Production scheduling with ant colony optimization

A S Chernigovskiy[1], **D V Kapulin**[1], **E E Noskova**[1], **T N Yamskikh**[1], **R Yu Tsarev**[1]

[1] Siberian Federal University, 79, Svobodny Prospect, Krasnoyarsk, 660041, Russia

E-mail: achernigovskiy@sfu-kras.ru

Abstract. The optimum solution of the production scheduling problem for manufacturing processes at an enterprise is crucial as it allows one to obtain the required amount of production within a specified time frame. Optimum production schedule can be found using a variety of optimization algorithms or scheduling algorithms. Ant colony optimization is one of well-known techniques to solve the global multi-objective optimization problem. In the article, we present a solution of the production scheduling problem by means of an ant colony optimization algorithm. A case study of the algorithm efficiency estimated against some others production scheduling algorithms is presented. Advantages of the ant colony optimization algorithm and its beneficial effect on the manufacturing process are provided.

1. Introduction

Modern science-based and high-tech product consists of a large number of parts and assembly units; a wide range of materials nomenclature and purchased components are used in its manufacture. The number of types of parts in the product can reach several tens of thousands, which greatly complicates the manufacture and requires a special approach to its organization. The production scheduling problems are set for optimum organization of control of the production process. They are characterized as problems of providing the specified amounts of output, in the required tempo and high quality.

From the mathematical point of view, the problem of production scheduling is a complex combinatorial problem that has multiple solutions, among which it is necessary to find the solution, which is optimum in terms of some criteria. This problem can be solved exactly (mathematical programming, Gomory's cut) or approximately (heuristics, Monte Carlo method) [1]. The main optimality criteria are: total throughput time minimization [2], readjustment time minimization [3], the minimum cost of schedule execution criteria [4], etc. The optimum production schedule searching can be performed using linear programming, dynamic programming, combinatorial or evolutionary algorithms [5].

As a result of the analysis of existing methods and algorithms of production scheduling, the decision of using evolutionary algorithms was made, such algorithms of the global multi-objective optimization allow obtaining the best solutions to problems of real production situations in a short time. Among the evolutionary algorithms, the most perspective solution of complex combinatorial optimization problems is to use an ant colony optimization algorithm and its modification with the elitist ants addition, which reinforce the best routes that are found from the beginning of the algorithm [6]. The advantage of this algorithm for the denoted problem is that this algorithm does not require the composition of a structural model of the production area itself.

2. Research method

According to the scheduling theory production, scheduling problem can be denoted as a batch processing tasks of n parts on m machines: n / m / G / $C_{\max}$ [7]. In case of assembly manufacture processes we will refer parts to manufacturing operations and machines to a device / worker (fitter, turner, etc.). Further, we will use terminology of scheduling theory.

Suppose we are given the sets $M = \{M_1,\ldots,M_m\}$ – specified number of machines; $J = \{J_1, \ldots, J_n\}$ – specified number of parts; $O = \{O_1, \ldots, O_n\}$ – specified number of operations. For each operation $u_{ij} \in O$ there is a part J_i, which it belongs to, machine M_j, to be processed on, and processing time f_{ij} of u_{ij} operation, where f_{ij} – nonnegative integer number. It is required to find the starting times of operations so that the completion time of the latest operation will be minimum. For each part, sequence of operations must be kept, each machine can process only one part at a time. It is needed to select a sequence of parts processing on the machines, i.e. make a schedule, for which processing time $C_{\max}$ will be minimum while satisfying all the constraints.

The generalized scheme of the algorithm of production schedule construction using the ant colony optimization method is narrowed down to the construction of an assembly process model as a directed graph, the ant colony formation and solution search considering the limits specified.

3. Construction of the assembly process model as a directed graph

The assembly process is represented by disjunctive weighted graph $G = (V, A, E)$, where V – set of vertices, A – connecting set of directed arcs, and E – disjunctive set of arcs. The vertices V correspond to all operations and two fictitious (dummy) vertices, namely, source and sink. A connecting directed arcs A represent precedence relations between operations for each part, and disjunctive arcs E represent all pairs of operations to be performed on the same machine. All outcoming directed arcs from a vertex are assigned to the processing operation durations. The source of fictitious operation O_s has connecting arcs of zero length for all operations for each part to be processed on the first machine, and the sink of fictitious operation O_f, have connecting directed arcs outcoming from all operations performed on the last machine. Performing scheduling corresponds to a choice of one directed arc from each disjunctive arc and connections so that the resulting directed graph become acyclic.

A disjunctive graph for solving the $3/3/G/C_{\max}$ scheduling problem is depicted in Figure 1, where O_{ij} – operation of processing the i part on the j machine.

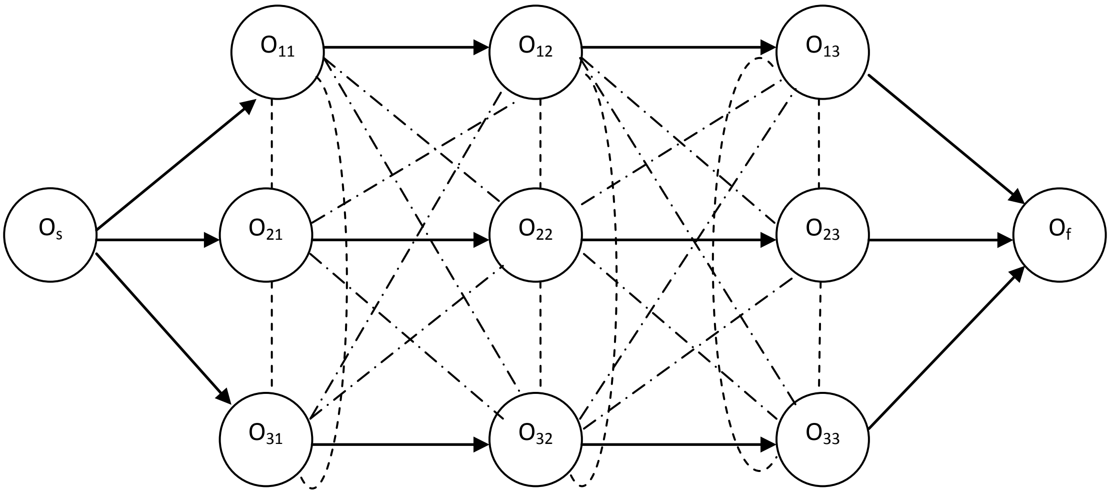

Figure 1. Disjunctive graph $3/3/G/C_{\max}$

Directed arcs connecting the O_{ij} operation with the O_{iu} operation are directed connecting arcs. Directed arcs connecting the O_{ij} operation with the O_{bu} operation are directed disjunctive arcs.

4. Ant colony formation

An ant colony formation includes the determination of ants quantity in accordance with the specified technological assembly process (for the modified ant colony algorithm is determined by the number of "elitist" ants as 1% of the total ants quantity); pheromone concentration correction rules; rules of ants behavior in the construction of solutions in the form of the transition probabilities.

The general plan of ant colony optimization algorithm for operative scheduling problem solving [6]:

1. Technological process graph formation.
2. Algorithms parameters initialization.
3. Ant colony creation and assigning to the graph.
4. Transition scheme construction among graph vertices.
5. Selection of the most optimum schedule.
6. Pheromone update.

Steps 3-6 are repeated until a stopping condition of the algorithm will be reached, which is a predetermined number of iterations.

Initialized algorithm parameters:

α – the weight of the pheromone trail (selection of short operation is most likely if $\alpha = 0$, algorithm becomes greedy);

β – visibility in the selection of the route (at $\beta = 0$, choice is made only on the basis of the pheromone, resulting in suboptimal solutions);

q_0 – a parameter which determines whether the next operation will be selected, $q_0 \in [0, 1]$;

ρ – pheromone evaporation coefficient, $\rho \in [0, 1]$;

Q – value of the order of the optimal path.

Between adjustable parameters α and β compromise is necessary, which can be found experimentally [6].

The number of ants in the batch depends on the number of vertices in the disjunctive graph. The number of ants batches is set arbitrarily. All ants are placed in the same starting point – O_S on the disjunctive graph. The algorithm for the optimum schedule search procedure is worked out for each k ant, until k reaches the quantity of ants in the l batch, i.e. performed for the number of consecutive loops, equal to the product of the ants quantity on the number of ant batches. At each step of the k ant loop, the place in the schedule for one of the jobs of the l batch is defined.

At this step the initial pheromone level is specified. It is initialized to a small positive number, so the initial step transition probabilities to the next vertices of the graph will not become zero values.

At each iteration of the algorithm, each ant builds a path to the end node step by step. In this case, at each vertex, each ant has to choose the next directed arc of the path from the list of candidates, which is a subset of the unvisited vertices list. Selected vertices are sequentially added to the list of visits. If an ant k is in vertex ij, it chooses the next vertex $bu \in N_{ij}^{k}$ based on transition probabilities.

When visits list is empty, k ant chooses the operation from the candidate list according to two rules of transition that combine pheromone level information and heuristic information:

$$bu^k(t) = \begin{cases} \arg\left\{ \max\limits_{hg \in C^k(t)} \left[\tau_{ij,hg}^{k}(t)\right]^{\alpha} \left[\eta_{ij,hg}^{k}(t)\right]^{\beta} \right\} & \text{if } q \le q_0, \\ bu & \text{if } q > q_0, \end{cases} \qquad (1)$$

where $\tau_{ij,\ hg}$ – a pheromone level between operations ij and hg;

$\eta_{ij,\ hg}$ – a heuristic distance between operations ij and hg;

C^k – a list of possible candidates-operations that an ant may visit after completing the previous operation;

q – a random number uniformly distributed in [0, 1];

q_0 determines whether the next operation will be selected in accordance with the expression (1).

The probability of a random selection of the following operation from the candidates list is determined by the expression [6]:

$$p_{ij,bu}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij,bu}^{k}(t)\right]^{\alpha}\left[\eta_{ij,bu}^{k}(t)\right]^{\beta}}{\sum\limits_{hg\in C^{k}(t)}\left[\tau_{ij,bu}^{k}(t)\right]^{\alpha}\left[\eta_{ij,bu}^{k}(t)\right]^{\beta}} & \text{if} \quad bu\in C^{k}(t),\\ 0 & \text{if} \quad bu\notin C^{k}(t).\end{cases} \tag{2}$$

When an ant has performed an operation, it moves from the list of unvisited operations to the list of visited operations and accordingly removed from the operations candidates list. The procedure is repeated until the ant visits all the vertices in the graph. The sequence of operations in the visited list represents a complete processing order of the batch of *n* parts by *m* machines, which identically determines the problem solution.

5. The solution search according to determined constraints

Optimum schedule selection is determined by the L^k parameter. L^k – the processing time of the parts batch on all the machines, i.e. the processing time of all operations in the graph that passed by *k* ant, from the first to the last vertex of the path, calculated using the formula (1). If found L_k value for the *k* ant is less than the stored value $C_{\max}$, the $C_{\max}$ takes on a value L_k, the visits list of the *k* ant is accepted as the best selected route at the moment and saved as an optimum path.

When all the ants within the technological assembly process have constructed a full path from initial to final vertex, each ant marks its way, putting for each directed arc pheromone in accordance with the following expression [6]:

$$\Delta\tau_{ij,bu}^{k}(t)=\begin{cases}\dfrac{Q}{L^{k}(t)} & (ij,bu)\in V^{k}(t),\\ 0 & (ij,bu)\notin V^{k}(t),\end{cases} \tag{3}$$

where $V^k(t)$ – the route passed by ant *k* at time *t*.

The rule of pheromone evaporation is defined as follows [6]:

$$\tau_{ij,bu}(t+1)=(1-g)\cdot\tau_{ij,bu}(t)+\Delta\tau_{ij,bu}(t),$$
$$\Delta\tau_{ij,bu}(t)=\sum_{k=1}^{N}\Delta\tau_{ij,bu}^{k}(t), \tag{4}$$

where g – an evaporation coefficient, $g\in[0,1]$; N – ants quantity in the colony. From (2) it follows that the total concentration of the pheromone for given directed arc is proportional to the “quality’ of paths, which include that directed arc.

To improve time characteristics of the ant colony algorithm, “elitist ants” are introduced. They enhance directed arcs of the best route, found at the moment. The pheromone quantity that retained on the directed arcs of the current best route $C_{\max}$ is assumed to be equal to Q/L^{+}, where L^{+} – the length of the $C_{\max}$ route.

This pheromone causes the ants to do the study of solutions containing several directed arcs of the best $C_{\max}$ route at the moment. If the anthill has *e* elitist ants, the $C_{\max}$ route directed arcs get overall strengthening [6]:

$$\Delta\tau_{e}=e\cdot\frac{Q}{L^{+}}. \tag{5}$$

In the modified ant colony algorithm *e* number of elitist ants are added to the initialized algorithm parameters. If at step 5, $C_{\max}$ value was updated, then at step 6 pheromone update is added in accordance with (5).

6. Results and Discussion

When selecting a production scheduling algorithm there were considered algorithms that implement generalizations of the Johnson's algorithm [8], the branch and bound algorithm [9], the modified branch and bound algorithm, the ant colony optimization algorithm [10], the ant colony optimization algorithm with addition of the elitist ants [11].

Among all the considered algorithms, the branch and bound algorithm, the ant colony optimization algorithms and their modifications most effectively solve the production scheduling problem. They are also the methods of global optimization.

To evaluate the effectiveness of algorithms, the computational experiment of production scheduling task execution using the above considered methods was carried out. The optimality criteria of that experiment were overall processing time of technological operations minimization. The results of the calculations are shown in Figure 2, in more detail they are shown in Table 1.

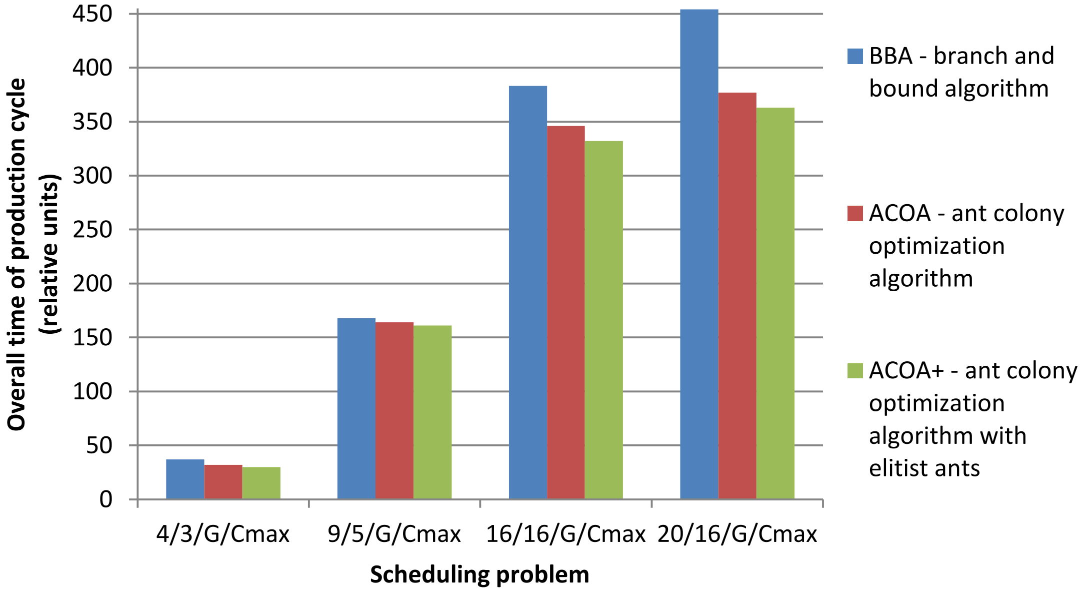

Figure 2. Dependence of production cycle duration on the number of technological operations

Table 1. Algorithms results

Number of technological operations	Number of machines	Overall time of production cycle (relative units)		
		branch and bound algorithm	ant colony optimization algorithm	ant colony optimization algorithm with elitist ants
4	3	37	32	30
9	5	168	164	161
16	16	383	346	332
20	16	454	377	363

The time taken to perform twenty operations by sixteen workers / devices is 17 percent less in case of modeling the schedule with the ant colony optimization algorithm than with the schedule based on a branch and bound algorithm. Application of the ant colony optimization with addition of the elitist ants shortens the overall process cycle time by 5 per cent while several times reducing the duration of calculations.

7. Conclusion

In this paper, we propose a production scheduling algorithm for assembly processes, using instrumental possibilities of ant colony optimization algorithms.

The results of optimization vary from 1 to 25 percent depending on the selection of the schedule formation algorithm, all other things being equal. These include reduction of time to perform all technological operations and time to calculate the schedule. Optimization of production schedules allows to reduce the total production cycle time, increase capital returns in general, capital productivity and output.

Numerous computational experiments on test examples have shown that efficiency of both classical ant algorithm and the one with the "elite" ants grows with the increase of the task dimension.

8. Acknowledgment

This work was supported by the Ministry of Education and Science of the Russian Federation in the framework of the Federal target program «Research and development of priority directions of development of the scientific-technological complex of Russia for 2014-2020» (agreement № 14.578.21.0116, unique ID project RFMEFI57815X0116).

References

[1] Shapiro J F 1993 Mathematical programming models and methods for production planning and scheduling *Handbooks in operations research and management science* **4** 371-443

[2] Jovanovic J R, Milanovic D D, Djukic R D 2014 Manufacturing cycle time analysis and scheduling to optimize its duration *Strojniški vestnik-Journal of Mechanical Engineering* **60(7-8)** 512-524

[3] Duncan W P 2011 Methods for reducing changeover times through scheduling

[4] Dessouky M M, Wilson J R 1991 Minimizing production costs for a robotic assembly system *Engineering costs and production economics* **21(1)** 81-92

[5] Ashlock D 2006 *Evolutionary Computation for Modeling and Optimization* Springer Science & Business Media

[6] Dorigo M, Blum C 2005 Ant colony optimization theory: A survey *Theoretical Computer Science* **344(2-3)** 243-278

[7] Pinedo M 1995 Scheduling: theory, algorithms and systems

[8] Pal S, Pattnaik P K 2016 Adaptation of Johnson sequencing algorithm for job scheduling to minimise the average waiting time in cloud computing environment *Journal of Engineering Science and Technology* **11 (9)** 1282-1295

[9] Ritt, M 2016 A branch-and-bound algorithm with cyclic best-first search for the permutation flow shop scheduling problem *Proc. IEEE International Conference on Automation Science and Engineering* (Fort Worth, US) 872-877

[10] Tsarev R Yu, Kovalev I V, Soloviev E V, Prokopenko A V, Knyazkov A N 2016 Ant colony optimization for multiversion software synthesis *International Journal of Soft Computing* **11(3)** 136-139

[11] Shi X, Kong D 2015 A multi-objective ant colony optimization algorithm based on elitist selection strategy *Metallurgical and Mining Industry* **7 (6)** 333-338