MODAPTO [101091996]: Modular Manufacturing and Distributed Control via Interoperable Digital Twins

5.4.3 Limitations of Exact Methods

2025-08-11

While exact methods provide optimal solutions with mathematical guarantees, their practical application in production scheduling faces significant limitations. Most production scheduling problems belong to the class of NP-hard problems, meaning no known polynomial-time algorithm exists for finding optimal solutions. Scheduling problems with larger dimensions usually belong to the class of NP hard combinatorial problems.

Memory requirements also grow dramatically with problem size. For large-scale scheduling problems, the number of variables and constraints can become so large that even modern computing systems struggle to process them efficiently. This makes exact methods less suitable for solving large or complex production problems within reasonable time limits.

Another limitation is the trade-off between model accuracy and solvability. Adding real-world features like sequence-dependent setup times, machine-specific capabilities, and resource availability can greatly increase model complexity. This often leads to longer solution times or makes the problem too difficult to solve exactly.

Exact methods also assume stable and predictable input data. However, real production environments are dynamic, with frequent changes caused by delays, breakdowns, or urgent orders. Since exact methods require time to recompute new solutions, they are not well-suited for responding quickly to such disruptions.

Finally, implementing exact methods can be difficult in practice. Building and maintaining detailed mathematical models requires specialized knowledge, and integrating them into existing systems may require significant technical effort. Licensing costs for commercial solvers and the need for high-quality data further limit their use, especially for small or medium-sized operations.

Because of these limitations, exact methods are most often used for small problems, strategic planning, or benchmarking. In larger or rapidly changing environments, faster and more flexible heuristic or hybrid methods are typically preferred.

References

1. Garey, M. R., & Johnson, D. S. (1979). Computers and intractability: A guide to the theory of NP-completeness. W. H. Freeman & Co.
2. Lenstra, J. K., & Rinnooy Kan, A. H. G. (1981). Complexity of machine scheduling problems. Annals of Discrete Mathematics, 11, 343-362.
3. Ouelhadj, D., & Petrovic, S. (2009). A survey of dynamic scheduling in manufacturing systems. Journal of Scheduling, 12(4), 417-431.

4. Aytug, H., Lawley, M. A., McKay, K., Mohan, S., & Uzsoy, R. (2005). Executing production schedules in the face of uncertainties: A review and some future directions. European Journal of Operational Research, 161(1), 86-110.
5. Wolsey, L. A. (2020). *Integer programming*. John Wiley & Sons.