Roser, Christoph, Masaru Nakano, and Minoru Tanaka. “Tracking Shifting Bottlenecks.” In Japan-USA Symposium on Flexible Automation, 745–50. Hiroshima, Japan, 2002.

TRACKING SHIFTING BOTTLENECKS

Christoph Roser
Masaru Nakano
Minoru Tanaka

Software Science Laboratory
Toyota Central Research and Development Laboratories
Nagakute, Aichi, 480-1192, JAPAN
croser@robotics.tytlabs.co.jp
nakano@robotics.tytlabs.co.jp
tanaka@robotics.tytlabs.co.jp

ABSTRACT

The throughput of a manufacturing system is constrained by the bottlenecks in the manufacturing system. Improving the bottlenecks improves the manufacturing system. Finding the bottlenecks is nontrivial for static manufacturing systems. Detecting bottlenecks in flexible manufacturing systems is even more complicated, as bottlenecks in flexible manufacturing systems change frequently. This paper provides a novel method for detecting bottlenecks and the shifting thereof in flexible manufacturing systems based on the duration a machine is active. The presented active duration method determines the bottleneck based on the duration a machine is active without interruption. The method detects not only the short term shifting bottlenecks but also determines the long-term bottleneck in form of the probability of a machine being the bottleneck. The method is very robust, easy to apply and able to detect the primary and secondary bottlenecks in a wide range of manufacturing systems. The method is demonstrated using different examples and compared to conventional examples. The measurement of the likelihood of a machine being the bottleneck aids in the decision-making regarding the allocation of the available resources.

1 INTRODUCTION

This paper describes a method to detect and monitor the bottleneck in steady state and non-steady-state manufacturing systems. A bottleneck is the stage in a manufacturing system that has the largest effect on slowing down or stopping the entire system. The paper further distinguishes between a momentary bottleneck, describing the bottleneck at any given point in time, and an average bottleneck, describing the bottleneck behavior over a selected period of time. In order to improve the momentary or average system throughput, the momentary or average bottleneck respectively has to be improved. Therefore, it is of interest to determine the bottleneck in order to improve the throughput of the production system by improving the throughput of the bottleneck, also known as the theory of constraints (Blackstone 2001; Goldratt 1992). Yet, finding the bottleneck is no trivial task, and (Cox 1997) for example simply recommends that “... the best approach is often to go to the production floor and ask knowledgeable employees...”.

The detection of the bottleneck is complicated as the bottleneck shifts between different machines, depending on the preceding random events. A non-bottleneck machine may become a bottleneck, for example due to a machine failure, and similarly a bottleneck machine may become a non-bottleneck machine. Over longer periods of time, a system therefore may not only have one primary bottleneck, but also secondary and tertiary bottlenecks, i.e. machines which are also occasional bottlenecks, yet to a lesser extent than the primary bottleneck. The method presented in this paper considers the shifting of both momentary and average bottlenecks.

There are a number of methods available to find the bottleneck for production systems. One approach measures the utilization of the different machines of the production system (Law 2000). The machine with the highest utilization is considered to be the bottleneck. However, the utilizations of different machines are often very similar, and it cannot be said for sure which machine is the bottleneck. Longer simulations may be required to generate meaningful and accurate results. Furthermore, this method is limited to steady state systems. The utilization method detects only the average bottleneck over long periods of time and is unable to determine the momentary bottleneck, making it unsuitable to detect and monitor shifting bottlenecks. (Adams 1988) uses disjunctive graphs to detect the bottleneck in order to optimize the scheduling in a shifting bottleneck procedure. (Uzsoy 2000) compared the shifting bottleneck procedure to the theory of constraints.

Another frequently used method analyses the queue lengths of the machines in the production systems. In this method, either the queue length or the waiting time is determined, and the entity with the longest queue length or waiting time is considered to be the bottleneck. This method has the advantage that a momentary bottleneck can be determined by simply comparing the queue lengths or waiting times. The average bottleneck can also be detected using the average queue length or waiting time. Yet, this method also has a number of other shortcomings. First and foremost, many systems have only limited queue lengths or no queues at all, in which case the queue length cannot be used to detect the bottleneck. Furthermore, if the batch sizes vary for different machines throughout the production system, the waiting time or queue length may in some cases give incorrect results. In addition, the waiting time of the parts or the queue length is a heavily dependent factor and not identically and independent (i.i.d.) distributed and it is difficult to estimate the accuracy of the average queue length or waiting time over periods of time. Finally, the queue length also fluctuates frequently and is an unreliable indicator of the bottleneck.

In summary, there are various methods available, yet all of them have one or more disadvantages, for example in terms of reliability, usability, resolution and accuracy. The method described below is able to overcome these disadvantages, and allows the detecting and monitoring of both momentary and average bottlenecks over any selected period of time. The method is also easy to implement, requiring no knowledge of the structure of the production system.

2 BOTTLENECK DETECTION METHOD

This section describes the underlying idea of the active duration as a measurement of the constraint. This is a continued development and improvement based on the method of the average active duration (Roser 2001). The active duration will then be used to detect shifting momentary bottlenecks and also the average bottlenecks.

2.1 The active duration

The presented method is based on the duration a processing machine is active without interruption. As a first step, it is necessary to group all possible machine states into two groups, being either active states or inactive states. A state is active whenever the machine may cause other machines to wait. For example working on one part may cause a subsequent idle machine to wait for the completion of the part, or a machine under repair may block previous machines. A state is inactive if the associated machine is not active but instead waiting for the completion of another task, for example the arrival of a part or service, or for the removal of a part. Table 1 shows a possible grouping of selected states for different entities of a production system into active and inactive.

Table 1: Selected active – inactive states

Machine	Active	Inactive
Processing Machine	Working, in repair, changing tools, serviced	Starving, blocked
Automated Guided Vehicles (AGV)	Moving to a pickup location, moving to a drop off location, recharging, being repaired	Waiting
Factory Worker	Working, on scheduled break	Waiting

Figure 1 shows an example of a series of active (work, repair, tool change) and inactive (waiting) states of a machine during a brief period of a simulation.

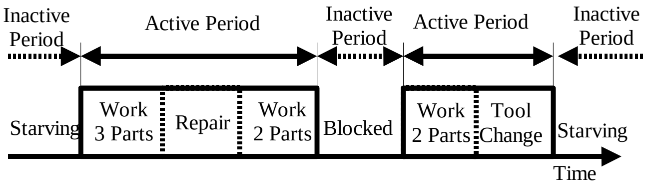

Figure 1: Active periods of machine during simulation

The bottleneck detection method compares the durations of the active periods of the different machines. If the analysis is based on simulation data or historical data, it is possible to determine the durations of all active periods for all machines. However, if the analysis is used for real time monitoring, the future is unknown and the durations of the active periods are known only until the present. In this case, the active duration is measured until the present and may be updated if further information becomes available with time.

2.2 The momentary bottleneck

The underlying idea of the method is that at any given time the machine with the longest uninterrupted active period is the momentary bottleneck at this time. In an interconnected production system, machines block and starve each other. If a machine is active, it is neither starved nor blocked. The longer a machine is active without interruption, the more likely it is that this machine blocks or starves other machines in the production system. The machine with the longest uninterrupted active period therefore has the biggest impact onto starving or blocking the other machines, therefore being the largest constraint a.k.a. the largest bottleneck. The following method describes how to determine which machine of a production system is the sole bottleneck or part of a shifting bottleneck at any time. The method is visualized in Figure 2 using a simple example consisting of only two machines. The figure shows the active periods of the machines over a short period of time. At the selected time *t*, both machines M1 and M2 are active. Yet, as M1 has the longer active period, M1 is the bottleneck machine for the time *t*. At the end of the current bottleneck period, M2 is active and has the longest active period. Therefore, the subsequent bottleneck machine is M2. During the overlap between the current bottleneck period and the subsequent bottleneck period the bottleneck shifts from M1 to M2. Now, M2 is the bottleneck machine. Similarly, at the end of the bottleneck period of M2, the bottleneck shifts back to M1. Processing all available data using this method shows at what time which machine is the bottleneck machine and when the bottleneck is shifting. If no machine is active then there is no bottleneck at this time. This allows the detection and monitoring of the sole and shifting bottlenecks at all times.

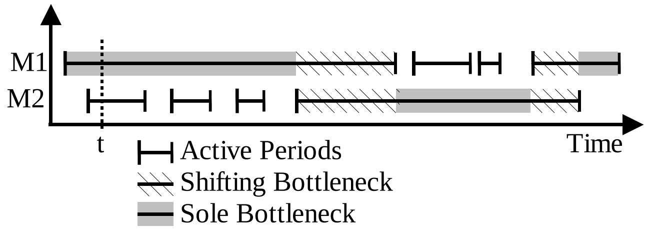

Figure 2: Shifting bottlenecks

2.3 Flexible manufacturing

The method as presented so far allows the detection of the momentary bottlenecks at all times based on the active period, where the active period is measured as the duration between two inactive periods. This causes some problems for real time bottleneck detection in flexible manufacturing systems, as it is not known when the currently active machine will be interrupted. In this case, the active period has to be substituted by the active period so far, i.e. the time between the last inactive period and the current time. This is visualized in

Figure 3, where the durations of the active periods after the present time are unknown. Therefore, it cannot be said if the bottleneck is switching from machine M2 to machine M1 or not.

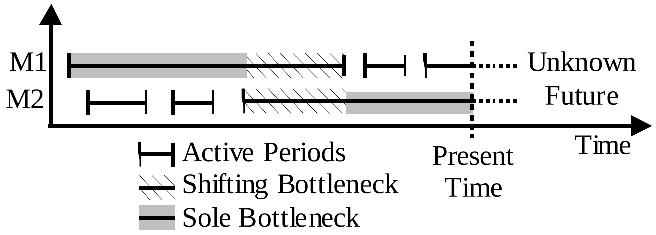

Figure 3: Real-time shifting bottlenecks

However, it is known that machine M2 is currently a bottleneck, although it is unknown if it is a sole or a shifting bottleneck. Therefore, available additional resources may be allocated to machine M2 in order to improve the overall system performance using a ?hase the bottleneck” approach. Furthermore, non-bottleneck machines may need fewer resources without impacting the system performance.

There are numerous possibilities to increase or decrease the system performance in response to the bottlenecks of the production systems. For example, available workers may be allocated to the bottleneck machines. On the other hand, cutting speeds of non-bottleneck machines may be reduced, increasing the lifetime of the tools without impacting the system performance. The specific implementation depends on the specific manufacturing applications. The presented active period method therefore provides valuable bottleneck information, which can be used to reallocate flexible resources in real time on the factory floor.

2.4 The average bottleneck

The above method detects and monitors the momentary bottleneck. This allows the flexible allocation of resources to improve the overall system performance. However, in many cases the available resources are not flexible, but rather a longer term investment, as for example a machine hardware improvement, or the purchase of additional machines. In this case, knowledge of the average bottlenecks is desired. This section describes how to adapt the active period method to determine the average bottleneck over a period of time.

First, the available data is analyzed and the momentary bottlenecks are determined. Next, the percentage of time a machine is the sole bottleneck machine and part of a shifting bottleneck is calculated for the selected period of time. These percentages also represent the likelihood of the machines being the bottleneck for the selected period of time.

Figure 4 visualizes this method using the example with two machines as shown in Figure 2. The percentages of the machines being the sole bottleneck or the shifting bottleneck have been measured over the period of time shown in Figure 2. The larger the percentages, the larger is the effect of the respective machine onto slowing down or stopping the system. Machine M1 is the sole bottleneck more often than machine M2, and is also involved in a number of shifting operations. Machine M2 is the smaller constraint, i.e. a secondary bottleneck, having being the sole bottleneck for a smaller percentage of time. Therefore, machine M1 is more likely to be the bottleneck than machine M2. Overall, an improvement of the performance of M1 would yield a larger overall improvement of the system than an improvement of M2, as M1 is the primary bottleneck during the selected period of time. However, if there are cost effective improvements of machine M2 available, it might be beneficial to also implement these improvements.

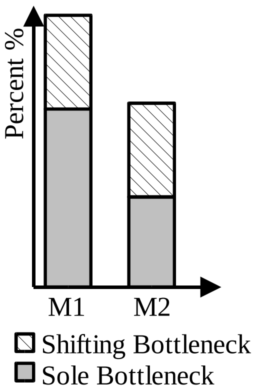

Figure 4: Average bottleneck

3 COMPUTATIONAL EXAMPLES

The method was implemented in the software tool GAROPS Analyzer to analyze the simulation data from the GAROPS simulation software (Kubota 1999), (Nakano 1994). The software tool analyses the machine status information over time and creates an excel file containing a statistical description of the simulation including the change of the sole and shifting momentary bottlenecks over time and also the sole and shifting average bottlenecks of the complete simulation.

This section will describe two computational examples. The first example is a flow shop with four stations, taken with small modifications from (Lawrence 1994). The second example is a complex branched system with seven machines and two different part types. (Lawrence 1994) also devised a bottleneck shiftiness measure β as shown in equation (1), where c_v is the coefficient of variation of the bottleneck probability of the different machines and n is the number of machines in the system. The bottleneck shiftiness measure β ranges from zero for a system with a unique bottleneck to one for a system where all machines are equally likely to be the bottleneck. The bottleneck shiftiness measure can also be applied to the active duration method and will be utilized in the examples below.

$$\beta = 1 - \frac{c_v}{\sqrt{n}} \tag{1}$$

3.1 Flow shop

The flow shop example has an exponential inter arrival rate with a mean inter arrival time of 1.25s. The processing times of the four machines has an exponential distribution with a mean service rate · of 1s for machines M1, M2, and M4, and 1.1s for machine

M3. All parts are processed by all machines in sequence. The utilization p_i is 80% for machines M1, M2, and M4, and 88% for machine M3, which is subsequently the overall bottleneck. Figure 5 shows the layout of the flow shop system.

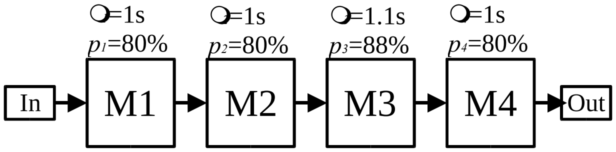

Figure 5: Flow shop layout

The simulation was analyzed using the GAROPS Analyzer. The percentages of the time a machine was the sole bottleneck and the percentages of time a machine was part of a shifting bottleneck have been analyzed using the active period method as described above.

Table 2 shows the results of the simulation. For each machine, the utilization is given in column two. The percentages of the time a machine is the sole bottleneck and the percentage of the time a machine is part of a shifting bottleneck as described above are given in column three and four. The sum of the shifting and sole bottleneck percentages is given in the last column. The last row shows the bottleneck shiftiness measure β for the different bottleneck measurements according to equation (1). The results of Table 2 are also visualized in Figure 6.

Table 2: Flow shop simulation results

Machine	Utilization	%Sole Bottleneck	%Shifting	%Sum
M1	80.1%	12.67%	20.43%	33.1%
M2	80.2%	6.73%	15.93%	22.7%
M3	88.0%	32.54%	29.27%	61.8%
M4	80.0%	7.25%	15.23%	22.5%
Shiftiness Measure β		0.59	0.84	0.74

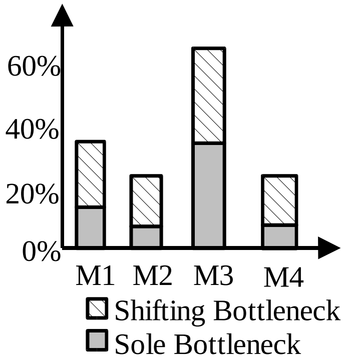

Figure 6: Shifting and sole bottleneck likelihood

Machine M3 is clearly the bottleneck, as all measures in Table 2 indicate M3 as the main bottleneck. Machine M3 is the sole bottleneck for about 1/3rd of the time, and a shifting bottleneck for another 1/3rd of the time. This makes M3 the primary bottleneck of the system. However, machines M1, M4 and M2 are secondary bottlenecks, and an improvement of these machines may also improve the system performance.

3.2 Complex example

The complex example consists of a branched system with seven machines, different buffer sizes, and two different part types as shown in Figure 7. The simulation was analyzed using the GAROPS Analyzer.

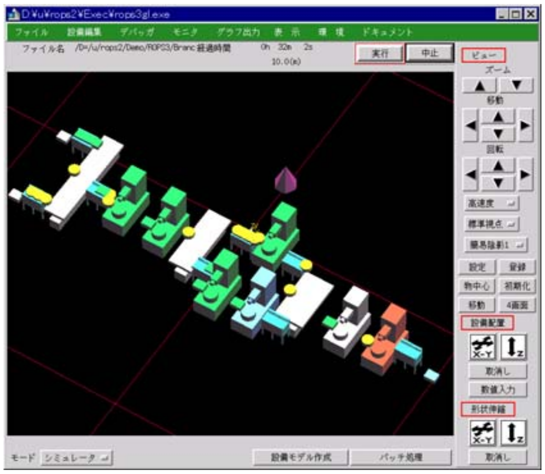

Figure 7: Complex example GAROPS screenshot

Figure 1 shows the sole and shifting bottlenecks during a short period of the complex example. It can be seen how the bottleneck changes between different machines over time.

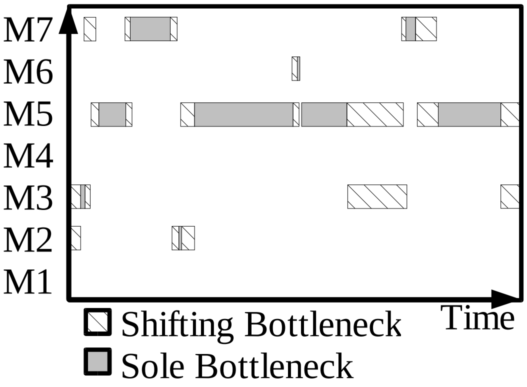

Figure 8: Shifting bottlenecks

Figure 9 shows the utilization of the seven machines, including the ranges of the 95% confidence intervals. The potential primary bottlenecks are shaded. Based on this simulation, it cannot be said for sure which machine is the primary bottleneck. It appears, that M5 is the primary bottleneck, yet the difference to M3 is too small, the 95% confidence intervals overlap and there is no statistical significance. Therefore, it is difficult to detect the primary bottleneck by measuring the utilization, let alone secondary and tertiary bottlenecks.

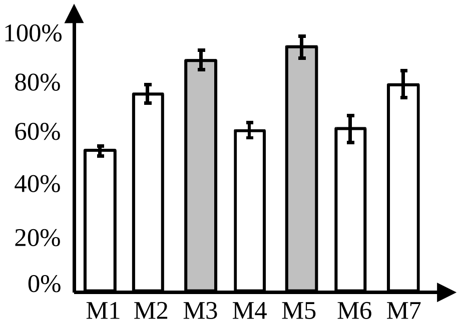

Figure 9: Utilization of complex example

Figure 10 and Table 3 show the result of the bottleneck detection using the active period. Here the results are very clear, showing that M5 is indeed the main bottleneck, being a sole bottleneck for 45% of the time and a shifting bottleneck for 37% of the time, i.e. M5 is part of a bottleneck for 82% of the time. Calculating the 95% confidence intervals reveals that the results are statistically significant and M5 is indeed the bottleneck. This example also indicates that M3 is a potential secondary bottleneck and M7 is a potential tertiary bottleneck.

Table 3: Complex example simulation results

Machine	Utilization	%Sole Bottleneck	%Shifting	Sum
M1	54%	0.0%	0.1%	0.1%
M2	76%	2.2%	3.3%	5.6%
M3	89%	1.2%	29.3%	30.5%
M4	62%	0.1%	0.0%	0.1%
M5	94%	45.1%	37.3%	82.4%
M6	63%	1.5%	3.6%	5.1%
M7	80%	7.0%	12.5%	19.5%
Shiftiness β		0.24	0.54	0.46

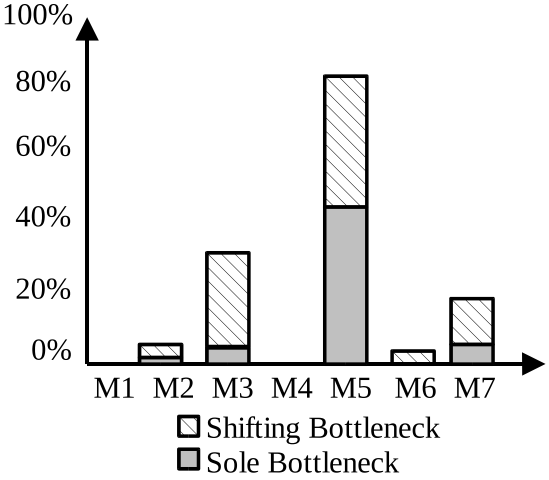

Figure 10: Average sole and shifting bottlenecks of complex example

In summary, an improvement of the performance M5 would improve the overall system performance. Machines M3, M7 and M2 may also be considered for improvements depending on the trade off between the cost of the improvement and the benefit of the improved system performance. Furthermore, the bottleneck analysis determines that an improvement of M1, M4 and M6 is unlikely to increase the system performance, and no resources should be invested into an improvement of M1, M4 and M6 at this time.

4 SUMMARY

The active period method is very suitable to detect both the momentary bottleneck and the average bottleneck. The accuracy, reliability and usability of the active period method exceed the other available bottleneck detection methods. For example, the measurement of the queue length or waiting time in order to detect the bottleneck cannot be used if the queue lengths are limited. In addition, the queue length may fluctuate frequently, complicating a reallocation of the resources in a ?hase the bottleneck” approach.

Using the utilization as a bottleneck detection method may give inaccurate results for the detection of the primary bottleneck, and it is usually impossible to detect secondary and tertiary bottlenecks.

The active period method as presented in this paper, however, is a very flexible tool and can be used for a wide range of systems as for example production systems, computer networks or traffic systems. The method is easy to apply, and the required data is usually readily available. As the active period is measured directly at the machine, there are no errors due to outside limitations as for example in the indirect measurement of the machine activity using the queue length. Knowing the likelihood of each machine to be the bottleneck aids in making a trade off between the effort of adding capacity and the benefits of improved throughput.

Research is in progress to adapt the active period method for the optimization of the production systems.

Roser, Christoph, Masaru Nakano, and Minoru Tanaka. “Tracking Shifting Bottlenecks.” In Japan-USA Symposium on Flexible Automation, 745–50. Hiroshima, Japan, 2002.

REFERENCES

Adams, Joseph, Balas, Egon, and Zawack, Daniel 1988. The Shifting Bottleneck Procedure for Job-Shop Scheduling. *Management Science*, 34, 391-401.

Blackstone, John H. 2001. Theory of constraints - a status report. *International Journal of Production Research*, 39(6), 1053-1080.

Cox, James F. III, and Spencer, Michael S. 1997. *The Constraints Management Handbook.* Boca Raton, Florida: CRC Press - St. Lucie Press.

Goldratt, Eliyahu M. 1992. *The Goal: A Process of Ongoing Improvement.* North River Press.

Kubota, Fumiko, Sato, Shuichi, and Nakano, Masaru 1999. Enterprise Modeling and Simulation Platform Integrated Manufacturing System Design and Supply Chain. In *IEEE Conference on Systems, Man, and Cybernetics*, 511-515, Tokyo, Japan.

Law, Averill M., and Kelton, David W. 2000. *Simulation Modeling & Analysis.* McGraw Hill.

Lawrence, Stephen R., and Buss, Arnold H. 1994. Shifting Production Bottlenecks: Causes, Cures, and Conundrums. *Journal of Production and Operations Management*, 3(1), 21-37.

Nakano, Masaru, Sugiura, Norio, Tanaka, Minoru, and Kuno, Toshitaka 1994. ROPSII: Agent Oriented Manufacturing Simulator on the basis of Robot Simulator. In *Japan-USA Symposium on Flexible Automation*, 201-208, Kobe, Japan.

Roser, Christoph, Nakano, Masaru, and Tanaka, Minoru 2001. A Practical Bottleneck Detection Method. In *Winter Simulation Conference*, Arlington, Virginia.

Uzsoy, Reha, and Wang, Cheng-Shuo 2000. Performance of decomposition procedures for job shop scheduling problems with bottleneck machines. *International Journal of Production Research*, 38(6), 1271-1286.