Simulation versus Optimization Based Scheduling

Scheduling problems are typically large and complex and they are classified mathematically in a group of problems referred to as NP-Hard (non-deterministic polynomial-time hard). In non-mathematical terms these are the hardest of the hard computational problems for which no practical optimal algorithms exist. As a result, all scheduling solutions make use of heuristics and hence none produce an optimal solution (regardless of what vendors might suggest). The best that can be hoped for with this class of problems is a "good" solution that is better and easier to obtain than trying to manually generate a solution using Excel or a planning board. In this white paper we will briefly compare the optimization and simulation based approaches to the scheduling problem.

Optimization-based Scheduling

Despite their name, optimization-based scheduling tools use heuristic approaches that attempt to generate a solution that satisfies a list of mathematical constraints, and also provides some reasonable performance on key measures such as the number of tardy jobs.   There are no guarantees of the quality of the schedule compared to other possible schedules or the computational time that might be required to get to a feasible or good schedule. These tools generally strive first to generate a feasible schedule, and then improve on that schedule by exploring variations of the feasible schedule to improve performance measures.   The quality of the final answer will be limited by how much time is allowed for the algorithm to search for better solutions.   Typically, a time limit is set (e.g. an hour) for the algorithmic search and the best solution found during that search time is used as the schedule.

The algorithm (sometimes referred to as a solver) searches based on a mathematical model of the production system that is configured during the installation of the software.   The mathematical model describes the resource constraints of the system, and in some cases may include details such as sequence dependent setup times for a resource. However, this mathematical model has a limited structure and cannot capture all constraints; e.g. complex material handling such as the movement of AGV's or multiple overlapping cranes. Even within the capabilities of the mathematical model it is often necessary to omit details of the facility (e.g. sequence dependent setup times) to keep the required computational time for the search algorithm within acceptable limits.   The problem with this is that the resulting schedules are often infeasible in the actual system due to the constraints that have been ignored by the scheduling tool. Because of the limitations of the mathematical model, optimization-based tools work best in simple straight-forward applications with few complex constraints.

One of the best known optimization-based scheduling tools is PP/DS from SAP.   This tool offers two alternative solvers for generating solutions to the mathematical model. The first solver is a ***constraint-based programming*** algorithm that is the best choice for searching for a feasible solution when a lot of complex constraints are present. However, this solver is less effective for finding good solutions.   The second solver is a ***genetic algorithm*** that may provide better solutions as long as not too many constraints exists.   The success of either of these solvers to provide a reasonable solution within the

bounds of a practical search time depends upon the complexity of the mathematical model and it's approximate fit to the actual system.

The solvers are often not capable of solving the problem in a practical time, and hence PP/DS also provides a set of simple heuristic algorithms to generate a feasible schedule.   Most of the PP/DS implementations actually use these simple heuristics and make no use of the two solvers that are available.

Simulation-based Scheduling

The simulation-based approach to scheduling replaces the set of mathematical constraints by a simulation model of the facility. Some simulation-based tools (e.g. Preactor) use a hard-coded simulation model of the facility that is configured using a database. These tools are limited to applications where the data-driven model is a reasonable fit for the facility. Other simulation-based tools (e.g. Simio) supports both a data-driven approach (for standard applications) as well as allow for custom creation of the underlying simulation model of the facility (for complex applications). This later approach provides complete flexibility in capturing the detailed process constraints in the system and greater transparency into the schedule generation process.

In a simulation-based tool the schedule is generated by simulating the flow of work through the model of the facility.   As the jobs move through the simulated facility they will queue up in front of machines, wait for material, tools, and operators to be available before proceeding.   The movement of work through the simulated facility model is recorded and displayed in the form of Gantt charts that show each job's flow through the system. The resource Gantt chart shows the loading of each job on each resource in the system, and the job Gantt chart shows the flow of each job across the resources that it utilizes during processing. The following shows a simple resource Gantt chart generated by the Simio simulation-based scheduler. This Gantt chart is interactive and allows the scheduler to drag and drop orders within and between machines on the Gantt chart. This allows the scheduler to make adjustments to the schedule that is generated by the simulation model.
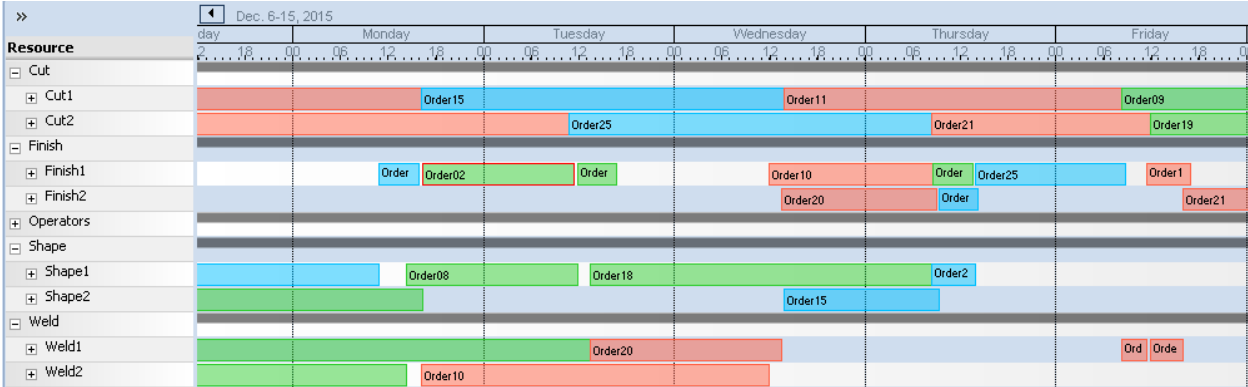

In addition to the interactive Gantt charts, Simio can also provide detailed reports and interactive dashboards that summarize the scheduled flow of work through the facility.   These can include standard reports/dashboards on the resources, materials, and constraints in the system, as well of

custom reports/dashboards for specific applications. The schedule results can be disseminated to multiple users on mobile devices using the cloud-based Simio Portal. The following shows a sample interactive dashboard depicting the machine states and work-to-list for the machines in the facility.
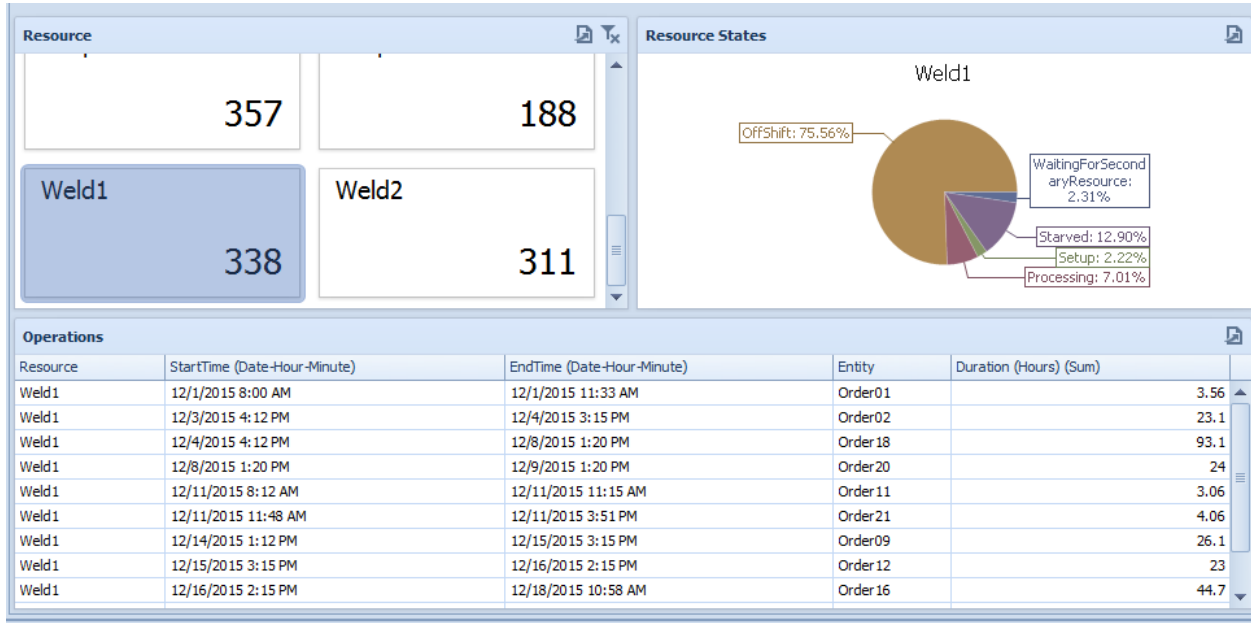
In Simio the flow of work through the system for generating the schedule can also be viewed in a hi-fidelity 3D animation (see below). This makes the schedule generation process a highly transparent process. The following depicts a snapshot of the animated schedule generation for the Simio simulation-based scheduler.
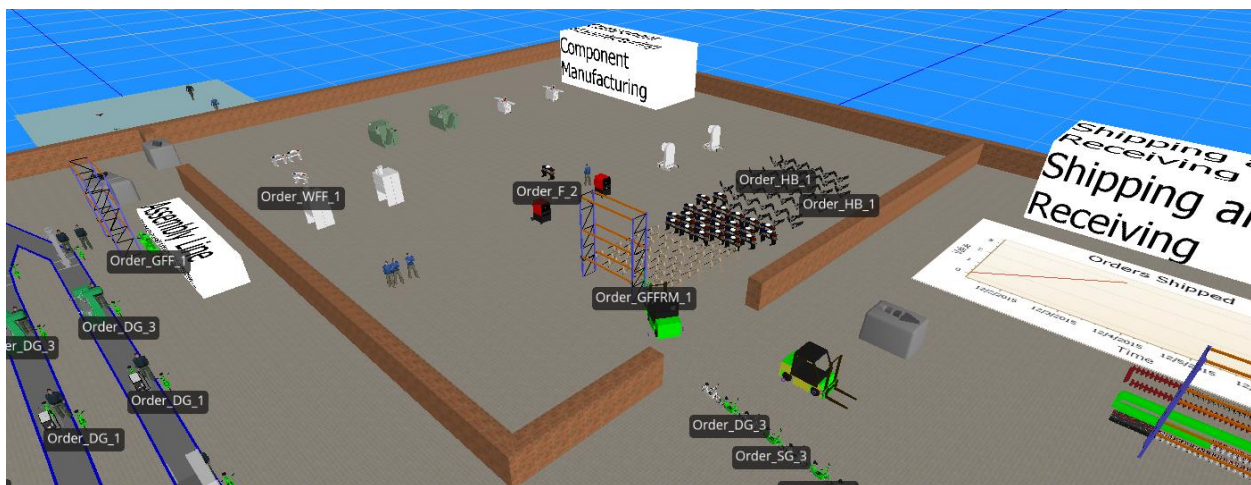
The optimization of the schedule occurs by applying heuristic scheduling rules in the decisions that get made during the execution of the model. For example, when a machine goes idle it may have a choice of which job to process next. Simulation-based scheduling tools typically provide a rich set of rules for making this selection. For example, in a process industry a rule may be specified that minimizes sequence dependent setup times/costs at a specific workstation. Other rules are available that focus on other key performance measures such as minimizing tardiness, maximizing throughput, or minimizing idle time on a bottleneck machine. It's also easy to create custom rules for application specific performance measures.

One key advantage of the simulation-based approach to scheduling is that when a disruptive event occurs (e.g. a machine breakdown) a new schedule can be generated in a few seconds or minutes (compared to hours for an optimization-based tool). The significantly faster speed of simulation-based scheduling tools makes them the superior choice in highly dynamic environments where the need to re-schedule occurs frequently.

Another key advantage of the simulation-based approach is the ability to capture the detailed constraints of the system in a high-fidelity facility model. This results in realistic and executable schedules that can be implemented "as-is" on the factory floor without adjustment by the shop floor operators. When a scheduling tool generates infeasible schedules that require shop floor changes, the operators may make "local" decisions that produce a non-optimal schedule.

A critical problem with the optimization-based scheduling approach is that it requires that all the data be fully known and deterministic. For example, all processing times must be fixed and there can be no unexpected events or delays. Unfortunately, this is the "happy path" through the system and the resulting schedule is by nature optimistic, and is typically very different from what occurs in the real facility. It is common that what starts of as a feasible schedule turns infeasible over time as variation and unplanned events degrade performance. It is normal to have large discrepancies between predicted schedules and actual performance. To protect against delays, the scheduler must buffer with some combination of extra time, inventory, or capacity; all adding cost to the system. Simio's simulation-based Risk-based Planning and Scheduling (RPS) extends traditional scheduling to fully account for the variation that is present in nearly any production system, and provides the necessary information to the scheduler to allow the upfront mitigation of risk and uncertainty.