Assoc.Prof. Bozena Skolud
Department of Engineering Processes Automation and Integrated Manufacturing Systems, Silesian University of Technology

PROJECT DRIVEN CAPACITY PLANNING IN SMALL AND MEDIUM SIZE ENTERPRISES

In the context of customer expectations concerned with new product quality and delivery dates, time and capacity management are the matters of short-term and medium-term planning. In this paper the problem of capacity management of both multi-assortment repetitive production (in steady state and synchronisation mode) and project driven management is presented. In both cases constraints oriented methods are proposed.

1. INTRODUCTION

In a competitive market small and medium enterprises (SMEs) should react to client or potential client expectations as fast as possible because in any enterprise, and especially in those belonging to the class of SMEs the most important capital consists of customer loyalty.

Under such circumstances, enterprises are characterised by single activity connected with very unique product or single – batch production realisation of unique production series, which is never repeated once it is finished. This situation makes it possible for us to suppose that small batch production in SMEs is managed by adopting similar rules as those involved in project management.

In this context planning and control aim to deliver products and order reliably by the specified due date. Time and capacity management are, first and foremost, the matters of short-term and medium-term planning.

Time management is the observation, control and manipulation of time elements. Time elements are the duration of operations, inter-operations times and administration times. The value for a lead-time can be a value based on prior experience and is often not precise enough (because some components are only needed for a later operation). For exact capacity planning we need to know the point in time at which the machine (resource) will be loaded by work to be executed and, thus, a start date for each operation.

Manufacturing lead time is the sum of three different time elements [4]:

- operation time that consists of set-up time and run time dependent upon lot size,
- interoperation time occurs before or after an operation and consists of technical interoperation time and non-technical interoperation time (e.g.: transportation time from the machine to the fictitious center),
- administrative time (time needed to release and complete an order, time for final control, time for preparing the order for shipment).

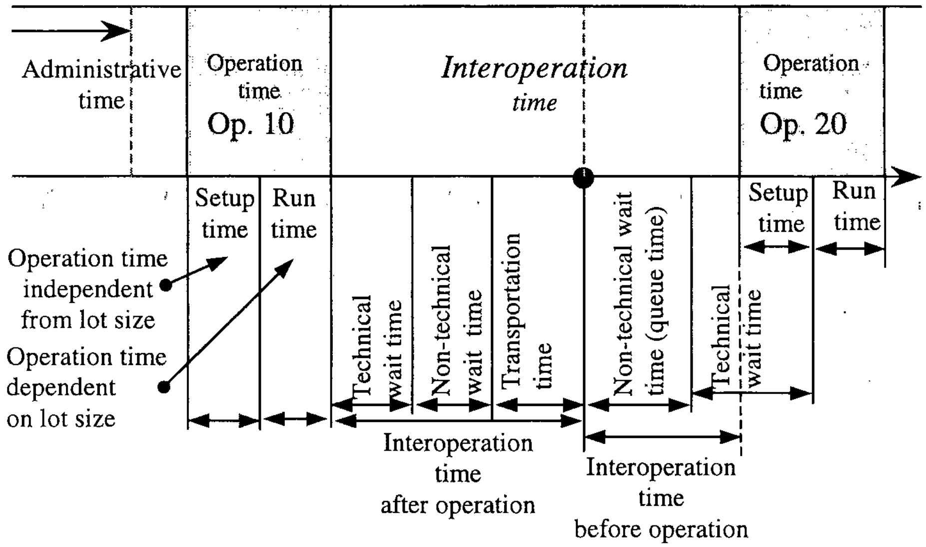


Fig.1. Time structure.

The allocation of resources and their capacities to activities in order to keep the time regime is one of the most difficult tasks.

Case study

Let us focus on a small firm producing and installing electronic equipment (controllers). Controllers differ in inputs and outputs, depending on the application required. Each order is treated individually in consideration of characteristic parameters of installation conditions.

For this case, production organisation depends on the conditions of finished goods distribution and can be achieved as multi-assortment repetitive production in a given, limited planning horizon. The activities related to this approach give an answer to the question if a new production order can be executed under the conditions of given resource constraints and in the expected time and create the dispatching rules controlling the production flow in the system.

On the other hand, the production of one product type can be executed separately in batches. In this case, the same activities can be executed under the condition of finishing the preceding ones, for example: hardware programming can be executed on condition that:

- the state of the object where the installation of a controller is planned is known,
- the controller is assembled.

Other activities may be executed concurrently, i.e. the controller hardware can be assembled at the same time at which the object and its state are identified.

One should remember that, the critical value is due time, contracted with the customer or even enforced by him.

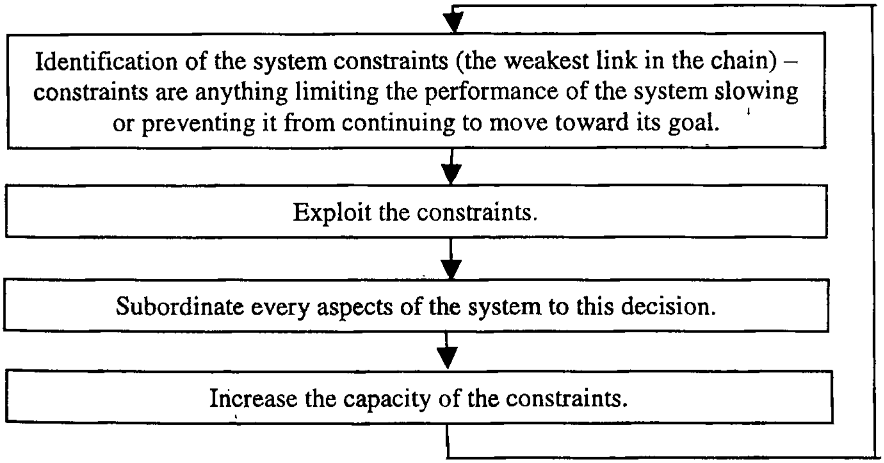
The presented case study presented above leads to the following observation. Management of the client oriented SMEs that realise make-to-order production is project driven. This means performance in accordance with the rules of project management. Each production order, in spite of the product similarity, is different in consideration of product installation conditions.

The paper is organised as follows. The second section focuses on the problem formulation. The concept of the problem solution based on constraints satisfaction and some conditions of multi-project scheduling are presented in the third section. Finally, conclusions and future challenges are discussed in the last section.

2. PROBLEM FORMULATION

Capacity management may refer to the decision of production order acceptance for realisation in the system of simultaneous repetitive production. It may also refer to the acceptance of single or small production order for realisation in the multi-project conditions. In both cases, the production order is one from either multi-assortment or multi-project components and the decision depends on constraints.

One should know that the goal of the system relates to the effects, for example: customer satisfaction or new orders realisation. Having realised the goal of the system and the fact of the constraints, the following steps should be executed as is shown in Fig.2.



Identification of the system constraints (the weakest link in the chain) – constraints are anything limiting the performance of the system slowing or preventing it from continuing to move toward its goal.

Exploit the constraints.

Subordinate every aspects of the system to this decision.

Increase the capacity of the constraints.

Fig.2. Constraints based decision making

The following constraints are identified in the considered system:

1. *Due time.* Expected realisation time
2. *Concurrency.* Admissible (quantity or percentage fraction) simultaneous resources occupation by activities belonging to one project and realised concurrently. A successive activity may be executed (started) before finishing the preceding one. (for example: during object identification, the controller assembly can be done. However, after completing some other phases).

3. *Potential occupancy.* The resource has limited availability in one time unit that can be more than one. Thus, besides "available" and "not available" "party available" is also possible. This partial availability may be explored for another activity of another simultaneously realised project.
4. *Structure of resources occupancy.* In order to realise complex operations the constraints of the minimal admissible occupancy of the resource in unity time can appear. (for example when the assembly operation has to be carried out by two workers).

One of the producer's constraints taken into account is the productivity of the system. This constraint limits the possibility of the acceptance of a production order for realisation in the system. In this context according to [Lova, Maroto, Tormos, 2000], the most dominant problem is to minimise idle resources and minimise realisation time (project duration). Therefore, the problem raises the question: *Is it possible to schedule activities in such a way as to satisfy the constraints that determine the system and client demands under the condition that a new task does not influence those previously accepted for realisation.*

Two different approaches to tasks allocation are presented in Figure 3, in Figure 3a the capacities in the case of "available" and "not available" states of resources are distinguished and in Fig 3b additionally the "partly available" state is possible.

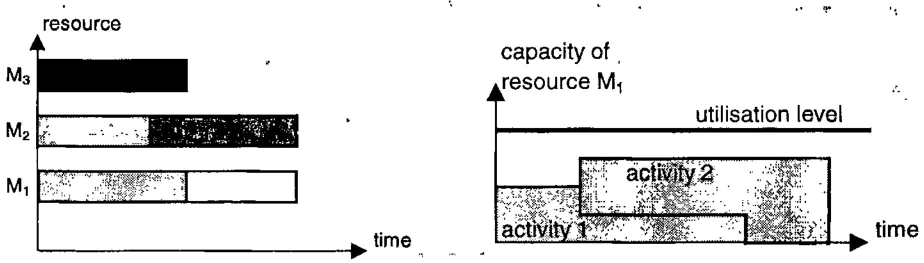


Figure 3 a) sequence scheduling b) multi-project scheduling (concurrently realised operations)

Additionally, in the case of multi-assortment repetitive production (section 3) and the case of multi-project realisation case (section 4), two approaches are distinguished. The first one is called "all together", where all activities are scheduled at the same time. The second one is called "successive scheduling", where new production tasks (production orders or project) are scheduled under the condition that they do not disturb the processes already accepted and realised.

3. REPETITIVE PRODUCTION

3.1. Multi-assortment repetitive production

The necessity of satisfying clients demands and the general tendency to limit work-in-process, result in the need for simultaneous manufacturing of elements in small batches but in a large number of repetitive supply operations. This means that autonomic processes co-operate and the sequences of operations are repetitive in each process. It is more efficient to prepare a cyclical schedule because of the simplicity of the structure of repetitiveness, which is especially important in controlling the system. The processes running in the system share machines and compete for access to the resource. It is important to synchronise these processes in the system, assuring the expected level of the system operation. The idea underlying the Optimal Production Technology (OPT) [5; 6] is to provide the maximum number of bottlenecks in order to maximise the resource utilisation and to enhance the throughput.

As a consequence of the bottleneck occurrences, the material flow has a rhythmic character, specified by a cycle of the bottleneck constraints. This observation implies the assumption of a steady state periodic behaviour imposed on the whole production system.

Repetitive production means that for every constant period T, the same sequence of operations is executed. Period T is determined by a sequence of the accesses of the operations on the shared resources, according to the dispatching rule [8].

Local dispatching rule $R_i = (Pa_1, Pa_2, ..., Pa_j, ..., Pa_n)$ determines the number and the sequence of processes executed in the i-th resource, where:

Pa_j – process,
$i \in \{1,2,...,m\}$,
$a_j \in \{1,2,...,n\}$.

In turn, batch sizes are limited by delivery constraints, i.e. by delivery period and delivery batch size. So, the performance evaluation measures such as the cycle time, the rate of system resource utilisation, possibility of due time realisation of production orders can be seen as a function of the above mentioned constraints. These parameters can also be precisely calculated.

3.2 Synchronisation

In a system of repetitive production mentioned earlier, however, the steady state should be preceded by the starting-up phase. The omission of either the starting procedure (initial production flow), or satisfaction of the starting condition (allocation of elements in different production phase in intermediate buffers) can lead to undesirable situations, e.g. deadlock appearance or synchronisation with an undesirable cycle.

The problem of the system synchronisation motivates the consideration of construction of dispatching rules allocated locally to the system resources, called “meta-rule” [9].

META RULE {[starting-up rule], [dispatching rule], [cease rule]}

The first part of the rule is the starting-up, executed one time only and assuring the synchronisation of the system with expected (desirable) cycle. The second one is executed repetitively and it guarantees steady- state behaviour of the system. The 3-th one is the procedure of the production cease. The most important is that the functioning dispatching rule should cause the self-synchronisation of the system according to the expected cyclical behaviour and bring the whole production to the end without the deadlock appearances.

The distributed control is taken into consideration. It means that resources must not communicate with each other. Hence, it is necessary to create the starting-up rule that synchronises the system to the expected steady state.

The procedure of the starting up rule is the following:

1	Given is the local dispatching rule $R_i(p_{i1}, p_{i2}, \ldots, p_{io_i})$, i=(1,...,m) allocated to the i-th resource.
2	Rank of processes $P_{p_{i1}}, P_{p_{i2}}, \ldots, P_{p_{io_i}}$ according to theirs increasing numbers N_{iw}, where: N_{iw} - successive number of the operation in process $P_{p_{iw}}$ executed in the i-th resource, $P_{p_{iw}}$ - successive occurrence of the process in the i-th resource, where $w=1,2,\ldots,o_i$.
3	For settled process succession, the repetitiveness of each process in the starting-up rule allocated to the i-th resource is $K_{i1}^R, K_{i2}^R, \ldots, K_{io_i}^R$ according to: $K_{iw}^R = (O_{iw} - N_{iw}) \cdot \chi_i$, where: K_{iw}^R - the product of operation numbers of process $P_{p_{iw_i}}$ remaining for execution after the i-th resource and the rule repetitiveness allocated to the i-th resource, O_{iw} - the number of operation of $P_{p_{iw_i}}$, χ_i - the repetitiveness of the dispatching rule allocated to the i-th resource.

According to the presented procedure, the starting-up rule is the following:

$$R_i\{\underbrace{(K_{i1}{}^R \cdot p_{i1}, K_{i2}{}^R \cdot p_{i2}, \ldots, K_{io_i}{}^R \cdot p_{io_i})}_{\text{starting-up rule}};\underbrace{(p_{i1}, p_{i2}, \ldots, p_{io_i})}_{\text{dispatching rule}}\} \tag{1}$$

Illustrative example

Let's assume a system of 4 resources (M_1, M_2, M_3, M_4). The following production orders Z_1, Z_2, Z_3 wait for realisation in the system. No other processes operate in the system up to this moment (the system is empty).

The route of P_1 is executed on resources M_1, M_3, M_4, process P_2 is executed on resources M_4, M_3, M_1, M_2, process P_3 is executed on resources M_2, M_1.

Matrixes (2) describe processes: the first row of the matrix describes the number of resources, the second row of the matrix represents operations time and the third one contains pre-set times.

$$M_{P_1}=\begin{bmatrix}1&3&4\\4&2&3\\0&0&0\end{bmatrix},\ M_{P_2}=\begin{bmatrix}4&3&1&2\\5&6&5&4\\0&0&0&0\end{bmatrix},\ M_{P_3}=\begin{bmatrix}2&1\\4&3\\0&0\end{bmatrix} \quad (2)$$

Dispatching rules allocated to shared resources that guarantee deadlock-free functioning and steady-state of the system are the following: $R_1(P_3,P_2,P_2,P_1)$, $R_2(P_2,P_2,P_3)$, $R_3(P_1,P_2,P_2)$, $R_4(P_2,P_2,P_1)$. The repetitiveness of the rules allocated to the resources are $\chi_1=\chi_2=\chi_3=\chi_4=1$.

According to the procedure presented in the previous section, the meta-rules allocated to the resources are the following:
$R_1\{(P_1,P_1,P_2,P_2);(P_3,P_2,P_2,P_1)\}$, $R_2\{(P_3);(P_2,P_2,P_3)\}$,
$R_3\{(P_1,P_2,P_2,P_2,P_2);(P_1,P_2,P_2)\}$, $R_4\{(P_2,P_2,P_2,P_2,P_2,P_2);(P_2,P_2,P_1)\}$.

The behaviour of the system is presented in Fig.4. One can note that the rule application assures deadlock-free behaviour of the system.

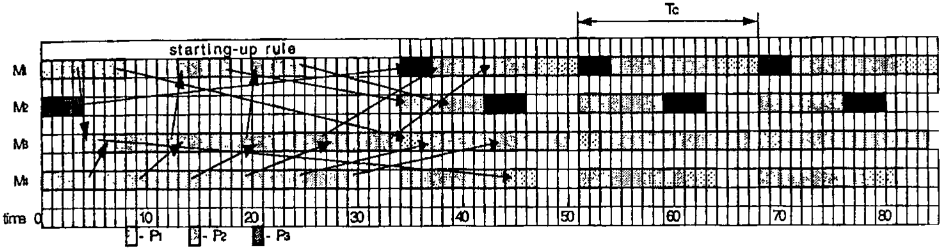


Fig. 4. Starting-up rules application. Gantt's Chart

The cycle is synchronised according to the critical resource (the occupancy of the critical resource is 100 %, it is a bottleneck of the system). The cycle of the system is 17 time units.

3.3. New process acceptance for realisation

Production order validation in the system of repetitive production consists in sufficient conditions verification, which guarantees a new production order (represented by process P_{n+1}) realisation in due time determined by given time limit (tz_{n+1}) and the moment of starting the batch realisation (to_{n+1}). This acceptance should not disturb the processes (P_1, P_2,..., P_n) already realised in the system [Skolud, 2000]. The decision is made in following steps:

In the first step, the batch production size of the new production order (B_{n+}) that does not disturb the already realised processes is determined. The following conditions should be performed:

- resources, shared with new process (P_{n+1}), are not critical,
- at least one single batch production can be executed on each resource belonging to the P_{n+1} production route.

In the situation when the process route passes on non shared resources and the condition $B_{n+1} \cdot t_w > T$ is met (where B_{n+1} is batch size, t_w – operation time on non-shared resource), the production batch delivery period (T'_{n+1}) is determined

The next step concerns the capacity of buffers allocated to the neighbouring resources $Cs(M_k, M_p)$. If condition (3) concerning the buffers capacity allocation holds, then the realisation of the batch size (B_{n+i}), is possible without any disturbance of the production process.

$$Cs(M_k, M_p) \geq 2 * B_{n+1} - 1, \tag{3}$$

The fourth step. If condition (4) is achieved, the production order with I_{n+1} number of elements will be finished in due time, in the period between tz_{n+1} and to_{n+1}.

$$\frac{I_{n+1}}{B_{n+1}} T'_{n+1} \leq tz_{n+1} - to_{n+1}, \tag{4}$$

When all the conditions are satisfied, the values describing the system performance, such as the efficiency of resources utilisation can be determined. The decision makes it possible to assign the control procedure in the form of dispatching rules (repetitive sequences of processes) allocated individually to certain resources.

Illustrative example

Let's assume a system of 4 resources (M_1, M_2, M_3, M_4). The following processes P_1, P_2, are realised in the system. The first row of the matrix contains successive numbers of resources, the second row contains operations times on those resources, the third row contains pre-set time.

Process P_3 is waiting for realisation. The routes of processes P_1, P_2 and P_3 are described by the matrixes (5) presented bellow. The number of the elements provided for realisation according to process P_3 is equal to 20 elements, due time for realisation is $tz_3=100$, available central storage capacity is $Cs=10$.

Cycle T of the system is equal to 7 and is determined by the occupancy of M_2, which is the critical resource.

The batch size for order Z_3 is $B_2 = 2$.

$$P_1 = \begin{bmatrix} 1 & 4 & 2 \\ 2 & 2 & 4 \\ 0 & 0 & 0 \end{bmatrix}, P_2 = \begin{bmatrix} 2 & 4 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix}, P_3 = \begin{bmatrix} 3 & 4 & 1 \\ 1 & 2 & 2 \\ 0 & 0 & 0 \end{bmatrix} \tag{5}$$

The occupancy of the resources is described by a set of vectors. The size of the vector is equal to the cycle time and each vector element corresponds to one time unit. The vector element is equal either to the number of a process that occupies the resource in this time unit, or to 0 when the resource is not occupied. The occupancy of the resources is presented below:

$V_1 = [1, 1, 0, 0, 0, 0, 0,]$,
$V_2 = [1, 1, 1, 1, 2, 2, 2,]$,
$V_3 = [2, 2, 0, 0, 0, 0, 0,]$,
$V_4 = [1, 1, 2, 2, 0, 0, 0,]$.

After the acceptance of production order Z_3 the occupancy is as follows:

$V_1 = [1, 1, 3, 3, 3, 3, 0,]$,
$V_2 = [1, 1, 1, 1, 2, 2, 2,]$,
$V_3 = [2, 2, 3, 3, 0, 0, 0,]$,
$V_4 = [1, 1, 2, 3, 3, 3, 3,]$.

The dispatching rules allocated to the resources are:

$R_1 = (P_1, P_3, P_3)$
$R_2 = (P_1, P_2)$,
$R_3 = (P_2, P_3, P_3)$
$R_4 = (P_1, P_2, P_3, P_3)$,

The dispatching rules describe the sequence of processes that are executed repetitively on the resources. P_3 realisation time $tr_3=70$ (<100). This means that due time realisation of this production order is possible.
In this section the theory of constraints based production flow has been presented. It has been shown that the system performance (throughput rates, resources utilisation) depends from both the effectiveness of the component elements and also the synchronisation of their interactions. The presented concept has been implemented in the Computer Aided production planning software package (http://cim.of.pl/~swz).

4. MULTI-PROJECT APPROACH

In this section the approach to the problem of capacity planning involves the tasks belonging to the category "project".
A project can be considered as an achievement of a specific objective, which involves a series of activities and tasks, which, in turn, consumes resources. It has to be completed with a definite start and end dates [2].
The function of project management includes defining the requirements of work, allocating the resources required, planning the execution of work, monitoring the progress of the work and adjusting deviations from the plan. Particularly, project scheduling is concerned with single-item or small batch production, where resources have to be allocated to activities over time.

The resources allocation, which involves the scheduling of a project to minimise its total duration, is the basic problem in project management. In [1] the presented approach is based on the following assumptions:

1. A project of different activities, which are represented in the activity –on-the-node format.
2. No activity can be started before all its predecessors have been completed.
3. No ready time or due dates are imposed on any of the project activities.
4. Each activity has constant duration.
5. The objective is to complete the project as soon as possible.

The scheduling task is to allocate activities to available resources, respecting all constraints. Two kinds of resources are distinguished [10]:

- unary resources,
- cumulative resources,

The first one can process just one activity at a given time; while in the second one the number of activities processed at a given time is limited by the resource capacity. The representative of such resources can be the cell of identical machines, team of workers etc [3].

Cumulative resources/cumulative constraints

A typical use of the cumulative constraints is found in resource restricted scheduling. A number of the tasks of different duration, where the tasks require certain amount of manpower during their operation is scheduled.

$$e_i^k - s_i^k = \Delta_i^k \tag{6}$$

$$\forall i \in M_i,\ \exists t \in \{s_i^k,\dots, b_i^k\},\ l_i^{\,t} = 0, \tag{7}$$

$$\exists m_i \in \langle\, s_i^k,\dots, s_i^k + \Delta\, l_i \,\rangle,\ \forall t_i \in \langle\, m_i,\dots\, m_i + \Delta_i^k \rangle,\ l_i^{\,t} = 0, \tag{8}$$

where:

Δ_i^k – task duration of the k-th (new) project on i-th resource

e_i^j – forward scheduling end time for the i-th resource and the j-th project,

e_i^k – forward scheduling end time for the i-th resource and the k-th project

s_i^j – forward scheduling start time for the i-th resource and the j-th project,

s_i^k– forward scheduling start time for the i-th resource and the new k-th project,

b_i^k – backward scheduling end time for the i-th resource of the new project,

$i = 1,2,3,\dots,M_i$ – number of resources,

$j = 2,3,\dots,J$– number of already implemented projects,

$k = J+1$ number of the new project,

Δl_i – time interval of the idle i-th resource

l_i^t – element of vector V_i

n –dimension of vector element V_i

For easier comprehension of the presented notation- see Fig. 5.

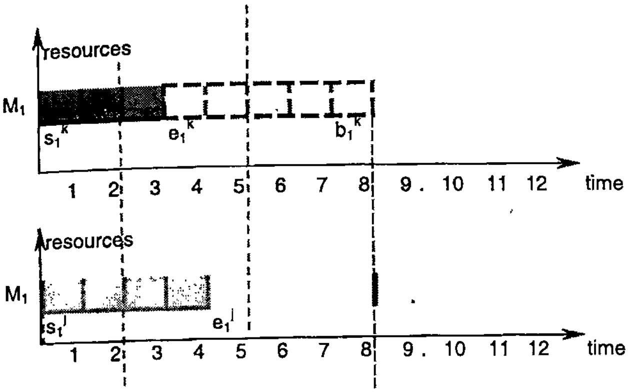


Fig.5. Illustrative example representing the notation

Formula (7) guarantees that there is a period of time on each resource, when the i-th resource is idle. Whilst formula (8) states that in the considered period of time there is a time interval of the idle i-th resource, longer or equal duration of the new project on that resource will ensure the realisation of the new project.

Let us now consider resources that are able to work simultaneously. It is useful to think about them as having partial occupancy that enables the realisation of more than one task on one resource at the same time.

While talking about resource utilisation planning, it is important to consider the state of partial occupancy, which can be represented as a next dimension of resources occupancy.

By checking the previous formulation on each level, which fully shows the possibilities of verifying the partial occupancy state and by adding condition (9), it is possible to state if the resource occupancy guarantees the new project realisation.

$$W_i + W_k \le (b_i^k - s_i^k) \cdot L_i, \qquad (9)$$

where:

W_i – processing time occupied by the already implemented projects between b_i^k and s_i^k from resource M_i;

W_k – processing time required on resource M_i for new project realisation;

L_i – maximum level of M_i resource, assigned to the project; $i = 1,2 \ldots, M_i$;

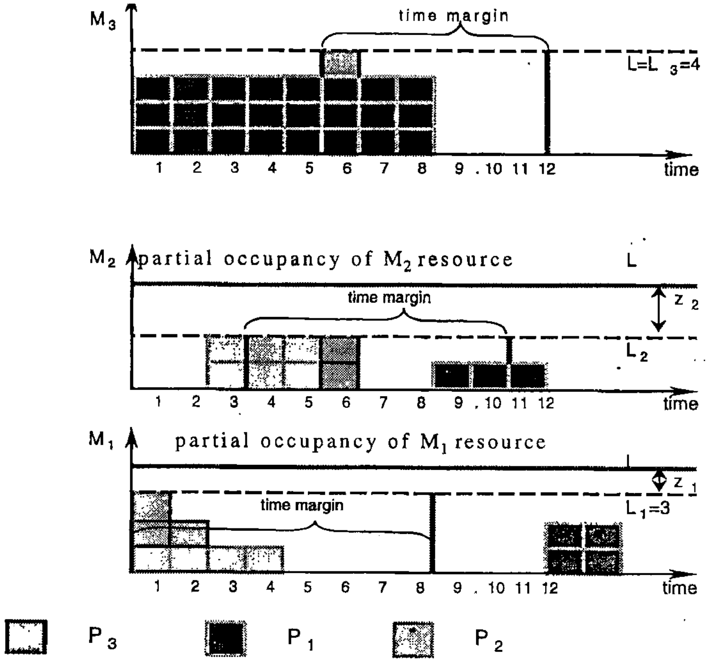
s_i^k – forward scheduling start time for the i-th resource and the k-th project;

b_i^k – backward scheduling end time for the i-th resource of the k-th project;

Illustrative example. Let us consider the system of 3 resources (M_1, M_2 and M_3). Figure 5 shows an example of resources utilisation planning in view of the state of partial occupancy. It shows how projects may be executed concurrently on the pool of shared resources.

We make an assumption that the state of partial occupancy for resources M_1, M_2 , M_3 has the following values:

- for already implemented project:
  P_1 – 4 time units on M_1, and 6 time units on M_2
  P_2 – 24 time units on M_3, and 3 time units on M_2 and 4 time units on M_1
- for the new project:
  P_3 –3 time units on M_1, 2 time units on M_2, and 1 time unit on M_3
  L_i – maximum level of resource M_1 equal 3, M_2 equal 2 and for M_3 equal 4
  $L = \max(L_i) = 4$



z_i - level of resource unavailability

Fig.6. Allocation of activities to partially occupied resources

In this example condition (5) is fulfilled, which is why the acceptance of the new project into the existing system already occupied by another task is possible. It is because the operation on M_2 resource can only be started after the preceding operation on M_1 resource is finished, which in this case can be executed earlier. □

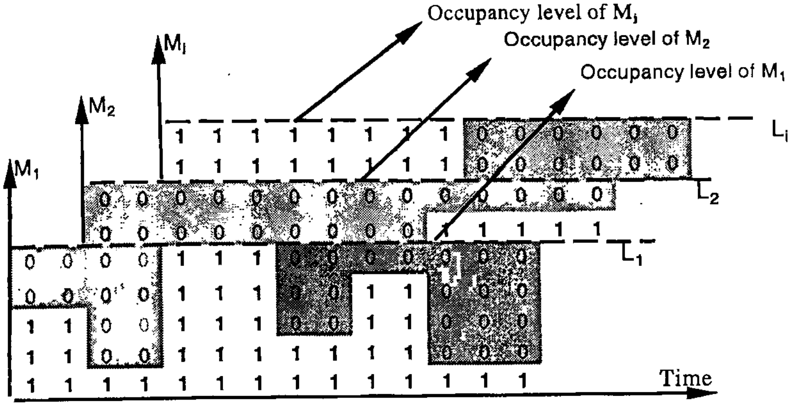
In such approach to the general resources capacity, scheduling resource levelling is considered as a third dimension. For each resource a project manager and the corresponding values can establish an indivisible unit of occupancy in time:

0 – if there is the idle i-th resource occupancy in the t-th time unit,

1 – if there is not the idle i-th resource occupancy in the t-th time unit, in other words the i-th resource is occupied.

Illustrative example. Let us see Fig.7, where it is established that $L=L_1=L_2=L_3$. Consequently, there are no units with the value equal to 2 and the resource occupancy cannot be changed, thus, the new project relocation is possible only among available time units in a way that enables their performance during the specified time.

If the free capacity area is larger or equals to the occupancy of the new project, its realisation in the given system is possible.



L_i – maximum occupancy level of the i-th resource;
W_i – overall processing time demanded from resource R_i.

Fig.7. Resource occupancy-consumption profile of a resource

While given project activities are being performed at maximum level L_i of each resource i, on which the project will be executed ($i = 1,2,\ldots, M_i$) are assigned, respectively. If the project that is being executed requires the resource capacity smaller than L_i, the project will use all that it needs and the remaining units will be idle for that resource. If for some processing stage, the considered project requires more resource units, then it will use all L_i units for an extended period of time and thus the total work served in that stage is executed.

For solving the problem of the resource restricted scheduling the cumulative constraints can be used [7]. A number of tasks of different duration where the tasks required certain amounts of manpower during their operation have to be scheduled. The available amount of manpower is fixed and total requirements at each time should not exceed the available limit.

This constraint can be modelled with a simple form of the cumulative constraint. For each of the n tasks, the start time Sj, duration Dj and resource use of the tasks Rj.

$$\text{cumulative}([S1,S2, \ldots Sn], [D1,D2,\ldots,Dn], [R1,R2, \ldots,Rn],\text{Limit}), \qquad (10)$$

where Limit ranges between 0 and the available resource limit.

The mathematical definition is the following:

$$\forall i \in [\min_{1\le j\le n}(S_j), \max_{1\le j\le n}(S_j + D_j)]: \quad \sum_{k:S_k \le i \langle S_k + D_k} R_k \le Limit \tag{11}$$

This means that at each point of time between the start of the first tasks and the end of the last task, the cumulative resource used by all tasks is smaller than the available resource limit.

In this section the utilisation of resources that are able to work concurrently and that can be partially occupied have been presented. The presented problem is still open. A solution of it may be constraint based scheduling that can be an efficient tool for solving such scheduling problem.

4. CONCLUSIONS

In the paper different approaches concerning the capacity planning in SMEs are discussed.

These approaches are focused on successful problem solution, answering the question if it is possible to accept a new process for realisation in the system where another processes are already being realised and how to effectively exploit resources capacity.

The author explores the solution that takes advantage of the constraint-based scheduling, which is a framework for solving scheduling problems by stating the constraints on the problem variables.

REFERENCES:

1. Demeuleemester E.L., Horroelen W.S.: An effective optimal solution procedure for the preemtive resource-constrained project scheduling problem. European Journal of Operational Research, , 90, 1996, pp. 334-348
2. Munns A.K., Bjermi B.F.: The role of project management in achieving project success. International Journal of Project Management, vol.14, No.2, 1996, pp.81-87
3. Lowa A., Maroto C, Tormos P.: A multicriteria heuristic method to improve rosource allocation in multi-project scheduling. European Journal of Operational Research. 127, 2000, pp.408-424.
4. Schonsleben P., *Integral Logistic s Management*, St. Lucie Press, London, 2001
5. Goldratt E:, Cox J.F.: 1987, *The Goal*, Coton on Hudson, NY; North River Press
6. Lepore D., Cohen O.: *Deming and Goldratt – The theory of constraints and the system of profound knowledge*. The North River Press, Great Barrington, USA 1999.
7. Simonis H.: Constraint Logic Programming, Cosytec S.A., 11/12/1997, France
8. Skołud B.: *Planning of the multiassortment repetitive production* (in Polish) Zeszyty Naukowe Politechniki Śląskiej, s. Mechanik, No.136, Gliwice, 2000
9. Skołud B., Krenczyk D.: *Flow Synchronisation of the Production Systems: The Distributed Control Approach.* The 6th IFAC Workshop on Intelligent Manufacturing systems IMS'2001, Poznań 24-26, 2001, pp. 127- 132.
10. Skolud B., Zientek A.: Constraints-based multi-project scheduling in Project Driven Manufacturing (Eds.: Z. Banaszak, J. Józefowska), WNT, Warszawa 2003.