ALGORITHMS FOR MANUFACTURING SYSTEM CONTROL UNDER OPERATING CONDITIONS CHANGE

V. E. Bolnokin[1,], S. A. Sorokin[2], A. V. Gorshkov[2]*

[1]Mechanical Engineering Research Institute of the Russian Academy of Sciences (IMASH RAS), Voronezh; [2]Computing System Research and Development Institute (NIIVK, JSC)
Russian Federation

[*] Corresponding Author, e-mail: vitalybolnokin@yandex.ru

Abstract: The paper is devoted to the mathematical problems of building control systems of modern and promising production systems that ensure the adaptation of production processes to changes in the external and internal conditions of the system. The proposed algorithm for solving the scheduling problem is based on a modification of the algorithm for finding the shortest path between the vertices of the graph. It is a simple and effective tool for solving the problem of minimizing the changeover time. Its effectiveness has been proven. The algorithm has received an adaptive extension when applied in cases where it is necessary to take into account random influences on the production process. For this purpose, planning algorithms are considered, taking into account the actual noise and interference. Computer algorithms for production planning using a system of correction tolerances for the required work schedules and equipment loading are synthesized.

Key words: manufacturing systems, technological structure, control systems, adaptation of production.

1. INTRODUCTION

The fundamental factor in improving the efficiency of technological modern production systems (PS) is the solution of planning problems (long-term and short-term, rescheduling) and operational management. In the PS the same technological equipment (machines, machine tools, centers, lines) is intended to output products of various nomenclature, so there is a need for readjustment. Herewith, time spent on readjustment can be commensurate with the processing time, and readjustments themselves are often associated with time-consuming and harmful operations performed by the service personal Planning the operation of technological equipment that minimizes the total readjustment time is a significant reserve for increasing the productivity of the PS, primarily by reducing equipment downtimes. This paper is devoted to the development of the control algorithms that enables to minimize the time spent on the readjustment of technological equipment.

The purpose of the paper is to create control algorithms aimed at improving the efficiency of modern production systems through operational reconfiguration to minimize the time spent on readjustment of technological equipment. The proposed algorithm for solving the planning problem is based on the algorithm for finding the shortest path between the vertices of the graph. The graph is designed in such a way that the route from the initial vertices of the graph to the final ones can be interpreted as some schedule for processing. This is the novelty of the approach used. The realization of the set goal is made in the next two sections, as section 2 discusses the deterministic algorithms and proposes the solution method and algorithm of readjustment optimization, and the section 3 presents the proposed stochastic algorithm based on the probabilistic evaluation of the optimality criterion of a dynamically generated graph.

2. DETERMINISTIC ALGORITHMS

Nowadays there are not any sufficiently effective methods to solve this problem, in particular, the theory of schedules [1-3], the theory of he readjustment optimization are insufficiently developed in the literature. In the theory of schedules time required for the equipment readjustment recorded in the following two ways. In the first method the readjustment time is added to the time of processing operation this allows to indirectly exclude the readjustment time from consideration [1]. The method is based on the assumption that the time of the module readjustment to perform the work j does not depend on work i, performed on the adjacent modules before work j. However, this assumption is unacceptable in many applications. In the second method the time of the processing operation is added to the readjustment time, while the time of technological operation is excluded from consideration and the task of minimizing the readjustment time is to be solved by the travelling salesman [1-3]. The disadvantage of this method is the lack of simple and effective methods to solve the travelling salesman problem. In a more general statement of the problem, in contrast to the above-mentioned models, when planning the sequence of readjustments, the time of the technological operation and the time of the equipment readjustment are taken into account.

2.1. Problem statement and solution method

The model, in which the proposed algorithm can be applied, is described. It should be noted that in the production process designed for the output of products of different nomenclature it may be necessary to readjust technological processes of different levels. The level that requires the optimal planning is considered (further it will be called a processing machine).It is assumed that the processing machine can be in one of the states S, S = 1,…,R, in each state the products are processed in accordance with the state number of the nomenclature. The transfer of the machine from state i to state j will be called readjustment. It is assumed that the technological subsystem selected as a result of the decomposition is included in the overall technological process, and, therefore, the products must be processed by certain time – directive deadlines. Performing f certain technological operation on the machine will be called work hereinafter. Taking into account these definitions the problem of minimizing the time spent on the machine readjustment for the described model can be reduced to the following problem of the theory of schedules.

It is necessary to perform N work on the machine. The directive deadlines to perform work D_i, i =1,…,N are determined. Work is submitted for maintenance in the following time period t_i, i =1,…,N. The machine can be adjusted to one of к states. The time of the machine readjustment from state i to state j is defined by the matrix $\Delta = \{\delta_{ij}\}, i, j = 1,..., k$. It is assumed that the matrix elements have the following properties: $\delta_{ij} = 0, \delta_{1i} + \delta_{1j} > \delta_{1j}$.

This condition is important for construction and justification of the algorithm. It is assumed that this condition is not limiting: if it is not met, the machine adjustment from state i to state j can be carried out through the intermediate state 1, so the condition will be met.

Work requiring maintenance on the machine is denoted through P: $P_1^S,…,P_N^S$, where i – a serial number of work, s = 1,…,k – required machine adjustment to perform i. It is supposed that the following conditions are met: work is performed without interruptions; no more than one work can be performed on the machine at any given time. It is necessary to determine the order of work on the machine that meets the directive deadlines to perform work D_i (i = 1,…,N) and minimizes the total time of the machine readjustment T(P).

The proposed algorithm for solving the problem of scheduling is based on the well-known algorithm for finding the shortest path between the vertices of the graph [2-4]. Consider the algorithm for finding the shortest path in more detail. Let be given graph G, uniquely defined by a set of its vertices and arcs. The vertices of the graph are denoted by an ordinal number, the arc of the graph connecting vertex i to vertex j, - by γ_{ij}. Each arc γ_{ij} is compared to a number r_{ij} – arc length γ_{ij}. If there is no arc γ_{ij}, it is assumed that $r_{ij} = \infty$. The graph G contains a set of initial ($i_1^h,…,i_n^h$) and final vertices. It is necessary to construct a minimal path of the graph to connect the initial vertices of the graph with the final ones, herewith the length of the path is equal to the sum of all included in it arcs. Each vertex of the graph G is compared to two numbers: R_1 – the path length from the initial vertices t0 the vertex i и N_1 – vertex number from which the transition to the vertex i was made. The shortest path in the graph G is calculated using the following algorithm.

Step 1. Assign the initial values:

$$R_1 = 0, \text{ if } i \in \{i_1^h,...,i_n^h\}, \; R_1 = \infty, \text{ if } i \notin \{i_1^h,...,i_n^h\},$$

$$N_1 = 0 \text{ for } \forall_i,$$

where ∞ denotes a number bigger than the sum of all the lengths of graph arcs.

Step 2. All the vertices i of the graph G, for $R_i < \infty$. are viewed gradually. The new value $R_j^{'} = R_2 + r_{ij}$. is calculated for each vertex j .

If $R_j^{'} < R_{j,}$ then $R_j = R_j^{'}; N_j = i$ is assigned.

Step 3. If the length R is changed after performing step 2, repeat step 2, otherwise go to step 4.

Step 4. Find the vertex with the minimal length R among the final vertices of the graph.

Step 5. Using marks N restore the shortest path from the vertex found in step 4.

This algorithm is applied to solve the problem of optimizing readjustments The graph of the problem is made so, that the route from the initial vertices of the graph to the final ones could be interpreted as some schedule for the processing machine.

The work requiring a single machine adjustment is grouped:

$$P_1^1, P_2^2,..., P_n^1,$$

$$P_1^2, P_2^2,..., P_n^2$$

....

$$P_1^k, P_2^k,..., P_n^k.$$

The work in groups is arranged in ascending order of the directive deadlines i.e. for any work P_1^s, P_j^s, if i<j, D_i<D_j. The time for work performing is plotted in groups on the orthogonal axes of the k-dimensional space in a fixed order. (k-1)-dimensional subspaces orthogonal to the corresponding axes are constructed using the obtained points. The intersection r-subspaces forms the vertices, and the intersection (k-1)-subspaces – arcs of the graph. The obtained graph (Fig. 1) is denoted as $G^{'}$. The movement in the graph $G^{'}$ along the arc parallel to a certain axis will be interpreted as the performance of the corresponding work plotted on this axis. The arc length is assumed to be equal to the time of corresponding work. Certain vertex i of the graph denotes such a state when all the work on the machine has been done (Pic.1), whose coordinates of the beginning are less than the ones of the vertex i. The initial vertex H and the final vertex K are selected in the graph G'. Then the path from the initial vertex to the final one can be considered as a certain schedule for the processing machine. The graph G' is an auxiliary one. It is transformed into the graph G – a working graph for solving the problem. It will be assumed that $\delta_{ij} = \delta_{ji}$, though this condition is optional. Each vertex of the graph G' is transformed into a group of vertices consisting of k vertices connected in pairs. Each vertex of the group is connected to two arcs of the graph G', which are parallel to the axis of the corresponding adjustment of the machine. The arcs, connecting the vertices in the groups, define the process of the machine readjustment and have the length equal to the time of the corresponding readjustment δ_{ij}. the initial and final vertices of the graph G' are transformed into the group of the initial and final vertices of the graph G respectively.

The length of any route on the graph G from the initial vertices to the final ones defines some schedule of the processing machine:

$$L = T + T(P^R),$$

where T – the total time of all the work performance, it is constant; $T(P^R)$ – the total time of the machine readjustment depending on the schedule P.

The formulated problem of the theory of schedules on the graph G is transformed into the following one: find the minimum path connecting the initial vertices of the graph G with the final ones, meeting the directive deadlines of the work performance.

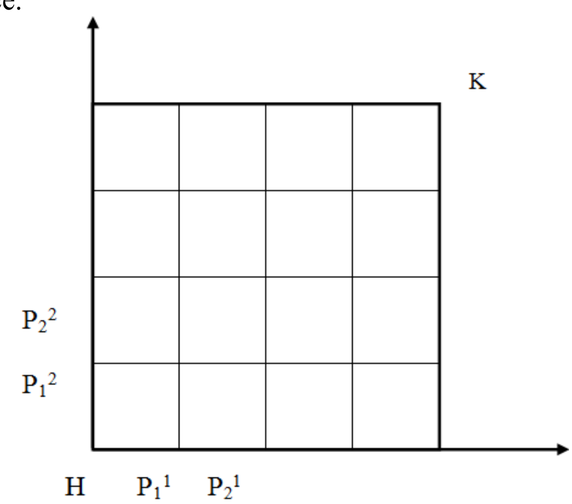

Fig. 1. Graph G' (k=2)

2.2. Algorithm of readjustment optimization

The problem is solved by using the shortest path search algorithm making the following changes. To set the limits the directive deadlines for the work performance will be taken. When passing the arc of the graph interpreted as the performance of some work, the new calculated value R is compared to the directive deadlines for work performance. If the directive deadline is less than R, the transition along this arc is not performed. Taking into account all the foregoing, the following algorithm for solving the problem is proposed.

Step 1. Assign the initial values:

$$R_i = 0,\ \text{if}\ i \in \{i_1^h,...,i_n^h\},\ R_i = \infty,\ \text{if}\ i \notin \{i_1^h,...,i_n^h\},\ N_i = 0\ \text{for}\ \forall_i.$$

Step 2. All the vertices i of the graph G, for which $R_i < \infty$, are viewed gradually. The new value $R_j' = R_1 + r_{ij}$ f or each vertex j is calculated.

If $R_j' < R_j$ and $R_i' < D(\gamma_{ij})$, then $R_j = R_j'; N_j = i$, is assigned where $R(\gamma_{ij})$ – directive deadline of the work γ_{ij} (is accepted equal to ∞).

Step 3. If performing step 2 leads to a changing R_i, repeat step 2, otherwise go to step 4.

4. If all the final vertices have $R_i = \infty$, i.e. the final marks are not obtained, the problem has no solution, otherwise go to step 5.

5. Find among the final vertices of the graph the vertex with minimum R_i.

6. Using marks N_i restore the shortest path from the vertex found in step 4.

The necessary condition for constructing graph G is to fix the order of work performance in groups according to the increasing the directive deadlines of work performance. The introduction of such an order restricts the search area of the optimal solution many times. There are $n_1!...n_k!$ graphs. It is natural to assume that there is another optimal order of work performance in groups, by fixing it the schedule of lower total time of readjustment compared to the described algorithm can be constructed. However, it is not the case, that proves the following theorem.

Theorem 1. Let it be given a valid schedule P (meeting the directive deadlines) with the total time of the readjustment of the machine $T^R(P)$, then there is a valid schedule P', that is a route on the constructed graph G, the total time of the readjustments is

$$T^R(P') < T^R(P). \tag{1}$$

The fixed order of work execution in groups is optimized for this task. The proof of this theorem is based on the use of inequality (1), valid for the matrix of the readjustment time $\{\delta_{ij}\}$.

The described algorithm is simple and effective means of solving the problem of minimizing the readjustment time. It can be used to solve the problems of scheduling in the PS of different designation. It is most appropriate to use it in cases where the time spent on equipment readjustment is significant.

3. STOCHASTIC ALGORITHMS

The deterministic algorithm was adaptively extended when applied in cases where it is necessary to take into account the random influences on the production process [4-5].

Let us consider the scheduling algorithms for discrete manufacturing in a stochastic formulation. The main result is as follows: let each batch-operation be regarded as a separate starting batch, for which a delivery time and a scheduled release time are set, and if (p,j) and (p,j+1) are two batch-operations on a single batch of the processed component, followed by each other, and the new batch numbers p' and p" correspond to them, in this case the scheduled release time of the previous batch becomes the delivery time of the next batch: $D_{p'} = d$. It is also assumed that there is no preventive maintenance. It is required to generate a delay-free schedule $L = (m_p, \underline{t}_p), \quad p = 1,2,...,P,$ minimizing the quality functionality $L = \max_p(\bar{t}_p - D_p)$ and satisfying the condition $\underline{t}_p \geq d_p$. Where: m_P – number of the module, which is used for processing the batch p=1,2,…,P; $\underline{t}_h$ - processing start; $\bar{t}_p = \underline{t}_p + t_p$ - end of the batch p processing.

Theorem 2. In order to ensure the existence of the schedule $L^{[M]} = \left\{t_{p_1}, t_{p_2}, ..., t_{p_p}\right\}$, satisfying the scheduled periods $D_p^{[N]}, \quad p = 1,2,...,P,$ it is necessary that

$$\sum_{i=1}^{\{p\}} t_{p_i} \leq ND_p^{[N]} + (t_{p-1}^{*} - t_p)(N-1),$$

and it is sufficient that

$$\sum_{i=1}^{\{p\}} t_{p_i} \leq ND_p^{[N]} - (N-1)t_p,$$

where $t_{p-1}^{*} = \max_{i \in 1,\overline{\{p\}-1}} t_{p_i}$. Please note that the inequality

$$\underline{t}_p^{[N]} \leq \frac{\underline{t}_p^{[1]}}{n} \leq \underline{t}_p^{[N[} + \frac{N-1}{N} t_{p-1}^{*},$$

leads to

$$\frac{\underline{t}_p^{[1]}}{N} - \underline{t}_p^{[N]} \leq \frac{N-1}{N} t_{p-1}^{*}, \quad p = 2,3,...,P.$$

From the definition t^{*}_{p-1} it follows that this difference will be minimal if the L list, from which the $L^{(1)}$ and $L^{(N)}$ schedules were obtained, was developed in the ascending order tp, p=1,2,…,P. Then the quality of the obtained schedule is determined when the processing times of the tp batch are random, while having the same distribution F(x).

Further, the planning algorithms are considered with allowances made for the actual noise and interference. Let the following be set for each batch n:

t_n - batch processing time, with due consideration of the preparatory and blanking operations;

d_n – blanks delivery period;

D_n – scheduled time of batch release.

It is required to find the optimal schedule

$$A = (m_n, \underline{t}_n, \bar{t}_n), \quad n = 1,2,...,N,$$

where $m_n \in \{1,2,...,N\}$- the number of the module, which is used for processing the batch n; t_n and t_n - start and end of batch n processing. The optimality criterion is the function $h(A) = \max\limits_{1\le n\le N}\left[\max(0, \bar{t}_n - D_n)\right]$, which has to be minimized over all possible schedules A. For each allowable schedule, the following limitations are fulfilled:

$$\underline{t}_n \ge d_n, \quad \forall_n = 1,2,...,N; \bar{t}_n = \underline{t}_n + t_n, \forall_n = 1,2,...,N;$$
$$m_i = m_j \Rightarrow \left[\underline{t}_i, \bar{t}_j\right] \cap \left[\underline{t}_j, \bar{t}_j\right] = 0, \ i, j = 1,2,...,N.$$

Assume that the processed blank parts come to the site long before the scheduled period, which makes it possible for us to simplify: $d_i = 0, \forall_i = 1,2,...,N.$ Let's limit ourselves to the class of compact schedules, for which the following condition is met: if $m_i = m_j$ and batch j is processed by m-м module immediately after batch i, then $\bar{t}_i = \underline{t}_j$. Each compact schedule A has a unique correspondence to a certain list L = {n_1,n_2,…,n_N}, from which schedule A is obtained using the list algorithm. The batches are included in the schedule in the sequence of the given list, where each batch is assigned to the module that is released earlier than the others, and no interruptions at the time of processing between the batches are allowed. The schedule for M modules, obtained from the list L with the use of the list algorithm, are specified as A^{M}_{L}.

Therefore, the problem of setting up a preliminary schedule can be written as follows: find such a list L_0 that

$$h\left(A^{M}_{L_0}\right) = \min_{L \in D(N)} h(A^{M}_{L}) = h_0,$$

where D(N) is a set of rearrangements on the set {1,2,…,N}.

The authors demonstrated that

$$h\left(A_L^M\right) < \max_{1 \le k \le N} h_k = \delta,$$

where $h_k = \max_{1 \le i \le N_k} \left(\max(0, \delta_i^k - D_i)\right)$.

Thus, we have obtained the probabilistic estimate δ for the optimality criterion of this schedule A_L^M, which depends on the list L and scheduled times Di, i=1,2,…,N. If the scheduled dates satisfy the condition

$$D_n \ge \theta_n^m + \Delta_n^m, \forall_n = 1,2,..., N, \quad (2)$$

then $\delta = 0$ with a probability of P0 and, therefore, the schedule A_L^M with scheduled dates satisfying the condition (2), is optimal with a probability of P0. If the scheduled dates fail to satisfy the condition (2), but it is possible to change them or set them independently, then condition (2) can be used during their selection.

If the scheduled dates fail to satisfy the condition (2) for this particular schedule and they cannot be changed, then the schedule must be changed in such a way as to fulfill the condition

$$\sum_{i=1}^{n} t_i = \theta_n^m \le D_n - \Delta_n^m,$$

where Δ_n^m is determined using the following formula:

$$\Delta_n^m = \Delta\left(P_0, \theta_n^m\right) = \frac{\lambda}{\mu}\theta_n^m + \frac{f\left(2P_0 - 1\right)\left(2\lambda\theta_n^m\right)^{1/2}}{\mu}.$$

This theoretical basis is applied to compose computer algorithms for manufacturing planning using system of tolerances correction for the required work schedules and machine utilization.

4. CONCLUSION

The paper is devoted to the conceptual and mathematical issues of building planning and operational management systems for various types of production systems. The developed algorithms can be used in solving calendar planning problems in production systems for various purposes. It is most appropriate to use it in cases where the time spent on the equipment changeover is significant. Computer algorithms for production planning using a system of correction tolerances for the required work schedules and equipment loading are synthesized.

REFERENCES

[1] Terekhov D., Tran T.T., Down D.G., Beck J.C. Integrating Queuing Theory and Scheduling for Dynamic Scheduling Problems. *Journal of Artificial Intelligence Research*, vol. 50, 2014, pp. 535-572. DOI: 10.1613/jair.4278.

[2] Gawiejnowicz S. The modern scheduling theory. In: *Models and Algorithms of Time-Dependent Scheduling. An EATCS Series* (Springer, Berlin, Heidelberg), 2020, pp.65-82. DOI: 10.1007/978-3-662-59362-2_5.

[3] Paul A., Martonosi S. Operations Research. In: *Science for Mathematicians* (Chapman and Hall/CRC), 2020, 52 p. DOI: 10.1201/9780429398292-6.

[4] Bolnokin V.E., Ho D. Loc. *The adaptive management on the base of fuzzy regulators and neural network technology* (Science Book Publishing House, Voronezh – in Russian), 2012, 280 p.

[5] Bolnokin V.E., Mutin D.I., Mutina E.I., Storozhev S.V. The synthesis of the algorithms for adaptive control by nonlinear dynamic objects on the basis of the neural network. *IOP Conference series: Materials Science and Engineering,* vol. 537, 2019, pp. 042013. DOI: 10.1088/1757-899X/537/4/042013.

Information about the authors:

Vitaly Evgenyevich Bolnokin – Doctor of Sciences (Engineering), Professor, chief researcher an Mechanical Engineering Research Institute of the Russian Academy of Sciences, Russian Federation, areas of scientific research – system analysis, optimization, simulation of complex objects.

Sergey Aleksandrovich Sorokin – Doctor of Sciences (Engineering), general designer, M. A. Kartsev Computing System Research and Development Institute (NIIVK, JSC), areas of scientific research – system analysis, optimization, simulation of complex objects.

Aleksey Vladislavovich Gorshkov – head of the Scientific and Technical Center, areas of scientific research – application of artificial intelligence methods for solving applied problems in the field of information security.

Manuscript received on 05 June 2021