

JOB SHOP SCHEDULING IN MECHANICAL ENGINEERING

Pavel Kopecek

Department of Industrial Engineering
University of West Bohemia, Czech republic

kopecek@kpv.zcu.cz

Common ERP systems do not solve the support for workshop management. This paper shows why and under what conditions computer support for workshop schedules is necessary. Demands on the quality of operation plans, factors having an influence on the scheduling process and criteria for selection of appropriate operations are discussed. A verification program was developed for testing the proposed hypotheses. This program proved that a repeated random selection and simulation of the production process could be a good tool for a small number of machines and production tasks. There is no need for sophisticated scheduling for a great number of machines and tasks, operational plans are enough for this purpose.

Keywords

Scheduling, Production Process, ERP, Direct Control of Production in Mechanical Engineering, Discrete Optimization

1. Introduction

A detailed/precise/daily scheduling is made at the workshop level. If there are only a few scheduled items and workplaces and due times are relatively long, then a need for accurate schedules is more important than for a large number of short operations (the paradox of filling a rucksack with boxes).

2. Is a detailed schedule necessary?

Many enterprises have successfully implemented information systems. These systems cover a variety of areas from purchase, preparation of productions, real production, to expedition. Production planning processes terminate in operative plans, which fix times to particular operations with precision in weeks. There is an effort to balance these operation plans with workplace capacities.

An operative plan describes the loading of groups of interchangeable workplaces with operations with their due dates. It does not consider their accurate sequences of fulfilment at individual workplaces.

In most cases production control is enough. This refers for example to large batch production where a great quantity of a small number of types is produced. Another case is piece production with short operation times at a small number of workplaces.

A typical task, where there is a lack of strong software support for employees' decisions in production or for automatic generation of control input for automatic centres, is the sequencing of operations for a small or medium number of workplaces with operation times comparable with the durations of the shift times. It cannot be assumed that work-in-process can balance dynamic capacity inequality. This can be illustrated by the following example.

As much sand can be poured into a backpack as the volume of the backpack will allow. Of course, if boxes of different sizes are put into the backpack, then it depends on the sequence of loading. Only in very special cases does the sum of the box volumes approach the total volume of the backpack. This is demonstrated in Fig 1. in case a) a backpack is filled with sand, in case b) a backpack is filled properly with boxes with the same total volume as in case a), and in c) a backpack is filled randomly with boxes and the volume of the backpack must be much bigger than in case a).

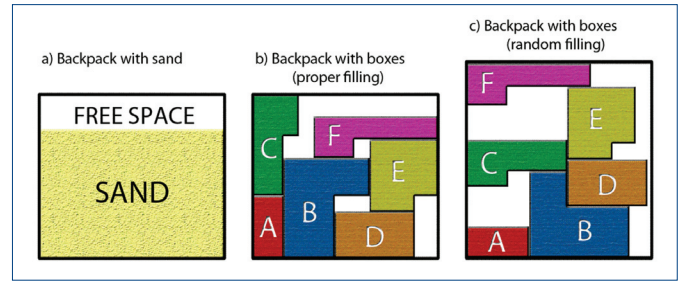


Figure 1. Backpack with sand and boxes

The same situation exists in production systems. If an operation on a certain batch at a certain workplace has not been started, then idle time at another workplace can be created. Hence a certain optimal sequence exists which minimizes retards and maximizes use of workplaces.

Large enterprise information systems do not contain such detailed schedules. They make a rough term specification and let the workshop managers (dispatchers, foremen) directly control production and require only feedback of operation fulfilment.

If operations need prepared tools, the machine adjusters must be told that they have to prepare and adjust tools to shorten the setting up time. If the input for the adjuster is only the operation plan then the increase in the number of tools could be enormous. In an ideal case, it is enough to prepare and adjust tools just before the real start of the operation

The same plan is necessary for stocking out and preparing material. In addition, when using bar and flat materials, it is necessary to combine a minimum lead time with an optimization of cutting plans.

Controlling material preparation, issuing and adjusting tools only according to an operative plan can either increase the work-in-process, drastically increase the need for single and special tools or in fact make preparation departments responsible for sequences of operation processing (operations with materials and/or tools have priority).

3. Properties of operation plans

3.1 Capacity balancing

An operation plan should be balance static. In practice this condition cannot always be fulfilled because there are workplaces which cannot be fully loaded but are necessary because of their technology. Then operations are delayed from the past.

When there is a strong static overload of workplaces, we cannot expect any feasible solution for detailed schedules, queues and increase of delays and it is not possible to meet due times.

When there is a strong static underflow, it does not depend on the quality of a detailed schedule, because tasks are fulfilled quickly and idle times of machines will appear.

3.2 Sliding and open nature of operating plans

An important property which is omitted in many theses is the sliding and open nature of an operation plan. It does not matter how fast all tasks of an operative plan are fulfilled, but how to process operations without delays against due times and without idle times and overtimes at static balanced workplaces. Operations are released from operation plans in batches (once a day, once a week, and once in ten days). A nice example of a sliding plan can be shown as a parable with a pipeline (Fig. 2).

New operations are released in batches from the operation plan. They are sequenced (pushed) by the detailed schedule into the production (pipeline) so as to avoid idle times (bubbles in the pipes). The production is a continuous process in time. Daily scheduling can be batch or real-time production event-based (the actual start, completion of the operation, incidence of a machine breakdown or end of the breakdown).

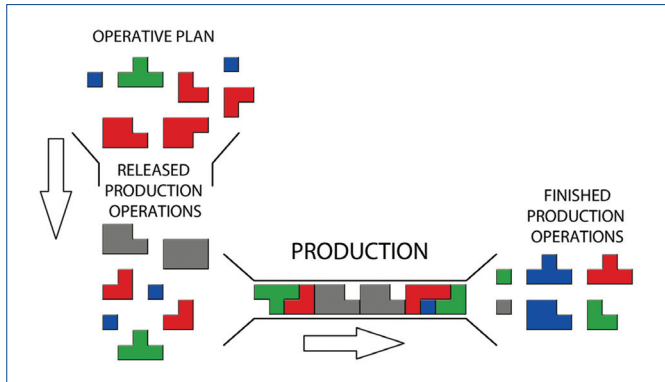


Figure 2. Daily plan as a way through a pipeline

Very often, two sets of operations are released from the operational plan. The first one includes operations that should be fulfilled in the next planning period (e.g. week). This performance should be mandatory for daily scheduling. The second one shows the prospects for further operations. It is hoped that these operations will be used to help solve specific divergent situations (workplace breakdowns, delays in cooperation, the possibility of overtime, sickness, qualified staff, and preparation of material and production equipment). Such a transfer plan is referred to as a "5 +5 plan", "plan binding and complementary".

4. Theoretical solutions

Whenever a task is specified, it is possible to precisely formulate it and try to find a method to solve it. If an analytical solution cannot be found, it is possible to try to find an iterative solution or a suitable simulation method which leads to the goal. If this fails, then some heuristic solutions could be found. In this case we usually stop trying to achieve the optimum solution, but look for solutions that are not bad and are as close to the optimum as possible.

The first option is to check all possibilities. Many computer users believe non-critically in line with current trends, that the possibilities of computer technology are growing exponentially, with the increasing size of memory and CPU power. While it is not known yet where this growth results in depletion of the physical properties of matter, it can be assumed that this growth will continue in the meantime.

This is good news. But the bad news comes from the theory of computers. There, algorithms are evaluated according to their complexity. Unfortunately, scheduling algorithms are very complex and known as 'NP-complete'. There is evidence that today's methods for calculating a relatively small number of operations would exceed the age of the universe – this is a combinatorial algorithm, in fact a factorial estimation of time.

It is possible to solve a few problems algorithmically. All others can be divided into two groups:

- Where a detailed schedule does not make sense, because a large number of scheduled tasks and workplaces leads to a convergence of static and dynamic usage of workplaces (queues are so long that it does not matter what we choose);
- Where a detailed schedule is the point, then a heuristic solution could be proposed that is certainly better than the worst, in a better case better than average, and at best a little worse than the optimum.

There are special cases, for example the Johnson algorithm [Stevenson 2009] for two workplaces A and B, and tasks of two operations using first A and then B. The principle of the Johnson algorithm is to find the shortest time. If it is at workplace A, then the task is scheduled as first, if it is at workplace B, then it is scheduled as last. The scheduled task is removed from the set of tasks and the process is repeated. The solution has no idle times for workplaces.

Most books about operations management omit tasks of precise scheduling and leave these problems to be solved by workshop managers. Some authors have proposed modern approaches ([Ridzon 2002] – Genetic algorithms, [Ghoniem 2002, Knyttl 2010] – Shifting Bottleneck) with good results. In many cases they use makespan criteria and do not respect the sliding nature of operative plans. Also input parameters (capacities, overtimes, etc) are considered to be constant even though they may be changed by creative workshop managers.

It is also possible to use a brute force solution; this means checking all possibilities. Because $N!$ solutions exist; this enormous number of solutions could be reduced to a number which can be computed in a minute, let us say 10 000. Random permutations could be checked and the best one chosen.

However, there is no sure guarantee that a good solution can be found. If it is assumed that there is a one in a thousand chance of finding a relatively good solution, there is a 999 in a thousand chance of not finding a good solution. For 10 random selections there is still a 990 in a thousand chance of finding a bad solution, for 100 random selections 904 in a thousand, for 1 000 only 36 % and for 10 000 random selections only 0.005 % possibility of a bad solution. It is possible to afford 10 000 attempts.

5. Criteria of schedule quality

- The basic criterion is the minimum lateness or tardiness of the latest necessary end of a production batch;
- The next criterion is the minimization of lateness of operation due dates;
- There is also a trend towards maximum use of all workplaces, no idle times;
- Sometimes the term of finishing the last operation is considered (makespan).

6. Influences on schedule quality

If an operative plan is not balanced properly in capacities and long periods, no changes to the sequences of operation can correct it.

At higher levels of a planning process, interchangeable workplaces are considered. In a job shop schedule, the planning algorithm must consider special tool equipment at workplaces, various workplace accessories and the qualifications of the workplace operators.

Also tool management must be considered. It is necessary to distinguish between common tools with short preparation times and enough tools, a limited number of tools and tools with long preparation times (various assembled tools).

Some detailed information is not included in the production database. The role of the human factor is unsubstitutable. It can decrease with an increasing ratio of automatic CNC machines, automatic interoperation transport and high-quality ERPs filled with authentic information. But workshop managers must operatively react to assembly department demands; solve scrap compensation, specify non-changeable workplaces, qualifications and presence of machine operators. Workshop managers can increase or decrease priorities of batches or operations, specify the workplaces in routings, insert new operations in routings, divide batches and insert or remove operations from scheduled queues at detailed workplaces.

7. Decisive criteria for selecting an operation

When creating a queue these factors can be considered:

- date and time when an operation was prepared for scheduling (FSFS, first come, first served);
- shortest processing time (SPT);
- earliest due date (EDD);
- critical ratio (remaining total time to due date divided by remaining processing time);
- static priority of a batch (rush);
- static priority of an operation;

- waste class;
- operation time;
- operation due date;
- sum of remaining operation times;
- number of remaining operations;
- total time from the start of the first operation (inclusive waiting);
- flag of prepared tools;
- immediate static usage of an workplace;
- critical workplace in the operation chain.

The weights of these factors should be a topic for research and parameters for fine schedule adjustment.

8. Hypothesis

- The more operations and workplaces that are scheduled, the less is the need for a detailed schedule (example with the rucksack). For many large production workshops, an operative plan is adequate for workshop control purposes;
- A random selection of permutations could be a good solution for a small number of workplaces and operations (say 10 workplaces and 20 tasks with 100 operations);
- The quality of workshop management is high, so some schedule decisions and parameters for automated scheduling could be made in a creative way.

9. Verification program

The main idea for the verification program is that if a set of well-balanced production data has been prepared then the program can also find a schedule which is good, better or a little worse than the original input data. It can be assumed that layouts of machines and transport times are ignored. Transport times are considerably shorter than processing and waiting times.

The goal of the verification program is to prove that using simulation and a random selection of operations in comparison with a deterministic method can lead to acceptable results.

The program is fully controlled by input data. As an example, a solution for 10 batches of parts, 40 operations and 6 various workplaces with 10 machines (one workplace can be represented with more fully reciprocal changeable machines) is presented.

The algorithm is based on simulation; in the simulated time the machine which will be free first is found. The most suitable active

operation is selected for this machine, and the end of the operation is estimated in the simulated time, simultaneously the next operation in the routing of the batch becomes active in the simulated time. This is repeated until all operations are scheduled.

A suitable operation is selected according to a scheduled parameter:

- Due date;
- Shortest operation time;
- Longest operation time;
- Random (also a number of random experiments is given).

The criterion for the schedule quality is the sum of (linear and quadratic) delays of operation due dates (tardiness). The verifying program can be run at full speed, with animation or step by step. The interface of the program is shown in Fig 3.

The horizontal axis represents time in hours. The lower part shows the operation plan for 10 batches of parts and their operations. The upper part shows individual machines (orange), capacities of machines (gray) and coloured scheduled operations.

10. Experiment result

What conclusion can be made from the experiments?

The criterion of shortest (value of this criterion is 29) and longest (value 42) operation times gives according to expectation or worse results than according operation due dates (value 17).

For 50,000 random schedules, criteria results lie in the range 0 – 121, zero was achieved in 8 cases and values less than 15 (better than in the case of due dates) in 94 cases. Frequencies of criteria values are displayed linearly and logarithmically in Fig 4.

Also for 1,000 random experiments, the results were much better than when using deterministic criteria. Concurrently, it is possible to see that one random selection has a high probability of a very bad schedule.

More than 100 experiments with various data and numbers of random selections were performed. It could be assumed that this simple algorithm based on random selection and a relatively small number of repetitions could be a good way of solving practical tasks with a small number of machines and tasks.

The simple heuristic solution (random selection) is very fast (10 machines, 6 workplaces, 10 batches, 40 operations and 1000 repeated schedules needs 25 seconds on Pentium 4, 2.27 GHz, 1GB RAM).

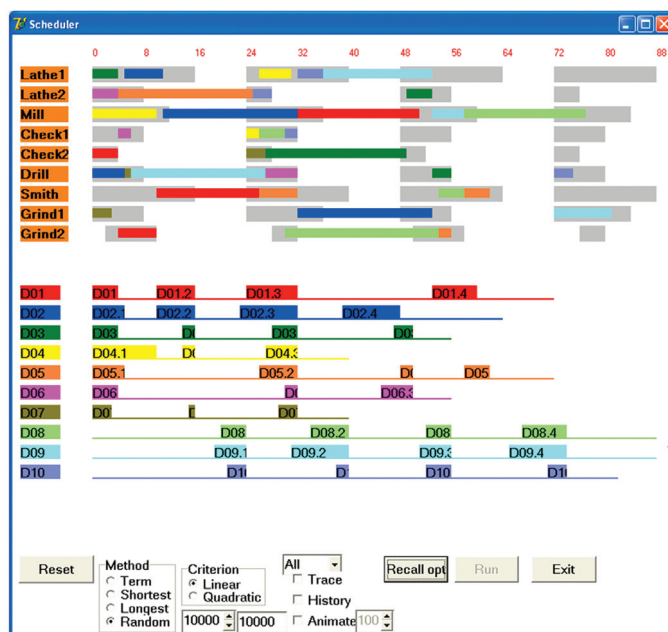


Figure 3. Graphic interface of scheduling program

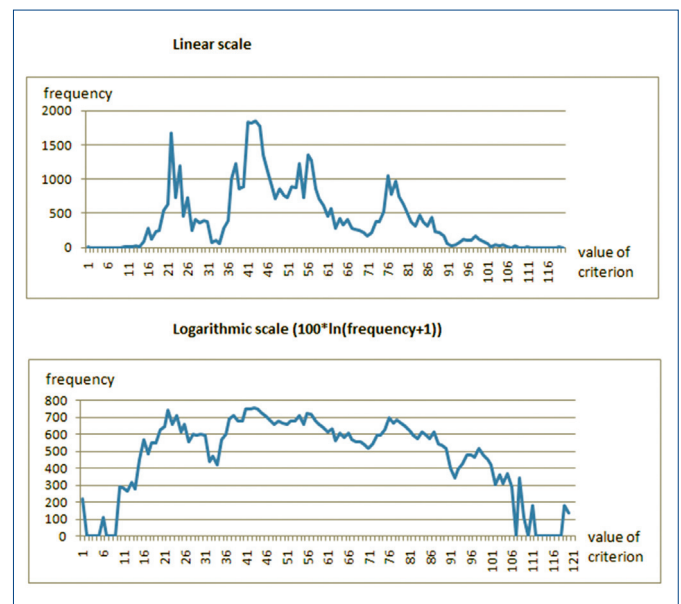


Figure 4. Results of scheduling

[Knyttl2010] states that heuristics give much better results than scheduling rules. Experiments having been made with the developed validation program are in agreement with this contention.

11. Conclusions

Although the first successful attempts at a dynamic schedule were made in the 1970s, modern enterprise information systems have only been introduced en masse in the last ten years. When combined with advanced programming technology, this creates the preconditions for the successful development of tools for operational management of production. The subject of further research should be to find selection criteria, including the impact of significant factors, modification and verification of the scheduling algorithm to minimize delays to scheduled due dates.

Previous results support the hypothesis that a small number of operations in relation to the number of machines are needed to optimize the schedule.

It was experimentally demonstrated that there is a better solution than the selection of scheduled operations according to the operation due date. This solution can be found heuristically.

A tool for creating daily schedules for experimental purposes has been developed. This tool allows tracking and analyzing the schedule for developmental and educational purposes, and to conduct experiments with different data samples.

The developed tool will be the basis for the final scheduling program. We expect to preserve the internal data structure, develop a graphical interface for user-friendly data entry, import data from ERP and print results. Hypotheses should be proved using real data from enterprises.

Acknowledgments

This article was prepared with support from the Internal Science Foundation of the University of West Bohemia SGS-2010-065.

References

- [Ghoniem 2002] Ghoniem, A., S.: Static and Dynamic Job-Shop Scheduling Using Rolling-Horizon Approaches and Shifting Bottleneck Procedure, Thesis Submitted to the Faculty of the Virginia Polytechnic Institute and State University, Nantes, France, 2002
- [Gregor 2007] Gregor, M. at all: Production Planning, University of Zilina, 2005, ISBN 80-8070-427-9, (in Slovak)
- [Knyttl 2010] Knyttl, J.: Dynamic Scheduling of Production, University of West Bohemia, 2010, (in Czech)
- [Pinedo 2009] Pinedo, M., L.: Planning and Scheduling in Manufacturing and Services, Springer, 2009, ISBN 978-1-4419-0909-1
- [Ridzoň 2002] Ridzoň, M.: Program for Solution of Production Scheduling Using Evolution Algorithms, Diploma Thesis, Technical University of Kosice, 2002, (in Slovak)
- [Stevenson 2009] Stevenson, W, J.: Operations management, McGraw-Hill, 2009, ISBN 978-0-07-337784-1
- [Wannenwetsch 2010] Wannenwetsch, H.: Integrierte Materialwirtschaft und Logistik, Springer, 2010, ISBN 978-3-540-89772-9

Contacts:

Assoc. Prof. Ing. Pavel Kopecek, CSc.
Department of Industrial Engineering, University of West Bohemia,
Univerzitní 8, 306 14 Pilsen, Czech Republic
tel.: +420 377 638 420, fax.: +420 377 638 402
e-mail: kopecek@kpv.zcu.cz, <http://www.kpv.zcu.cz>