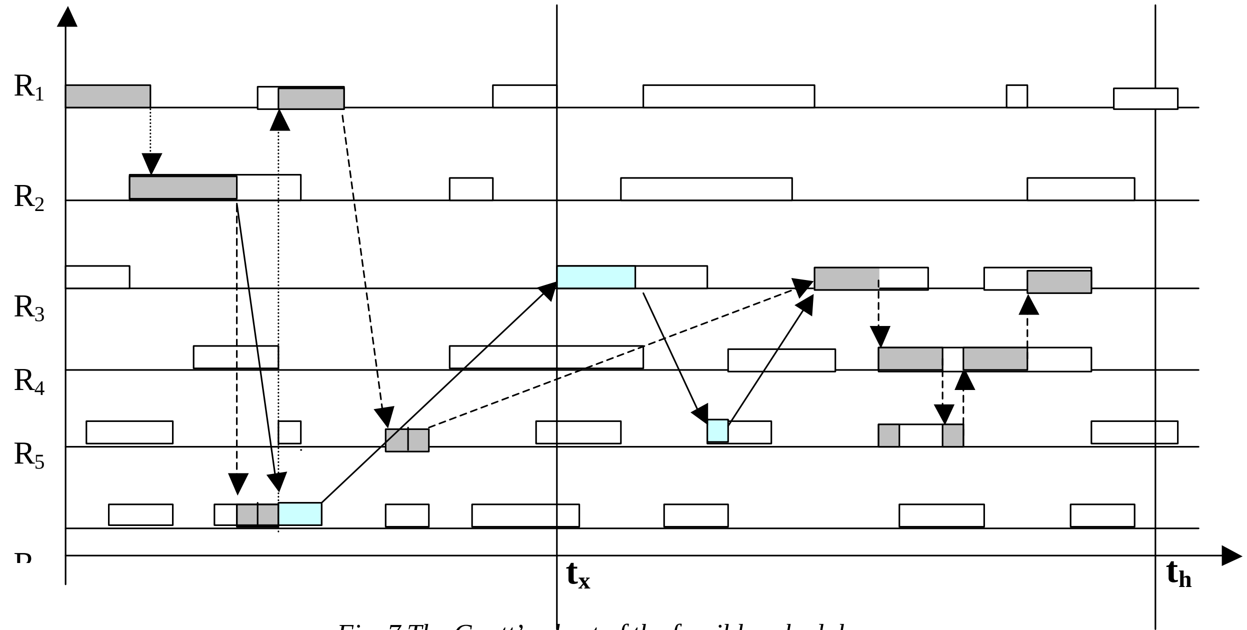


Fig. 7 The Gantt's chart of the feasible schedule.

5. CONCLUDING REMARKS

A modeling framework supporting decision making systems design, which in turn are aimed at consistency checking between a given production order requirements and a VE capability offered is considered. The approach proposed seems to be useful for the project-driven production flow management applied in a kind of make-to-order companies as well as in the virtual enterprises. It provides a good platform for treating the planning and scheduling phases of VEs prototyping as loosely coupled stages. Besides of the causal and resource reasoning the model-checking scheme seems to be promising in on-line decision making whether a production order completion time and manufacturing overhead matches constraints imposed by the enterprise capability and constraints imposed by other products manufacturing processes within assumed horizon time.

References

[1] J.M. Atlee, and J. Gannon, *State-based model checking of event-driven system requirements, IEEE* Trans. Software Eng., vol. 19, no.1, Jan. (1993), pp. 22-40

[2] P. Brucker., et al., *Resource-constrained project scheduling: Notation, classification, models and methods*. European Journal of Operational Research, 112, 1999, pp.3-41.

[3] P.Grefen et al., *CrossFlow: cross-organizational workflow management in dynamic virtual enterprises*. Int. J. Comput. Syst. Sci & Eng (2000), 5, 277-290.

[4] H. Khamooshi, *Dynamic Priority-Dynamic Programming Scheduling Method* $(DP)^2$*SM: a dynamic approach to resource constraint project scheduling.* Int.J.of Project Managmenet, (1999), Vol.17, No.6, pp.383-391.

[5] H. C. W. Lau, and E. T. T. Wong, *Partner selection and information infrastructure of a virtual enterprise network*. Int. J. Computer Integrated Manufacturing, (2001), Vol.14, No.2, pp. 186-193.

[6] F. Leymann, and D. Roller, Production workflow *Concepts and techniques*. Prentice-Hall, Inc. New Jersey, 2000.

[7] A. Lova, and P. Tormos, *Multi-project scheduling: heuristics based on random sampling.* The 18th Int. Conference on CARS&FOF, July 3-5, 2002, Porto, Portugal, Vol.1, pp. 365- 374.

[8] C. C. Wei, P.-H. Liu, and Y.-C. Tsai, *Resource-constrained project management using enhanced theory of constraint.* Int.j of Project Management, (2002), Vol.20, No.2, pp.561-567.

[9] B. Srivastava, S. Kambhampati, and M.B. Do, *Planning the project management way: Efficient planning by effective integration of causal and resource reasoning in RealPlan*. Artificial Intelligence (2001), No.131, pp.73-134.