Advanced Finite Capacity Planning and Scheduling – Theoretical vs. Practical

Andrew R. Gilman
WATERLOO MANUFACTURING SOFTWARE

Introduction

Interest in advanced finite capacity planning and scheduling (APS) is clearly growing. Why? What has changed that is driving the surge in interest? Often it is the desire to serve customer needs better, to use customer service as a competitive weapon. Companies who were unable to compete on cost or quality are no longer in the game. Responsiveness is the new dimension of competition.

Customer service encompasses many kinds of interactions between a company and its customers. For our purposes we will concentrate on responsiveness as it relates to the customer's order (i.e., what was ordered, how much was ordered and when it can be delivered). In this sense, better customer service is characterized by offering competitive lead times, dealing with just-in-time quantities, and responding when the customer needs to change items, quantities or dates.

Many recognize APS as an enabling technology to make manufacturing more responsive to customer needs. There are three major reasons.

1. First, in order to offer a delivery commitment, you must be able to see how your capacity is booked.
2. Second, if a change is requested or a problem occurs, you must know how it will affect existing customer commitments.
3. Finally, the shop floor must execute to meet the delivery commitments without sacrificing efficiency.

Companies are learning that traditional manufacturing software and ERP/MRP II systems were not designed with this in mind. Many of these companies are turning to APS.

Because of this "gap" in ERP/MRP II, finite capacity planning and scheduling is typically handled manually, either through the laborious manual calculations required to develop a schedule, or through endless meetings and order expediting in the absence of a true schedule. As companies transition from these manual methods to computerized APS, they must bear in mind the essence of the scheduling challenge. Successful manufacturers will be those who can continually manage the tradeoffs between customer service and efficiency. The production schedule is the road map for how they plan to do it. Therefore, they must understand how this balance can be achieved with commercial APS systems. This article examines some major approaches found in commercial systems in light of the pressures that are causing companies to take action.

Balancing Service and Efficiency

Improving customer service creates one kind of pressure, to do whatever it takes to satisfy the customer. However, in this age of cost consciousness and downsizing, most firms also face pressure to do more with less. They must use existing equipment, people, and tooling as efficiently as possible and only add new equipment, people and tooling when absolutely necessary. Concentrating on either customer service or efficiency alone can be disastrous. As the pressure to improve on both fronts increases the importance of scheduling becomes obvious.

Many companies try to get by with no true scheduling. Planners release work and manufacturing supervision determines what jobs to work on based on how they are measured, typically utilization and throughput. They are concerned only with the manufacturing efficiency side of the problem. "Hot lists" and expediters represent the customer service point of view. Management's time is consumed in production meetings where hot jobs, identified by customer service or sales, are reviewed and prioritized. As expediting occurs manufacturing efficiency is disrupted. Setups are torn down, non-standard or outside processes are employed, and overtime is expended. Worse still, the expedited job of today displaces other jobs that will become the hot jobs of tomorrow.

When the situation becomes bad enough, a scheduling function might be established to provide accountability for both customer service and efficiency. The result is usually a manual schedule that covers only a few days and a few key work centers. Limited though it may be, it is a step in the right direction.

Ultimately, there is recognition of both how important and how difficult the scheduler's job has become. Manual scheduling is often so time consuming that only one possible schedule can be produced and no other scenarios can be tried. Worse still, when order changes come along or a machine goes down, the whole thing may have to be re-worked. At this point the organization begins to look for a system to improve the process.

Finite Capacity Scheduling

The manual process a scheduler uses <u>is</u> finite capacity scheduling. He manually accounts for capacity as he lays out the schedule. With hundreds or thousands of jobs, it becomes painfully obvious that a computer is better suited to the task of accounting for how capacity is allocated.

That, of course, is the role of APS systems. APS is a broad term however. It leaves open the question of how capacity is measured and how jobs are assigned to that capacity.

How Is Capacity Measured?

The first major distinction between the approaches to APS is in how capacity is measured. The simplest representation of capacity, used mainly in finite loading features imbedded in some ERP/MRP II systems, is based on the work center concept. In this representation a work center is said to have a certain amount of capacity per time bucket, for example 24 hours per day.

There are a number of problems with this model. If the work center contains say three machines, each available 8 hours (therefore 24 hours of capacity), and a 16-hour job arrives, according to this method it would consume 16 hours of capacity that day and be completed. This is inaccurate in many cases as the job may actually be setup and run on only 1 machine. It would use only 8 hours of work center capacity the first day and not be complete until late the next day.

Sometimes the stated work center capacity is meant to represent labor hours. For example, the three-machine work center above could be set to a capacity of 16 hours per day. This is meant to imply that only two operators are available for the work center. If important customer orders are queued at this work center and other work centers are idle, it may well be possible (and obviously desirable) to staff the third machine. Unfortunately, the work center method will never produce such a schedule. It forces you to allocate operators to the work centers before you know where they are needed.

These simplistic work center representations may be useful for rough cut capacity planning. However, they do not offer the level of detail needed to produce a realistic shop floor schedule designed to meet customer delivery requirements.

An adequate representation must offer the ability to identify individual resources that comprise capacity: machines, operators, or even tools. The scheduling process must assign a particular resource to the job for the required duration. By scheduling to the individual resource, the limitations described above can be avoided.

How Are Jobs Assigned To Capacity?

There are a number of commercially available APS systems that offer the ability to manage capacity as individual resources needed for a job. The difference between the approaches is found in how jobs are assigned to the resources. Here the distinction between the different methods is subtler. The major distinction is between interactive decision support approaches and batch based, "rule-reliant" approaches.

Interactive decision support taps the strengths of both human schedulers and the computer to ensure that the schedules produced strike the appropriate balance between efficiency and customer service. The rule-reliant approaches require the computer's "rules" to make all the tradeoffs necessary to produce a schedule that strikes the same balance. The difference between these approaches becomes most clear when the systems are actually put into practice.

Rule-Reliant Approach

The most common and the oldest approach in the rule-reliant camp is that of discrete event simulation. This technique has been applied to the problem of scheduling (if only academically) as far back as the late 1960's. Discrete event simulation is a technique that creates a simulated "world" with simulated orders entering simulated queues, being selected with simulated decisions to be processed by simulated resources, orchestrated in accordance with a simulated clock. Simulated time advances by processing events (e.g., an order being completed at a machine or a new order being released). At these events decisions are required. For example, of all the jobs available in the queue, which should be run next? This is where the rules come in.

The rule-reliant approach supplies a "rule" to stand in for management's judgement, essentially to replace the depth and breadth of human decision-making. At this point in the simulation, a predetermined decision rule is applied (e.g., "*pick the order with least slack time remaining*" or "*pick an order that uses the same setup*"). On the surface either example might seem like a potentially reasonable decision. Either is a <u>generally</u> good tendency, the first aimed at meeting deliveries and the second at improving efficiency.

The problem is that with the <u>specific</u> set of orders you are facing today, will such a simplistic rule make the appropriate tradeoffs to balance efficiency and delivery? If the least slack time rule is employed, setups will rarely be combined and efficiency will suffer. If the matching setup rule is employed, critical customer requirements may be ignored in favor of jobs with common setups that are not needed for some time.

The rule-reliant approach answers this dilemma by making the rule more sophisticated (e.g., "*pick the order with the least slack time unless an order with a common setup is available*"). If "common setup" refers to several criteria, for example material type vs. diameter vs. tooling, the rule must be made more sophisticated still to distinguish which types of common setups should be preferred over lower slack time.

Notice that even with the more sophisticated rule, a job with negative slack time might be present (i.e., a job already past due). This rule could continue to pick jobs based on setup, ignoring the past due job.

To avoid this disastrous possibility, more conditions must be added to the rule to decide how low slack time should be before taking precedence over the various types of common setups. In fact, each exception requires that more conditions be added to the rule making it more and more complicated.

This is the first disadvantage of the rule-reliant approach. To create schedules that take into account the realities of day-to-day manufacturing decisions, complex rules must be developed. As product mix, processes, equipment or policies change or are added, these rules must be created, revised and maintained. Typically this task, essentially a programming task, is beyond the scope of the scheduling department.

In addition, rules that consider only what jobs are in the current queue are often insufficient. Common sense tells us that upstream and downstream work centers must also be considered in the scheduling rule. A sophisticated rule to select jobs from the current queue will be ineffective if the upstream work centers are not feeding the right jobs. For example, if an upstream work center employs a least slack time rule, we may miss the opportunity to combine setups at this work center because the jobs have not reached this queue. Similarly, a setup minimization rule at this work center may cause a bottleneck at one downstream work center while another downstream work center starves due to differences in routings.

This is the second disadvantage of the rule-reliant approach. In most implementations, the rules at each work center operate effectively in isolation, unable to look upstream or downstream to see the effect of a decision. Some systems offer the ability to look upstream or downstream (only to detect the current state of other work centers, not to see the effect of a decision). Essentially, this provides the ability to heap more complexity into the rules in an attempt to coordinate the already complex rules operating locally at each work center.

The third disadvantage, and perhaps the most overlooked, of the rule-reliant approach, is the fact that the rules are evaluated within the isolated confines of the computer. No criteria can be included in the decisions unless it is entered into the computer. For example, imagine two orders with the same due date, one is a restocking order from a distributor and the other is a custom order destined for a critical project. While both have the same due date, being on time for the custom order is clearly more important. To the scheduler this will be obvious, as he will recognize the customers. To a rule inside the computer these orders are the same. For the computer to distinguish between them a condition must be added to the rules, <u>and</u> additional data must be supplied to distinguish between the two.

Furthermore any data that is wrong, either through error or lack of maintenance, will be treated as if it is accurate. Obviously the “garbage in, garbage out” axiom applies. However, since the computer is solely responsible for generating the schedule, there is no opportunity for the human scheduler to filter out inaccurate and obsolete data. Nor is there an opportunity for him to alter the decisions of the computer to reflect considerations for which the computer has wrong or incomplete data.

Ideally, there should be a way for the scheduler to override or modify the schedule. That is, to deal with the exceptions and the considerations that the system does not handle. Some rule-reliant systems provide a module to manipulate a graphic representation, usually a Gantt chart, of the schedule to make these modifications.

This points out the fourth disadvantage of the rule-reliant approach. Any manual overrides made to the graphic display of the schedule are not maintained when the schedule is regenerated. The schedule created is purely a function of the rules. The system has no way of knowing that the scheduler moved an order up on the Gantt chart. Therefore, a schedule regeneration will put the job right back where it was before the move.

Interactive Decision Support Approach

Interactive decision support approaches to APS have grown out of the graphical, interactive environments available in today’s powerful desktop computers. In fact, some systems have been developed in response to the difficulties of implementing batch-based, rule-reliant approaches. The interactive decision support approach is characterized by the ability to specify the schedule more directly, rather than indirectly through rules.

The schedule is often displayed in Gantt chart form (as well as other forms). It can be manipulated as part of the scheduling process rather than after scheduling has occurred. This method avoids the difficulties of specifying complex rules by allowing the scheduler to directly specify the desired order of work.

This allows the scheduler the flexibility to handle any situation, including those situations where no data is present in the system that might suggest the best course of action. For example, suppose a key machine at a downstream operation has just gone down - indefinitely. A human scheduler will know that there is little point in running jobs destined for that machine if other jobs are available.

Interactive decision support will allow the scheduler to rearrange the work as necessary to avoid the downstream problem. A rule-reliant system would need several extra layers of contingency rules to handle this situation (and similar contingency rules for a multitude of other unplanned events). It would also require that data be entered to make the system aware of the condition.

Critics argue that the interactive decision support approach may require too much manual manipulation of the schedule.

However, the approach does not preclude automatic schedule generation. Nor does it necessarily limit the types of automatic scheduling procedures that may be used. Complex automated procedures, for example bottleneck scheduling, may be developed. Most important, the results of the automated scheduling can be adjusted by the scheduler and desired portions of the schedule locked to preserve his desired sequences. This type of control can ultimately reduce the need to rearrange the schedule and promote the stability often required in the near term schedule.

Interactive decision support allows the scheduler to balance delivery needs against efficiency based on today's conditions and today's customer orders, not simply according to some predefined rule. If next week's backlog grows or shrinks, the best balance for the schedule may be different.

At some point, however, the scheduler may discover that the interactive procedures he uses to select, sort and/or group jobs for the schedule are consistently producing good results because conditions have stabilized. He can then automate those procedures, producing schedules as good or better than those that result from a sophisticated scheduling rule. The difference is that this procedure is developed and understood by the scheduler and he is therefore capable of modifying it as business conditions change.

Of course, the safety valve of manual interaction is always available to reposition jobs in any schedule created by an automated procedure. Interactive decision support relies on features that identify conflict in the schedule rather than complex rules that attempt to resolve or avoid it. The conflicting jobs or problem jobs can be handled by exception with manual adjustments. Rule-reliant systems do not eliminate the need for schedule adjustments, rather they prevent them from being made and maintained.

Interactive decision support errs on the side of flexibility. The real life of a customer driven manufacturing company is rarely predictable enough to be run according to a predefined set of rules. The decision support approach allows those aspects of scheduling for which the decisions are routine to be automated and those that are not to be handled by people. The decision of how much should be automated versus how much is handled interactively rests with the user.

Conclusion

APS has suffered its share of doubt and criticism, though not because the need to schedule capacity has been in question. To improve customer service, a clear view of how capacity is booked is critical. The doubts have arisen mainly from failed implementations. Simplistic, work center capacity representations offer too little detail to create achievable shop floor schedules, and often shops have been unable to execute.

In response, more detailed models have been developed to provide more accurate representations of the capacity constraints. Initially, most of these systems were batch-based and rule-reliant.

The promise of the rule-reliant approach is that a series of theoretically good decision rules, applied in isolation, can together produce a theoretically "good" schedule. The danger is that when applied to a specific set of orders and factory conditions what may be generally good rules can produce schedules that are specifically bad for a critical customer order or critical machine.

Since no facility exists to interactively correct such problems, the schedules can only be improved by adding more complexity to the rules. The net result can be an implementation that drags on (with mounting programming costs) as the rules are revised and new data requirements are identified.

Recently, newer systems based on the concept of interactive decision support have become available. Some of these systems offer resource constraint features as detailed as those found in the rule-reliant approaches. The difference is found in the ability for the system and scheduler to work together to ensure that the schedule successfully balances customer service and efficiency. While some or the entire schedule may be created automatically, the scheduler always retains the ability to override.

As customer demands cause lead times to shrink even further, the prospect of "theoretically good rules" producing acceptable schedules dims considerably. It will become more important that people can specify critical decisions, often involving important customer commitments, where necessary.

People are equipped with far greater problem solving capability and have access to greater and more current information than any computer system. Interactive decisions, combined with automatic scheduling procedures, stand a much greater chance of producing schedules that are responsive to the needs of customers while respecting efficient manufacturing practices.

References

Federman, R., "Finite Scheduling - The Myth of the Computer", APICS 1991 Conference Proceedings, Nov 1991

Kanet, J., "The Leitstand - Real Decision Support for Production Scheduling and Control", Production & Inventory Management, Sep 1991

McKay, K., and C. Murgiano, "Decision Support Requirements for Planning & Scheduling within the Automated Factory", AUTOFACT'91, Nov 1991

About the Author

Andrew Gilman is currently a principal with Waterloo Manufacturing Software. He has over ten years experience helping clients with software for interactive decision support, simulation and scheduling. Mr. Gilman holds a B.Sc. degree in Operations Research from Cornell University.

More Information

This paper was published in APICS The Performance Advantage. It is being provided with compliments from Waterloo Manufacturing Software. For more information about Waterloo Manufacturing Software's advanced finite capacity planning and scheduling system, TACTIC, or Mr. Gilman's other papers, contact:

Waterloo Manufacturing Software
P.O. Box 81264
Wellesley, MA 02481-0002

Voice: 781-237-2678
Fax: 781-237-9999
E-mail: sales@waterloo-software.com
Web: www.waterloo-software.com