Cognitive Computing and Internet of Things, Vol. 124, 2024, 133–143
https://doi.org/10.54941/ahfe1004711

From Simple to Sophisticated: Investigating the Spectrum of Decision Support Complexity With AI Integration in Manufacturing

Sylwia Olbrych, Alexander Nasuta, Marco Kemmerling, Anas Abdelrazeq, and Robert H. Schmitt

Laboratory for Machine Tools and Production Engineering: Intelligence in Quality Sensing (WZL-IQS) of RWTH Aachen University, 52068 Aachen, Germany

ABSTRACT

In the evolving manufacturing landscape, the integration of Artificial Intelligence (AI) into Decision Support Systems (DSSs) has become crucial for enhancing decision-making. However, a visible challenge arises from the wide range of methodologies available, requiring a thoughtful choice of a suitable method for a given problem description. The absence of adequate resources for guiding developers in selecting an appropriate method is evident. In response to this gap, the presented work aims to improve the clarity and understanding of integrating existing methods, including AI, into DSSs. The clarity is achieved by introducing a structured grouping of DSSs based on the implemented methodology into four categories: rule-based, optimisation-based, simulation-based, and learning-based. Furthermore, this research illustrates decision-making with real-world examples by drawing insights from the literature. It underlines the user-centric importance in decision-making, emphasising that the effectiveness of the chosen DSS category depends on user interaction and comprehension. Looking ahead with the continuous evolution of AI, the ongoing incorporation of methodological advancements into DSSs is crucial for the continuous improvement of decision-making processes and alignment with the dynamic needs of users and the challenges present in modern manufacturing.

Keywords: Decision support system, DSS application, Manufacturing, User-centric, Artificial intelligence

INTRODUCTION

The research on Decision Support Systems (DSSs) is a sub-field of Information Systems (IS) research. Such a system is commonly a computer-based application or software that supports human decision-making by providing interactive tools, data management, reporting, analytics, modelling or planning. The concept of DSS in manufacturing was first introduced in the early 1980s with the development of a DSS prototype for aerospace manufacturing (Miner, Grant and Mayer, 1981). Since then, DSS has evolved to incorporate various technologies, such as Manufacturing Execution Systems (MES), Enterprise Resource Planning (ERP), Advanced Planning Systems (APS), Big

Data, and Business Intelligence (BI) (Felsberger, Oberegger and Reiner, 2016). These systems have been implemented for managing the planning of continuous manufacturing (Mallya, Banerjee and Bistline, 2001), development of real-time optimisation of manufacturing processes (Terblanche Swanepoel, 2004), and analysis of the production from a sustainable perspective (Zarte, Pechmann and Nunes, 2019).

In recent years, the growth in data generation and information processing and the ongoing development of new technologies and applications have prompted the integration of artificial intelligence (AI) methodologies into DSSs. The purpose of incorporating AI into decision support is to accelerate decision-making, enhancing both consistency and velocity. DSSs equipped with Machine Learning (ML) and Deep Learning (DL) are situated within the realm of intelligent DSSs. In such a system, the decision-making process is supported by various models or algorithms developed based on available data (Tariq and Rafi, 2012).

However, AI methodologies tend to be so-called "black-box" approaches, where the mechanisms behind the output are not always transparent and possible to clarify. Consequently, developers face challenges in explaining these processes to end-users, which slows the integration of advanced methodologies in real-world scenarios. Therefore, bringing a user into the development is crucial for building trust for the system they use.

A thorough examination of the literature about DSS classifications led to the identification of different DSS classifications. Yet, none contain the guidelines for the developers to choose the right methodology for generating recommendations. Therefore, a novel grouping of DSSs, based on the implemented methodology, is introduced to cover this gap. This work addresses the following questions:

1. How can DSSs be grouped based on the methodological strategies employed in generating recommendations?
2. What criteria should be considered when selecting an appropriate methodological approach for a specific problem within a DSS?
3. How does the proposed DSS classification system enhance user-centric development?

The structure of this work is as follows: Section 2 introduces related works, followed by DSS industrial applications in Section 3. Section 4 provides a description of the proposed classification, and Section 5 outlines the criteria for development. Additionally, the importance of user-centric development is discussed in Section 6. Finally, the document concludes with a summary.

RELATED WORKS

As outlined in the "Barcelona Declaration for the Proper Development and Usage of Artificial Intelligence in Europe", there are two distinct types of AI: knowledge-based AI and data-driven AI (Steels and Lopez de Mantaras, 2018). Knowledge-based AI aims to translate human knowledge into computational terms, beginning with individuals' self-reported concepts and problem-solving approaches. On the other hand, data-driven AI relies on vast datasets to identify patterns, correlations, and insights, allowing the system to make predictions or decisions without explicit programming of rules. This approach harnesses the power of ML algorithms to uncover hidden relationships within the data, enabling the system to adapt and improve its performance over time based on the information it processes (Steels and Lopez de Mantaras, 2018). The aforementioned AI categories were utilised to identify a potential AI allocation within existing DSS classifications.

Research on the taxonomy and classification of DSS has been growing over the past decades (Musbah, Omar and Ayodeji, 2019). Some classifications are based on the mode of assistance, such as file drawer systems, data analysis systems, analysis information systems, accounting models, representational models and optimisation models (Alter, 1980). In this classification, AI is evident in categories such as representational models, where ML generates predictions; optimisation models, where algorithms optimise suggestions; and suggestion models, supporting the final recommendation appointment. Another classification, introduced by (Holsapple and Whinston, 1996), is based on orientation to the primary focus and comprises six types of DSS: text-oriented, database-oriented, spreadsheet-oriented, solver-oriented, rule-oriented, and compound-oriented. AI methodologies can be identified in rule-oriented DSS, where they process rules and generate recommendations to support decision-makers. Power (2004) extended the classification of DSS by introducing five types defined based on the leading component: communications-driven, data-driven, document-driven, knowledge-driven, and model-driven. Among these DSS types, AI methodologies can be applied in document-driven DSS, where this technology can enhance intelligent web search engines. In the context of ML, the model-driven DSS is also well-suited for its integration.

In the classifications mentioned above, it is evident that various AI methodologies emerge across DSS categories. However, the absence of a comprehensive classification system directly encompassing the diverse landscape of existing AI methods is apparent. This gap highlights the dynamic nature of DSSs and the necessity for a novel classification approach.

The proposed DSS classification addresses this gap by providing clarity and structure while facilitating understanding and comparing methods. It is a valuable tool for developers and end-users, enabling them to make an informed selection and implementation of methodologies for generating recommendations.

DSS IN INDUSTRY

A literature review was conducted to find a valuable classification of DSSs. This section presents a summary of various real-world applications that were found during the review process (see Table 1). This table serves as a base resource providing insights into practical implementations showcasing various applied methodologies. The examination of DSS applications in industry has guided the development of the classification, whose implementation is included in Table 1.

DSS CLASSIFICATION BASED ON APPLIED METHOD

Table 1. Implementation of proposed DSS classification on selected applications.

DSS Class	Method	Application	Reference
rule-based	Computer-Aided Process Planning Graphplan Algorithm	process planning	(Marchetta and Forradellas, 2007)
	Analytic Hierarchy Process fuzzy logic	robot selection	(Kapoor and Tak, 2005)
	rule-based algorithm fuzzy logic	robot manipulation	(Son, 2016)
	MAUT fuzzy logic	cost estimation	(Zhao *et al.*, 2006)
	fuzzy TOPSIS	robot selection	(Chu and Lin, 2003)
simulation-based	discrete-event simulation	operational production and planning	(Heilala *et al.*, 2010)
	discrete-event simulation	industrial field service	(Hertz *et al.*, 2014)
optimisation-based	Genetic Algorithm	worker assignment	(Kotwal and Dhope, 2015)
learning-based	Fuzzy Wavelet Neural Network	supplier selection	(Guo, Mo and Sun, 2012)
	Random Forest	tool wear prediction	(Wu *et al.*, 2017)
rule-based and optimisation-based	Computer-Aided Process Planning Evolutionary Algorithm	process planning	(Leo Kumar, 2017)
optimisation-based and learning-based	Reinforcement Learning	production scheduling	(Waubert de Puiseau, Meyes and Meisen, 2022) (Samsonov *et al.*, 2021) (Tassel, Gebser and Schekotihin, 2021)
	Reinforcement Learning Graph Neural Network	production scheduling	(Zhang *et al.*, 2020)
	Genetic Algorithm Artificial Neural Network	feature selection	(Ghahramani *et al.*, 2020)

The novel DSS classification, responding to the primary research question of this study, is based on applied methodology for generating recommendations and comprises four distinct groups: rule-based, optimisation-based, simulation-based, and learning-based (see Figure 1). These groups vary in terms of the methods applied, thereby affecting the system's complexity and the interpretability of the output information. As the methodology advances, end-user trust in the methodology tends to decrease due to the increased difficulty of the method and often a lack of tools to visualise the processes behind it. For instance, rule-based DSS relies on fixed rules and offers an easily visualised logical path for the final recommendation, whereas optimisation and simulation-based systems require a deeper understanding of the applied algorithms. Although explanations are available for these methods, they can be challenging to convey to non-experts or decision-makers. The last group, mainly learning-based DSS, utilises methods that learn patterns based on available data. While some ML methods offer explanations, DL methods are often considered "black-box" approaches. Sometimes, trading off this interpretability to arrive at higher-quality recommendations may be required to meet the end-user requirements.

It is important to note that the term ML covers various methodologies, including Linear Regression, Random Forest, Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), among others. In the proposed classification, AI methodologies refer exclusively to ML and DL to distinguish them from optimisation and simulation methods.

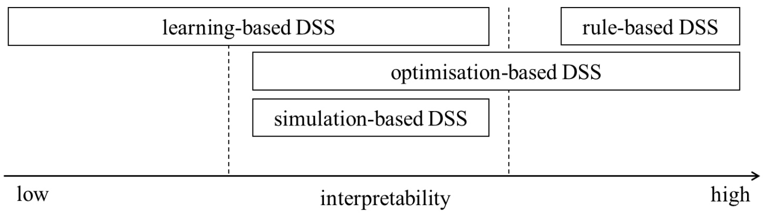

Figure 1: DSS classes based on applied methods for generating recommendations in DSS and their interpretability.

CRITERIA FOR DEVELOPMENT

The design and development of a DSS necessitates carefully considering multifaceted criteria, ranging from the nuanced requirements of end-users to the technical constraints of scalability and computational efficiency. Central to this process is the pivotal step of comprehending the underlying problem and selecting an appropriate methodology for its resolution (see Figure 2).

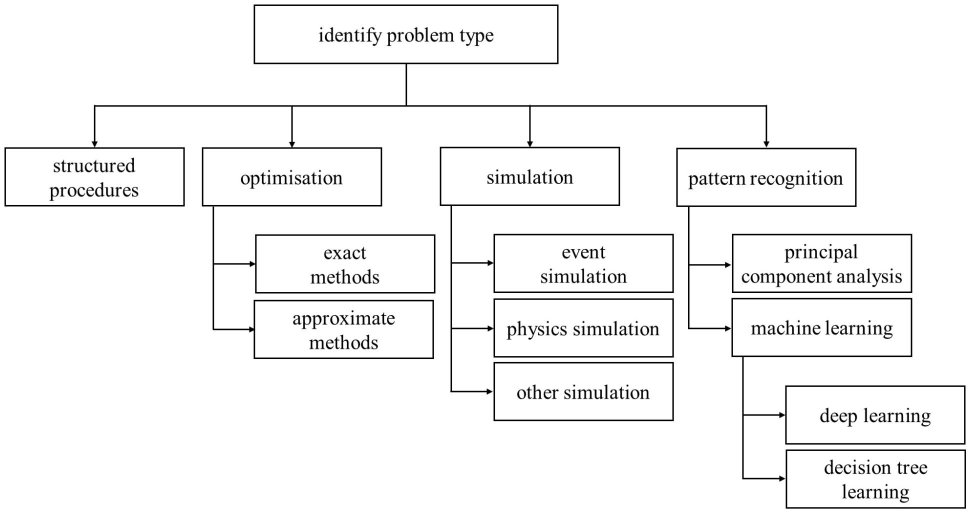

Figure 2: Flowchart representing a selection of methods based on a problem type.

Fundamentally, the problems a DSS addresses exhibit a spectrum of characteristics, each dictating the suitability of specific methodologies. Based on the literature research presented in Section 3, four primary categories are distinguished in the classification of DSS by problem types, such as structured procedures, optimisation, simulation, and pattern recognition. Each category not only encapsulates a distinct problem-solving approach but also reflects the nature and complexity of the underlying problem.

Structured procedures and optimisation are adept at handling problems characterised by well-defined parameters and a rather deterministic nature. They excel in scenarios where decision-making processes can be algorithmically formalised through deterministic rules or iterative optimisation techniques. On the other hand, simulation and pattern recognition are performed when confronted with problems imbued with stochastic elements and complex data structures.

One of the key guiding principles in this classification is the complexity of the underlying problem. Structured procedures typically cater to straightforward tasks with clear-cut decision pathways and minimal intricacies. Optimisation confronts a spectrum of complexities, ranging from polynomial-time solvable tasks to NP-hard challenges, necessitating the employment of sophisticated algorithms for effective resolution. Conversely, simulation finds its niche in addressing inherently complex systems, where the interactions of numerous variables defy simplistic analytical frameworks. Similarly, pattern recognition grapples with intricate problems, leveraging advanced techniques to distil meaningful patterns from vast and intricate datasets.

In subsequent paragraphs, a deeper delve into each problem type is undertaken, elucidating the methodologies employed, their strengths and limitations, and the contextual factors guiding their applicability. Through this comprehensive exploration, researchers and practitioners are equipped with a nuanced understanding of how DSS can be tailored to meet the diverse demands of problem-solving across various domains.

Structured Procedures

Structured procedures address problems that can be expressed in a rule-based structure. Comparatively, rule-based DSS is applied for undemanding problems with structured information and is considered the most straightforward for developers and end-users to implement and interact with. These systems often employ deterministic rules, statistical analysis of historical data, and experts' knowledge represented as fixed rules or decision trees. They also utilise fuzzy logic to obtain fuzzy output, storing practical knowledge of human operators about the process. For example, Computer-Aided Process Planning (CAPP) systems employ predefined rules and algorithms to generate process plans based on input criteria such as product specifications, manufacturing capabilities, and cost constraints.

Optimisation Problems

Optimisation involves seeking the optimal solution from feasible solutions, often by maximising or minimising an objective function while adhering to specific constraints. DSSs can utilise various optimisation techniques to address complex challenges, such as resource allocation and scheduling. Such problems can, for example, be expressed using sets of equations and variables, which can be solved using techniques like Constraint Programming (CP) or

Integer Programming (IP) (Hesamoddin and Hengameh, 2022). Although flexible, there are specialised optimisers and algorithms that are designed to solve specific issues more efficiently. For example, in supply chain management, where systems can be modelled as graphs, a Minimum Cost Flows (MCF) optimiser may be more effective than the IP optimiser in providing solutions.

CP, IP, and other specific optimisers are known as exact methods that provide optimal solutions based on real-world problem equations. However, the computational complexity of optimisation problems increases with problem size, especially regarding the number of variables that need assignment. When it comes to problems classified as NP-hard, they may scale up quickly, making it impossible to find solutions to large instances within reasonable time frames. In such cases, approximate algorithms are used, providing solutions without any formal optimality guarantees. These approximate methods mostly include (meta-) heuristics and Reinforcement Learning (RL) techniques.

SA and Monte Carlo Tree Search (MCTS) are commonly used meta-heuristics applicable to a wide range of problems. SA traverses the solution space of an optimisation problem inspired by the cooling process in metal casting (Talbi, 2009). MSTC iteratively explores promising solutions, balancing exploring new solutions with refining previously found ones (Mańdziuk, 2018; Kemmerling, Lütticke and Schmitt, 2024). It is worth noting that the term simulation in SA and MCTS refers to their iterative search processes and should not be confused with the class of "simulation" problems. A wide range of generic meta-heuristics can be applied to various optimisation problems. Talbi (2009) provides an extensive overview of these methods.

Simulation Problems

Simulation within the realm of DSS tackles complex problems characterised by dynamic states subject to change. Simulations can be broadly categorised into event simulation and physics simulation, each serving distinct purposes. Event simulations involve modelling discrete events using methods such as Moore- and Mealy-Automata, as well as Petri-Nets. These models help to simulate and analyse complex systems that involve discrete events. These models capture and analyse discrete changes in states over time. On the other hand, physics simulation predominantly describes the evolution of physical states over time, often utilising mathematical models such as differential equations and the Finite Element Method (FEM). Simulation techniques provide valuable insights into dynamic systems, enabling informed decision-making processes within DSS frameworks.

Pattern Recognition

Pattern recognition is a crucial component of DSS, offering insights into trends, anomalies, and correlations within data to facilitate informed decision-making. Techniques like Principal Component Analysis (PCA) are commonly employed to tackle pattern recognition tasks. While PCA is not classified as a ML technique, it plays a vital role in extracting underlying patterns from complex datasets by reducing dimensionality. Its ability to reveal hidden correlations and structures makes it invaluable in various domains. Additionally, decision tree learning, a branch of machine learning, offers a structured approach to decision-making by partitioning data based on feature attributes. Overall, integrating techniques like PCA alongside ML enhances the effectiveness of DSS by providing a comprehensive understanding of complex data.

Concerning the second question, which addresses factors for selecting the method within DSS, the criteria discussed in this work include the nature of the problem and the degree of interpretability. The guidelines for developers are summarised in the form of a decision matrix and presented in Table 2.

Table 2. Decision matrix for DSS selection.

		Problem Type			
		Structured Procedures	**Optimisation Problems**	**Simulation Problems**	**Pattern Recognition**
Interpretability	**High**	rule-based DSS	optimisation-based DSS		
	Medium		optimisation-based DSS	simulation-based DSS	learning-based DSS
	Low				learning-based DSS

USER-CENTRIC IMPORTANCE

User-centricity within DSS development involves designing systems prioritising user needs, preferences, and cognitive capabilities. In the context of AI integration into DSS, this approach becomes particularly crucial due to the complexities inherent in AI methodologies, which are often challenging for users (Wanner, 2021). User-centric design aims to bridge this gap by focusing on delivering insights in formats that are intuitive, understandable, and aligned with the users' mental models. It also ensures increased usability, acceptance, and effectiveness by tailoring the interface, interaction, and presentation of information to the users' preferences (Lal, Ballamudi and Thaduri, 2018).

In the context of the last research question in this study, the proposed DSS classification prevents unnecessary complexity in the system. It ensures that the choice of method used to generate recommendations is explainable, which is significant for developers and end-users. The guidelines summarised as a decision matrix offer developers a practical tool to tailor DSS development, ensuring that the final product effectively addresses user preferences (see Table 2).

CONCLUSION

This study contributes to the field by addressing the identified research questions. Firstly, a structured categorisation of DSSs into four distinct classes has been provided. The classification include rule-based, optimisation-based, simulation-based, and learning-based DSSs. Secondly, the essential criteria for selecting methodological approaches within DSSs have been identified. This contribution is particularly important because it recognises the challenge posed by the difficulty in explaining chosen methodologies to end-users. Lastly, the study aims to enhance user-centric development by promoting interpretability and trust. In summary, the proposed classification system aligns seamlessly with the dynamic needs of users and the evolving manufacturing landscape. By incorporating AI techniques into the classification, the DSSs remain adaptive, effective, and finely tuned to the user requirements in the modern manufacturing domain.

ACKNOWLEDGMENT

This work has been supported by the FAIRWork project (www.fairwork-project.eu) and has been funded within the European Commission's Horizon Europe Programme under contract number 101069499. This paper expresses the opinions of the authors and not necessarily those of the European Commission. The European Commission is not liable for any use that may be made of the information contained in this paper.

REFERENCES

Alter, S. (1980) Decision support systems: Current practice and continuing challenges. (Addison-Wesley series on decision support). Reading Mass.: Addison-Wesley Pub.

Chu, T.-C. and Lin, Y.-C. (2003) 'A Fuzzy TOPSIS Method for Robot Selection', The International Journal of Advanced Manufacturing Technology, 21(4), pp. 284–290. doi: 10.1007/s001700300033.

Felsberger, A., Oberegger, B. and Reiner, G. (2016) 'A Review of Decision Support Systems for Manufacturing', SAMI@ iKNOW, p. 8.

Ghahramani, M. et al. (2020) 'AI-based modeling and data-driven evaluation for smart manufacturing processes', IEEE/CAA Journal of Automatica Sinica, 7(4), pp. 1026–1037. doi: 10.1109/jas.2020.1003114.

Guo, R., Mo, R. and Sun, HB (2012) 'Study on Supplier Selection and Evaluation in Serviced-Oriented Manufacturing Network Based on NN', Advanced Materials Research, 479–481, pp. 644–647. doi: 10.4028/www.scientific.net/amr.479–481.644.

Heilala, J. et al. (2010) 'Developing simulation-based Decision Support Systems for customer-driven manufacturing operation planning', Proceedings of the 2010 Winter Simulation Conference, pp. 3363–3375. doi: 10.1109/WSC.2010.5679027.

Hertz, P. et al. (2014) 'A simulation-based decision support system for industrial field service network planning', International Conference on Advances in System Simulation.

Hesamoddin, T. and Hengameh, F. (2022) 'A Literature Review On Combining Heuristics and Exact Algorithms in Combinatorial Optimization', European Journal of Information Technologies and Computer Science, 2(2), pp. 6–12.

Holsapple, C. W. and Whinston, AB (1996) Decision support systems: A knowledge-based approach. Minneapolis/St. Paul: West Pub. Co.

Kapoor, V. and Tak, S. S. (2005) 'Fuzzy Application to the Analytic Hierarchy Process for Robot Selection', Fuzzy Optimization and Decision Making, 4(3), pp. 209–234. doi: 10.1007/s10700-005-1890-3.

Kemmerling, M., Lütticke, D. and Schmitt, R. H. (2024) 'Beyond games: a systematic review of neural Monte Carlo tree search applications', Appl Intell 54, 1020–1046. https://doi.org/10.1007/s10489-023-05240-w

Kotwal, J. G. and Dhope, T. S. (2015) 'Solving Task Allocation to the Worker Using Genetic Algorithm', International Journal of Computer Science and Information Technologies, 6(4), pp. 3736–3741.

Lal, K., Ballamudi, V. K. R. and Thaduri, U. R. (2018) 'Exploiting the Potential of Artificial Intelligence in Decision Support Systems', ABC Journal of Advanced Research, 7(2), pp. 131–138. doi: 10.18034/abcjar.v7i2.695

Leo Kumar, S. P. (2017) 'State of The Art-Intense Review on Artificial Intelligence Systems Application in Process Planning and Manufacturing', Engineering Applications of Artificial Intelligence, 65, pp. 294–329. doi: 10.1016/j.engappai.2017.08.005

Mallya, S., Banerjee, S. and Bistline, W. G. (2001) 'A Decision Support System for Production/ Distribution Planning in Continuous Manufacturing', Decision Sciences, 32(3), pp. 545–556. doi: 10.1111/j.1540–5915.2001.tb00971.x

Mańdziuk, J. (2018) 'MCTS/UCT in Solving Real-Life Problems', in Gaweda, A. E. et al. (eds.) Advances in Data Analysis with Computational Intelligence Methods. (Studies in computational intelligence). Cham: Springer International Publishing, pp. 277–292.

Marchetta, M. and Forradellas, R. (2007) 'Artificial Intelligence Planning for Generative Computer Aided Process Planning'.

Miner, R. J., Grant, M. E. and Mayer, R. J. (1981) 'Decision support for manufacturing', Online World Conference on Soft Computing in Industrial Applications.

Musbah, J. A., Omar, A. N. and Ayodeji, T. A. (2019) 'Decision support systems classification in industry', Periodicals of Engineering and Natural Sciences (PEN). Available at: https://api.semanticscholar.org/CorpusID: 201143299.

Power, DJ (2004) 'Specifying an Expanded Framework for Classifying and Describing Decision Support Systems', Communications of the Association for Information Systems, 13 (12). doi: 10.17705/1CAIS.01313.

Samsonov, V. et al. (2021) 'Manufacturing Control in Job Shop Environments with Reinforcement Learning', Proceedings of the 13th International Conference on Agents and Artificial Intelligence, pp. 589–597. doi: 10.5220/0010202405890597.

Son, C. (2016) 'Intelligent rule-based sequence planning algorithm with fuzzy optimisation for robot manipulation tasks in partially dynamic environments', Information Sciences, 342, pp. 209–221. doi: 10.1016/j.ins.2015.08.020.

Steels, L. and Lopez de Mantaras, R. (2018) 'The Barcelona declaration for the proper development and usage of artificial intelligence in Europe', AI Communications, 31(6), pp. 485–494. doi: 10.3233/AIC-180607

Talbi, E.-G. (2009) Metaheuristics: from design to implementation: John Wiley \& Sons.

Tariq, A. and Rafi, K. (2012) 'Intelligent decision support systems-A framework', Information and Knowledge Management, 2(6), pp. 12–20.

Tassel, P., Gebser, M. and Schekotihin, K. (2021) A Reinforcement Learning Environment For Job-Shop Scheduling.

Terblanche Swanepoel, K. (2004) 'Decision support system: real-time control of manufacturing processes', Journal of Manufacturing Technology Management, 15(1), pp. 68–75. doi: 10.1108/09576060410512338.
Wanner, J. (2021) 'Do you really want to know why? Effects of AI-based DSS Advice on Human Decisions', AMCIS 2021 Proceedings, 13.
Waubert de Puiseau, C., Meyes, R. and Meisen, T. (2022) 'On reliability of reinforcement learning based production scheduling systems: a comparative survey', Journal of Intelligent Manufacturing, 33(4), pp. 911–927. doi: 10.1007/s10845-022-01915-2.
Wu, D. et al. (2017) 'A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests', Journal of Manufacturing Science and Engineering, 139(7). doi: 10.1115/1.4036350.
Zarte, M., Pechmann, A. and Nunes, I. L. (2019) 'Decision support systems for sustainable manufacturing surrounding the product and production life cycle – A literature review', Journal of Cleaner Production, 219, pp. 336–349. doi: 10.1016/J. JCLEPRO.2019.02.092.
Zhang, C. et al. (2020) Learning to Dispatch for Job Shop Scheduling via Deep Reinforcement Learning.
Zhao, W. et al. (2006) 'Feature-based cost estimation for composite structures with Fuzzy Multi-Attribute Utility Theory', International Journal of Industrial and Systems Engineering, 1(1/2), p. 284. doi: 10.1504/IJISE.2006.009061.