Production Systems and Information Engineering
Volume 14 (2), pp. 1-14 1
doi: https://doi.org/10.32968/psaie.2026.2.1

Survey on Ant Colony System Algorithm in Flow Shop Scheduling Problem

ANITA AGÁRDI
University of Miskolc, Hungary
Institute of Information Technology
anita.agardi@uni-miskolc.hu

Abstract. This article provides a literature review for a production scheduling task, the Flow Shop Scheduling. The literature review focuses on a metaheuristic algorithm namely the Ant Colony System. This algorithm is a population-based heuristic. It is inspired by the behavior of ants, which evolved during the search for a path to food. Ants deposit pheromones along their path to provide the given path more attractive to other ants. This article describes the Ant Colony System Algorithm and the search results based on Google Scholar and Elsevier. Then, several publications related to the Ant Colony System production scheduling task are presented.

Keywords: Ant Colony System, Flow Shop Scheduling

1. Introduction

The article investigates a metaheuristic algorithm, the Ant Colony System. The paper also presents solutions to one of the most common tasks, the Flow Shop Scheduling Problem, using the ACS algorithm.

The Ant Colony System [1] is an ant-based metaheuristic algorithm. The algorithm is inspired by the path-finding behavior of ants to find food. It is an improved version of the Ant Colony Optimization (ACO) family. It is mainly used to solve combinatorial optimization problems. Ants leave pheromone trails along the path segments as they search for their path. Other ants are more likely to search for paths that are shorter and contain more pheromones. The algorithm consists of the following steps:

1. The first step is the initialization of the pheromone values.
2. Then the ants start searching for their path.
3. The ants create their path. Meanwhile, the algorithm performs local pheromone updates at each step.
4. The algorithm selects the best solution.
5. The next step is the global pheromone updating.
6. The algorithm also investigates the termination condition. The termination condition can be reaching a certain number of iterations or running time, or convergence.

Flow Shop Scheduling [2] is a classic production scheduling problem. Several jobs need to be processed on several machines in the same order. The goal is usually the makespan minimization. In a typical flow shop model, there are n jobs and m

machines. Each job runs in the same machine order: $M_1 \rightarrow M_2 \rightarrow \cdots \rightarrow M_m$. A machine can only process one job at a time. The job cannot be interrupted. The processing times of the jobs are known in advance.

2. Literature review based on publication databases

This section examines the number of articles, journal types, scientific classifications, etc. based on publication databases.

2.1. Literature review based on Google Scholar

The searches were conducted using Google Scholar, focusing on the Ant Colony System.

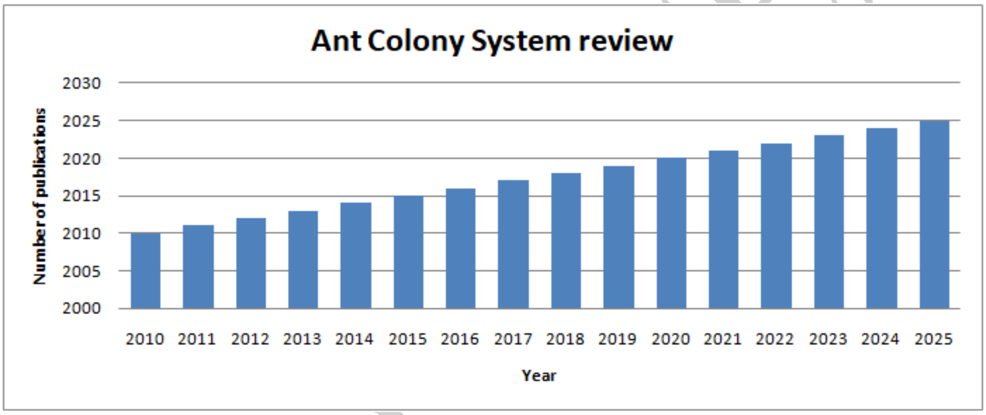

Figure 1. Google Scholar search: "Ant Colony System" and "review"

Figure 1 shows the results of the Google Scholar search, where the following keyword was applied: "Ant Colony System" and review". The number of publications was examined by year between 2010 and 2025. While only 414 publications were published in 2010, there were 775 in 2015, 1100 in 2020 and 1220 in 2025. It can therefore be seen that the number of articles published on the topic has been continuously increasing every year.

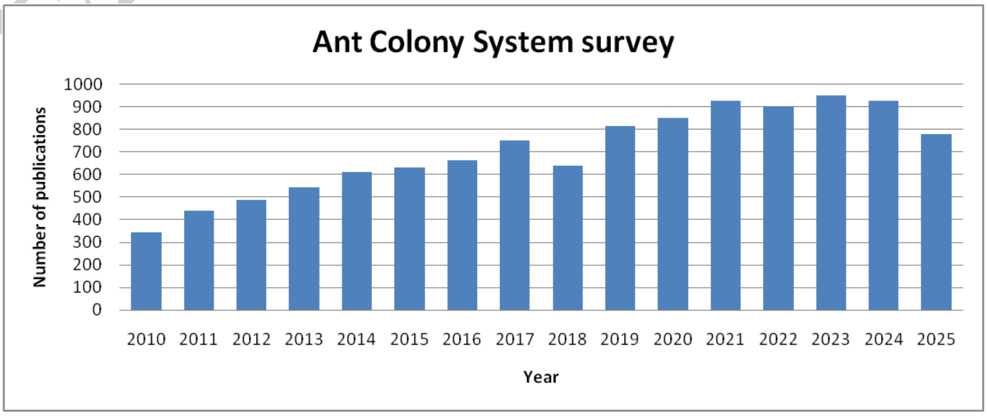

Figure 2. Google Scholar search: “Ant Colony System” and survey”

The keywords “Ant Colony System” and “survey” were also searched for between 2010 and 2025. The results also showed an increasing trend (except for 1-2 years). In 2010, 343 articles were published, in 2015 633, in 2020 852, while in 2025 779 articles were published.

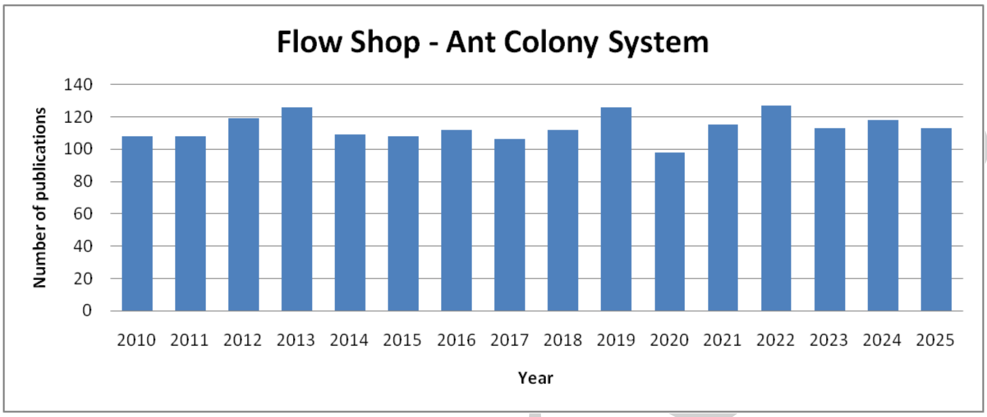

Figure 3. Google Scholar search: “Flow Shop” and “Ant Colony System”

Figure 3 shows the solution of the Flow Shop problem using the Ant Colony System algorithm. It presents how many articles were published on the topic each year between 2010 and 2025. We can see roughly similar numbers in the annual breakdown. In 2010, 108 publications were published, in 2015 also 108, in 2020 98 and in 2025 113 articles were written by the researchers.

2.2. Literature review based on Elsevier

This subsection examines the articles in the Elsevier database. The following parameters were taken into account: number of articles published per year, the article type, the publication title, and the subject areas.

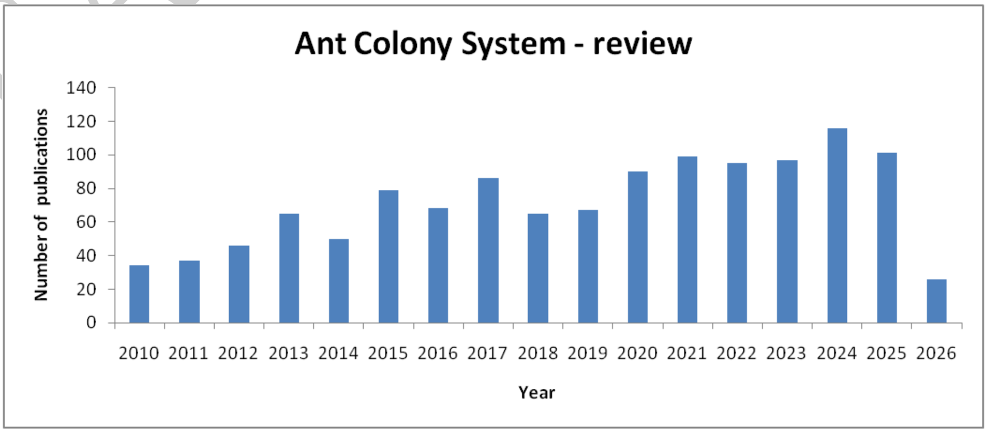

Figure 4. Elsevier search: “Ant Colony System” and “review”

Figure 4 shows the number of Ant Colony System review articles by year. While 34 articles were published in 2010, 79 articles were published in 2015, 90 in 2020, and 101 in 2025.

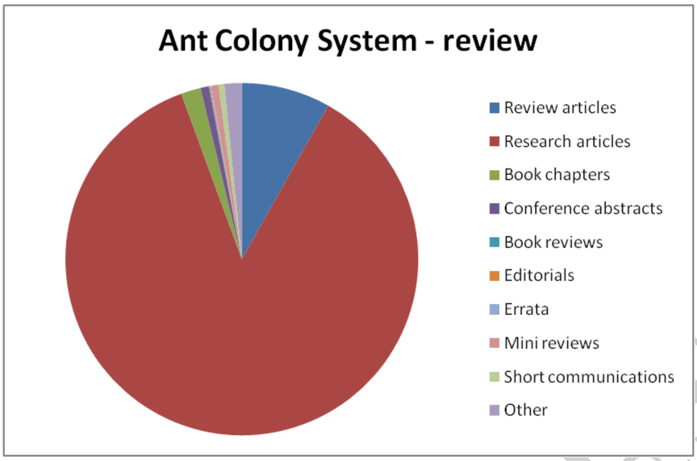

Figure 5. Elsevier search based on article type: "Ant Colony System" and "review"

Figure 5 shows the Elsevier search results in the article type category. There were 108 review articles, 1143 research articles (by far the most articles fell into this category), 24 book chapters, 10 conference abstracts, and 21 other articles.

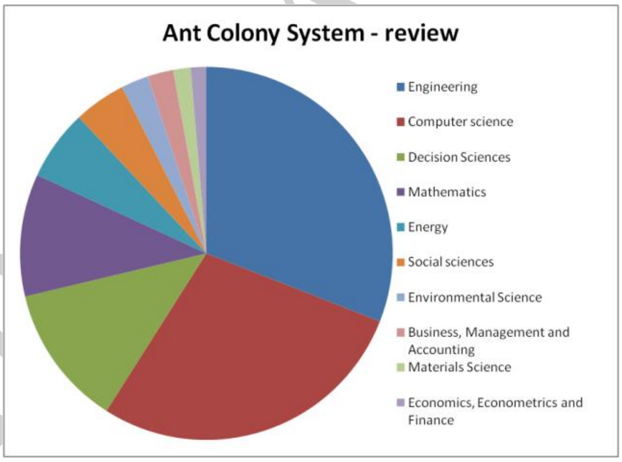

Figure 6. Elsevier search based on subject areas: "Ant Colony System" and "review"

Figure 6 shows the search results by subject area. It can be seen that the subject areas of most articles are engineering, computer science, decision sciences and mathematics.

Next is the keyword "Ant Colony System" and "survey", which yielded the following results.

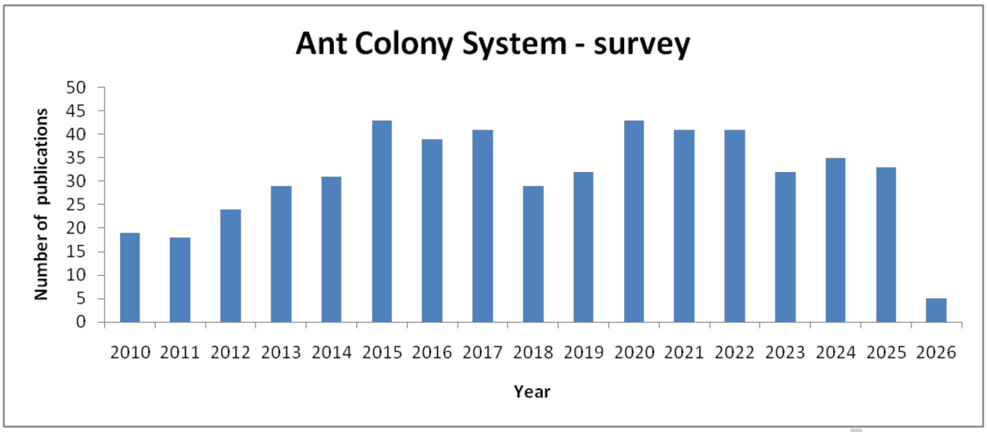

Figure 7. Elsevier search: "Ant Colony System" and "survey"

Figure 7 shows the number of publications published each year since 2010. In 2010, 19 articles were published, in 2015 43 publication, in 2020 also 43 articles, and in 2025 33 publications.

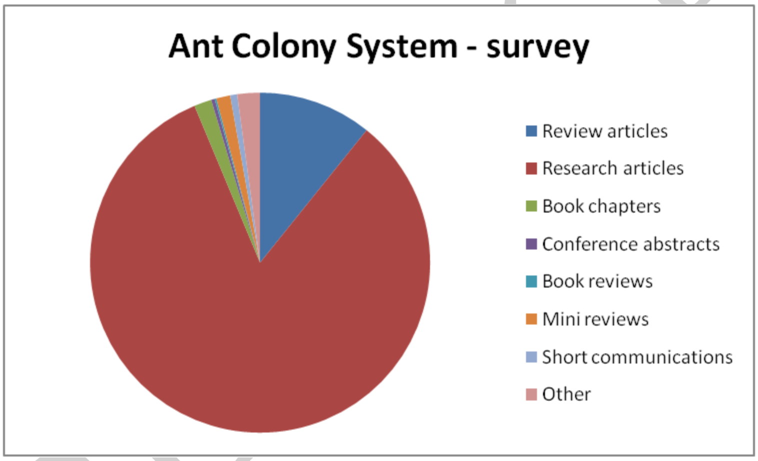

Figure 8. Elsevier search based on article type: "Ant Colony System" and "survey"

Figure 8 shows the results by article type. It can be seen that research articles have the highest value, with 500 articles.

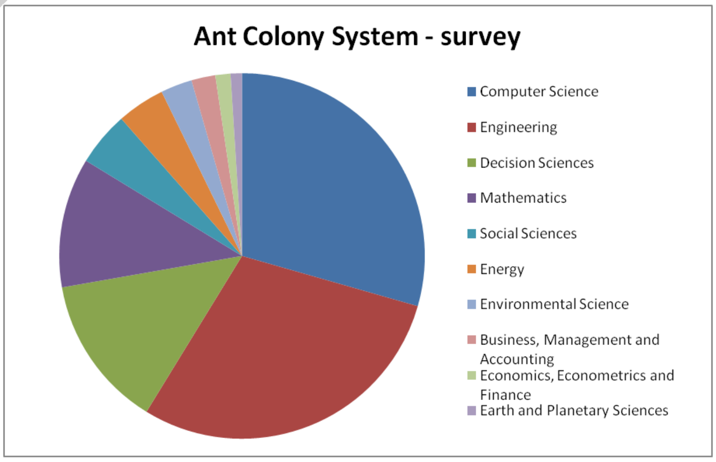

Figure 9. Elsevier search based on subject areas: “Ant Colony System” and “survey”

Figure 9 shows the results of the Ant Colony System survey by subject area. It can be seen that computer science and engineering subject have the most articles, with 264 and 263 articles respectively.

3. The presentation of important articles on the topic

This section presents the published articles on production scheduling problems solved with the Ant Colony System algorithm. First, the articles are presented one by one, and then the results are summarized in a summary table.

The authors of [3] publication investigated the m-machine Flow Shop Scheduling task. Their goal is to minimize the total completion time. The processing of the tasks and the setup of each machine requires the continuous presence of a single operator. And the setup of each machine requires the continuous presence of a single operator. Due to operator limitations, only one machine can operate at a time. Each job must be performed exactly once on each machine. A machine can only perform one job at a time. The setup is non-anticipatory, which means that the setup cannot start before the corresponding job can be started on the given machine. The task is called Flow Shop Scheduling Problem with batching and the assumption of item availability. The authors solved it Ant Colony Optimization. The test dataset was the following: 12 test task sets of different complexity, where each task set contains 10 tasks. The test dataset had the following complexities, where n is the number of jobs and m is the number of machines: Set 1: $\mathrm{m} = 2, \mathrm{n} = 5$, Set 2: $\mathrm{m} = 2, \mathrm{n} = 10$, Set 3: $\mathrm{m} = 3, \mathrm{n} = 10$, Set 4: $\mathrm{m} = 5, \mathrm{n} = 10$, Set 5: $\mathrm{m} = 3, \mathrm{n} = 20$.

The authors of [4] publication used the Ant Colony algorithm to minimize the flow time on the Permutation Flow Shop task. They compare three different Ant Colony Optimization algorithms and use the Taillard dataset for this purpose. They examine the efficiency for the following machine (m) and job (n) numbers: $\mathrm{m} = 5, 10, 20$ $\mathrm{n} = 20, 50, 100$.

The authors of publication [5] investigated the Job Shop Scheduling Problem (JSSP). In this problem, the authors consider the delivery time of jobs, which occurs when we switch from one machine to another while performing a job. The task is optimized using the Ant Colony algorithm.

In the paper of [6], a modified version of the Ant Colony algorithm is used to solve the Flow Shop Scheduling problem. The specific Flow Shop Scheduling problem is the Distributed Job Shop Scheduling Problem (DJSP). This problem deals with the assignment of jobs to factories. The factories are geographically distributed in the task. The goal is to minimize the global makespan for all factories. The following algorithms are used to solve the task: Ant Colony Optimization (ACO), Ant System (AS), Modified Ant Colony Optimization (MACO). The following were taken into account when defining the task: all jobs are independent of each other. Jobs are available at time 0. Once a job is assigned to a factory, it cannot be transferred to another factory. Operations cannot be interrupted, if a given machine has started it, it must also finish it, only then can another operation be started. A job can only be processed on one machine at a time, so a job cannot be processed on multiple machines at the same time. A task can only be performed once by a machine. The authors also ran the algorithm on the Taillard dataset. The MACO, ACS, and AS algorithms were compared on the dataset. The test runs were made from Tai01 to

Tai40.
In the article [7], the authors present the Flow Shop Scheduling task with assembly operations. An assembly flow shop consists of two types of stages: a manufacturing or machining stage and an assembly stage. The final products have a hierarchical assembly structure. The jobs must first be assigned to a machine and then assembled based on an assembly structure. The goal is to optimize the given objective functions. The article mentions several algorithms that have already been used to solve the Flow Shop task. Examples include the following: Artificial Bee Colony, Ant Colony Optimisation, Branch and Bound, Constraint Programming, Dynamic Programming, Distributed Production-Assembly Scheduling, Discrete Particle Swarm Optimisation, Earliest Due Date, Genetic Algorithm, Greedy Randomised Adaptive Search Procedure, General Variable Neighborhood Search, Hybrid Genetic Algorithm, Hybrid Tabu Search, Imperialist Competitive Algorithm, Integer Programming, Longest Processing Time, Multi-objective Iterated Local Search Algorithm, Mixed Integer Linear Model, Mixed Integer Nonlinear Programming, Multilayered Neural Network, Multi-objective Nawaz–Enscore–Ham Heuristic, Non-dominated Ranking Genetic Algorithm, Non-dominated Sorting Genetic Algorithm, Particle Swarm Optimization, Reduced Variable Neighborhood Search, Simulated Annealing, Self-adaptive Differential Evolution, Shortest Processing Time, Tabu Search, Variable Neighborhood Search. The article mentions the following objective functions: makespan minimization, total completion time, maximum lateness or total lateness, maximum tardiness or total tardiness, minimizing inventory holding costs, etc.
In the paper of [8], the authors solved a production scheduling with the Ant Colony System algorithm. The paper investigates the multi-phase HFSP. The authors propose a new Ant Colony System (ACS)-based heuristic to solve the problem. The results are compared with Genetic Algorithm (GA) and Tabu Search (TS) on a benchmark problem set in the paper. In the case of the Hybrid Flow Shop (HF), there is an (unrelated) parallel processing unit available in each stage. The availability of more than one processing unit reduces the production time. A processing unit can only perform one task at a time. The authors also investigated construction heuristics for solving the problem, such as shortest processing time heuristic, shortest final stage's processing time heuristic, Revised Campbell, Dudek, and Smith heuristic, Revised Raghu and Rajendran heuristic, Revised Palmer's heuristic. The construction heuristics were also compared with the Ant Colony System. The algorithms were tested with the following numbers of jobs and machines: $n = 5, 10, 20, 50, 100$ and $m = 2, 5, 8$. The article also compares the results with the Tabu Search heuristic.
In the paper of [9], the authors solved the Dynamic Job Shop Scheduling problem. They used the Ant Colony Optimization algorithm for it. The algorithm is tested in a manufacturing environment, where three different machine utilization levels are specified. In addition, the authors specified three types of processing time distributions during the test data set.
In paper of [10], the authors present a solution to a Multi-Phase Hybrid Flow Shop Scheduling Problem. An integrated Ant Colony Optimization Algorithm (IACS-HFS) is proposed for the problem.
In [11], the authors solve the Flow Shop Sequencing problem using the Ant Colony System algorithm. They compare the results with different algorithms such as Genetic Algorithm and Simulated Annealing. In terms of the problem sizes, the numbers of jobs and machines were as follows: $20/5, 20/10, 20/20, 50/5, 50/10, 50/20, 100/5, 100/10, 100/20, 200/10, 200/20, 500/20$.
In [12], the Permutation Flow Shop Scheduling Problem is solved using the Ant

Colony Optimization algorithm and the Cuckoo Search algorithm. The paper uses eight instances of the Car benchmark dataset (Car1, Car2, …, Car8) for testing. It compares the results of the hybridization of Cuckoo Search and Ant Colony.

The paper [13] also uses the Ant Colony Optimization algorithm to solve the Flow Shop Scheduling problem. A Flow shop scheduling problem is solved by multi-objective optimization, where the objective function is to minimize the makespan, the total flow time, and the total idle time of the machines. The authors use twenty-one test cases for testing. The number of cases varies between 20 and 75, and the number of machines varies between 5 and 20. Three test cases are assigned to each problem size. The results of ACO are compared with the NEH heuristic and the HC heuristic.

The paper [14] solves a production scheduling problem where uncertainties also play an important role. Maintenance also plays an important role in the paper. The costs resulting from unplanned machine failures often exceed the costs associated with predictive maintenance. Therefore, production and maintenance tasks must be scheduled together. The paper describes the application of ACO to the scheduling problem of production and maintenance tasks. The authors also present test results.

The paper [15] solves the Group Shop Scheduling Problem using the Ant Colony Optimization algorithm. This problem actually includes the Open Shop and Job Shop Scheduling problems. On the one hand, the authors propose a new neighborhood structure. On the other hand, they solve the problem using the ACO algorithm. The authors use different priority (or dispatching) rules: the total workload of the given task, the operation with the least/most remaining work within the given task, the operation with the longest/shortest processing time, the operation with the earliest start/finish time, a randomly selected operation. The authors also compare the results of the Ant Colony Optimization and Tabu Search algorithms in the test results.

In the paper of [16], a k-phase Hybrid Flow Shop (HFS) problem is solved. Due to the complexity of the problem, a hyper-heuristic is used to solve the problem. For each sub-problem (such as part assignment, part sequencing, and batch formation), this sub-problem is first solved by Genetic Programming (GP). Then, the Ant Colony Optimization (ACO) algorithm is used for optimization. The goal is to minimize the total weighted tardiness of the parts. The runs were performed with the following test data, where m is the number of machines and n is the number of jobs: (n=10, m=8), (n=15, m=8), (n=20, m=11), (n=25, m=11), (n=30, m=7), (n=35, m=7), (n=40, m=13), (n=45, m=13), (n=50, m=15), (n=55, m=15), (n=60, m=16), (n=65, m=16), (n=70, m=20), (n=75, m=20), (n=80, m=19), (n=85, m=19), (n=90, m=24), (n=95, m=24), (n=100, m=30), (n=105, m=30), (n=110, m=26), (n=115, m=26), (n=120, m=30), (n=125, m=30), (n=130, m=34), (n=135, m=34), (n=140, m=35), (n=145, m=35), (n=150, m=35), (n=155, m=35), (n=160, m=39), (n=165, m=39), (n=170, m=40), (n=175, m=40), (n=180, m=39), (n=185, m=39).

The paper [17] solves the No-Wait Flow Shop Scheduling Problem. The authors use a Hybrid Ant Colony and Whale Optimization Algorithm. The authors describe in detail the types of Flow Shop Scheduling problems, such as single machine problems, parallel machines, mixed work flow shop, no wait flow shop, flexible construction system, parallel machines problems, operation sequencing. Tests were run for the following complexity problems: (n=6, m=2), (n=6, m=3), (n=8, m=2), (n=8, m=3), (n=10, m=2), (n=10, m=3), (n=10, m=4) where n is the number of jobs, m is the number of machines. The following problem was solved: jobs are entered into the system according to their access time. It is not certain that they can be processed at the starting time (time 0). During the sequence of jobs, their preparation time must be taken into account, so it may be necessary to wait for the job to start on a given machine. The setup time of each job depends on the sequence of the given machines. A machine can only process one job at a time, and a job can only be done by one machine at a time. Machines are unavailable during certain periods. Processing of a job cannot begin until the processing of the previous job has finished.

In the publication of [18], the authors used the Ant Colony Optimization algorithm to solve the Hybrid Flow Shop Scheduling Problem. The goal of the problem is to minimize the maximum completion time (makespan) and the number of fixtures.

In [19], the authors solve the Flexible Job-Shop Scheduling with Routing Flexibility and Separable Setup Times problem. The authors solve the problem using the Ant Colony Optimization algorithm. Routing flexibility means that for a given process plan, the schedule can include alternative routes. The setup time is for the given job and two machines (which perform the job in sequence). The following algorithms are used for comparison in the test results: CDMT-AS, KTS-AS+, BS-MMAS*, RD-GA, GT-BDDR, RD-ACS (the algorithm proposed by the authors). The runs were performed on the following test datasets of complexity: (n=10, m=5), (n=15, m=5), (n=20, m=5), (n=10, m=10), (n=15, m=10), (n=20, m=10), (n=30, m=10).

In the paper of [20], the authors solve the Hybrid Flow Shop Scheduling task. The task contains lot sizing and calendar constraints. Different types of machines only operate during the working hours specified in the calendar. The task is defined in a printed circuit board assembly environment. Additional factors are also applied during the task, such as unrelated parallel machines and the possibility of skipping phases (stage skipping). The authors' approach breaks down the basic task into subtasks, such as job sequencing and lot scheduling with lot sizing.

In the paper of [21], the Quantum-inspired Ant Colony Optimisation Algorithm is used to optimise the Flow Shop problem. Two-stage permutation and batch processing machines are also used in the task. In the QIACO algorithm, the ants are divided into two groups: one group selects the largest job and the other the smallest (according to the size of the job).

In the publication of [22], the authors also apply the Hybrid Ant Colony Algorithm to a production scheduling task. This task is the No-wait Flow Shop task. In the task, each operation of a job follows the next one without interruption, immediately after the completion of the previous operation. Here too, the goal is to minimise the makespan. The algorithms were tested on the Car and Rec data sets. The algorithms were run for several complexities, such as: (n=11, m=5), (n=13, m=4), (n=12, m=5), (n=14, m=4), (n=10, m=4), (n=8, m=9), (n=7, m=7), (n=8, m=8), (n=20, m=5), (n=20, m=5), (n=20, m=5), (n=20, m=10), (n=20, m=10), (n=20, m=10), (n=20, m=15), (n=20, m=15), (n=20, m=15), (n=30, m=10), (n=30, m=10), (n=30, m=10), (n=30, m=15), (n=30, m=15), (n=30, m=15). The results of the following algorithms are also included in the table which indicates the test results: RAJ, VNS, GASA, TS, TS+M, TS+MP, HPSO, HGA, TS/PSO, ACO–SA, ACO, ACO-VNS, F&V.

In the paper of [23], a modified algorithm is presented, called the Local Search Ant Colony Optimization (LSACO) algorithm. The authors solve the Flow Shop Scheduling problem. This algorithm is based on the Alternative Ant Colony Optimization (AACO) algorithm. The algorithms were run on a total of 14 test data sets. The results of the following algorithms were compared: Random-order, Pheromone-related, Hybrid PSO. Here, the authors also chose the Taillard data set to analyze the efficiency of the algorithms.

In the paper of [24], the authors solve the Hybrid Flow Shop Scheduling problem. They use the Ant Colony Optimization algorithm for this. The authors presented the efficiency of the algorithm for several problem sizes, such as 10x5 10x10 15x5 15x10.

Reference	Authors	Publication year	Problems	Solving methods	Short summary
[3]	Li, X., Baki, M. F., & Aneja, Y. P.	2011	Flow Shop Scheduling Problem with batching and item availability	Ant Colony Optimization (ACO)	m-machine FSS with total completion time minimization, operator restriction (one machine can operate at a time). Non-anticipatory setup, batching assumption. 12 small-sized test tasks.
[4]	Rajendran, C., & Ziegler, H.	2005	Permutation Flow Shop Scheduling Problem	Ant Colony Optimization (3 version)	Minimizing flow time. Comparison of different ACO variants on Taillard benchmark (m=5–20, n=20–100).
[5]	Saidi-Mehrabad, M., Dehnavi-Arani, S., Evazabadian, F., & Mahmoodian, V.	2015	Job Shop Scheduling Problem with Delivery Times	Ant Colony Optimization	Extension of JSSP with taking into account transport times between machines.
[6]	Chaouch, I., Driss, O. B., & Ghedira, K.	2017	Distributed Flow Shop Scheduling Problem (DJSP)	ACO, AS, MACO, ACS	Geographically distributed factories, global makespan minimization. Algorithm comparisons on Taillard (Tai01–Tai40) benchmark.
[7]	Komaki, G. M., Sheikh, S., & Malakooti, B.	2019	Assembly Flow Shop Scheduling	Several methods (ACO, GA, PSO, TS, etc.)	Gyártási + összeszerelési szakaszokkal rendelkező flow shop. Irodalmi áttekintés számos algoritmusról és célfüggvényről.
[8]	Ying, K. C., & Lin, S. W.	2006	Hybrid Flow Shop Scheduling Problem (HFSP)	Ant Colony System (ACS)	Solving multi-phase HFSP with a new ACS-based heuristic. Compared with GA and TS.
[9]	Zhou, R., Nee, A. Y. C., & Lee, H. P.	2009	Dynamic Job Shop Scheduling Problem	Ant Colony Optimization	Dynamic environment, examination of different machine utilization levels and processing time distributions.
[10]	Khalouli, S., Ghedjati, F., & Hamzaoui, A.	2011	Multi-stage Hybrid Flow Shop Scheduling	Integrated ACO (IACS-HFS)	Solving a multiphase HFS problem using an integrated Ant Colony method.
[11]	Ying, K. C., & Liao, C. J.	2004	Flow Shop Sequencing Problem	Ant Colony System (ACS)	Comparison of ACS with GA and SA on different problem sizes (n=20–500, m=5–20).
[12]	Zhang, Y., Yu, Y., Zhang, S., Luo, Y., & Zhang, L.	2019	Permutation Flow Shop Scheduling Problem	ACO + Cuckoo Search (hibrid)	Hybrid ACO–CS approach on Car benchmark dataset (Car1–Car8).

[13]	Yagmahan, B., & Yenisey, M. M.	2008	Multi-objective Flow Shop Scheduling	Ant Colony Optimization	Multi-objective optimization: minimizing makespan, total flow time and idle time. Compared with NEH and HC heuristics.
[14]	Paprocka, I., Krenczyk, D., & Burduk, A.	2020	Production Scheduling with maintenance	Ant Colony Optimization	Joint scheduling of production and maintenance tasks under uncertainty.
[15]	Blum, C., & Sampels, M.	2004	Group Shop Scheduling Problem	Ant Colony Optimization	Generalization of Open Shop and Job Shop. New neighborhood structure, ACO vs Tabu Search comparison.
[16]	Chen, L., Zheng, H., Zheng, D., & Li, D.	2015	k-phase Hybrid Flow Shop Scheduling	GP + ACO (hyper-heuristics)	Part assignment, sequencing and batching with GP followed by ACO algorithm.
[17]	Rostamzadeh, R.	2024	No-wait Flow Shop Scheduling Problem	Hybrid ACO + Whale Optimization	No-wait conditions, machine unavailability, setup times taken into account. Small-medium test sizes.
[18]	Xu, L., Guo, J., & Zhu, L.	2023	Hybrid Flow Shop Scheduling	Ant Colony Optimization	Minimizing the number of makespan and fixtures.
[19]	Rossi, A., & Dini, G.	2007	Flexible Job Shop Scheduling with route flexibility	Ant Colony Optimization (RD-ACS)	Handling alternative routes and separable setup times, compared with several algorithms.
[20]	Qin, W., Zhuang, Z., Liu, Y., & Tang, O.	2019	Hybrid Flow Shop Scheduling with lot sizing	Decomposition approach + ACO	PCB assembly environment, calendar constraints, parallel machines and phase skipping.
[21]	Chen, Z., Zheng, X., Zhou, S., Liu, C., & Chen, H.	2020	Flow Shop Scheduling Problem	Quantum-inspired ACO (QIACO)	Two-stage permutation and batch processing machines, two types of ant strategies.
[22]	Riahi, V., & Kazemi, M.	2018	No-wait Flow Shop Scheduling Problem	Hybrid Ant Colony Algorithm	Makespan minimization, Car and Rec benchmarks
[23]	Chen, R. M., Lo, S. T., Wu, C. L., & Lin, T. H.	2008	Flow Shop Scheduling Problem	Local Search ACO (LSACO)	AACO-based method with local search, Taillard benchmark, compared with several algorithms.
[24]	Alaykýran, K., Engin, O., & Döyen, A.	2007	Hybrid Flow Shop Scheduling	Ant Colony Optimization	Solving HFS problems on different problem sizes (e.g. 10×5, 15×10).

4. Conclusion and future research direction

In this article, a literature review of one of the most common and well-known production scheduling tasks - the Flow Shop Scheduling - was presented, where the solution with the Ant Colony System algorithm was the focus. The article first presents the Ant Colony System algorithm and the classic Flow Shop Scheduling tasks. Then, the search results are based on two large databases, which are the Google Scholar and the Elsevier. After that, the most important publications published on the topic are presented in outline. The article also presents a tabular summary of the published publications, where the specific Flow Shop Scheduling tasks, the solved algorithms and the summary are also included in the table.

It can be stated that Ant Colony Optimization (ACO) and its various advanced versions can be effectively applied to complex manufacturing scheduling problems. Based on the presented articles, it can be seen that the application area of ACO extends from simpler flow shop sequencing and permutation flow shop problems to hybrid flow shop, job shop, flexible job shop, no-wait, multi-objective, distributed, as well as maintenance and uncertainty-driven scheduling problems.

In previous works, ACO mainly focused on minimizing a single optimization objective, such as makespan or total lead time. However, as time went on, more and more emphasis was placed on multi-objective optimization and modeling complex manufacturing systems. Such as delivery times, machine unavailability, setup times, parallel machines, batch processing, lot sizing, maintenance tasks, phase skipping, and dynamic environments.

However, based on the results, the classic ACO does not perform the search space exploration efficiently in all cases. Therefore, many studies have used hybrid approaches, combining ACO with genetic algorithms, tabu search, local search, particle swarm optimization, cuckoo search, whale optimization, and other metaheuristic and hyper-heuristic methods. The goals of hybridization are: accelerate convergence, avoid local optima, and improve the quality and robustness of solutions.

Future research directions can be a literature review with other metaheuristic algorithms, or to comparision the solution of a specific Flow Shop task with metaheuristic algorithms and benchmark data.

Acknowledgement. „SUPPORTED BY THE UNIVERSITY RESEARCH SCHOLARSHIP PROGRAM OF THE MINISTRY FOR CULTURE AND INNOVATION FROM THE SOURCE OF THE NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION FUND."

References

[1] Dorigo, M., & Gambardella, L. M. (2002). Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on evolutionary computation*, *1*(1), 53-66. https://doi.org/10.1109/4235.585892

[2] Ruiz, R., & Vázquez-Rodríguez, J. A. (2010). The hybrid flow shop scheduling problem. *European journal of operational research*, *205*(1), 1-18. https://doi.org/10.1016/j.ejor.2009.09.024

[3] Li, X., Baki, M. F., & Aneja, Y. P. (2011). Flow shop scheduling to minimize the total completion time with a permanently present operator: Models and ant colony optimization metaheuristic. *Computers & Operations Research*, *38*(1), 152-164. https://doi.org/10.1016/j.cor.2010.04.009

[4] Rajendran, C., & Ziegler, H. (2005). Two ant-colony algorithms for minimizing total flowtime in permutation flowshops. *Computers & Industrial Engineering*, *48*(4), 789-797. https://doi.org/10.1016/j.cie.2004.12.009

[5] Saidi-Mehrabad, M., Dehnavi-Arani, S., Evazabadian, F., & Mahmoodian, V. (2015). An Ant Colony Algorithm (ACA) for solving the new integrated model of job shop scheduling and conflict-free routing of AGVs. *Computers & Industrial Engineering*, *86*, 2-13. https://doi.org/10.1016/j.cie.2015.01.003

[6] Chaouch, I., Driss, O. B., & Ghedira, K. (2017). A modified ant colony optimization algorithm for the distributed job shop scheduling problem. *Procedia Computer Science*, *112*, 296-305. https://doi.org/10.1016/j.procs.2017.08.267

[7] Komaki, G. M., Sheikh, S., & Malakooti, B. (2019). Flow shop scheduling problems with assembly operations: a review and new trends. *International Journal of Production Research*, *57*(10), 2926-2955. https://doi.org/10.1080/00207543.2018.1550269

[8] Ying, K. C., & Lin, S. W. (2006). Multiprocessor task scheduling in multistage hybrid flow-shops: an ant colony system approach. *International Journal of Production Research*, *44*(16), 3161-3177. https://doi.org/10.1080/00207540500536939

[9] Zhou, R., Nee, A. Y. C., & Lee, H. P. (2009). Performance of an ant colony optimisation algorithm in dynamic job shop scheduling problems. *International Journal of Production Research*, *47*(11), 2903-2920. https://doi.org/10.1080/00207540701644219

[10] Khalouli, S., Ghedjati, F., & Hamzaoui, A. (2011). An ant colony system algorithm for the hybrid flow-shop scheduling problem. *International Journal of Applied Metaheuristic Computing (IJAMC)*, *2*(1), 29-43. https://doi.org/10.4018/jamc.2011010103

[11] Ying, K. C., & Liao, C. J. (2004). An ant colony system for permutation flow-shop sequencing. *Computers & Operations Research*, *31*(5), 791-801. https://doi.org/10.1016/S0305-0548(03)00038-8

[12] Zhang, Y., Yu, Y., Zhang, S., Luo, Y., & Zhang, L. (2019). Ant colony optimization for Cuckoo Search algorithm for permutation flow shop scheduling problem. *Systems Science & Control Engineering*, *7*(1), 20-27. https://doi.org/10.1080/21642583.2018.1555063

[13] Yagmahan, B., & Yenisey, M. M. (2008). Ant colony optimization for multi-objective flow shop scheduling problem. *Computers & Industrial Engineering*, *54*(3), 411-420. https://doi.org/10.1016/j.cie.2007.08.003

[14] Paprocka, I., Krenczyk, D., & Burduk, A. (2020). The method of production scheduling with uncertainties using the ants colony optimisation. *Applied Sciences*, *11*(1), 171. https://doi.org/10.3390/app11010171

[15] Blum, C., & Sampels, M. (2004). An ant colony optimization algorithm for shop scheduling problems. *Journal of Mathematical Modelling and Algorithms*, *3*(3), 285-308. https://doi.org/10.1023/B:JMMA.0000038614.39977.6f

[16] Chen, L., Zheng, H., Zheng, D., & Li, D. (2015, May). An ant colony optimization-based hyper-heuristic with genetic programming approach for a hybrid flow shop scheduling problem. In *2015 IEEE Congress on Evolutionary Computation*

(CEC) (pp. 814-821). IEEE. https://doi.org/10.1109/CEC.2015.7256975

[17] Rostamzadeh, R. (2024). An attempt to resolve no-wait flow shop scheduling problems using hybrid ant colony and whale optimization algorithms. https://doi.org/10.15240/tul/001/2024-2-007

[18] Xu, L., Guo, J., & Zhu, L. (2023). Ant Colony Optimization Algorithm for the Hybrid Flow Shop Scheduling Problem with Integrated Consideration of Fixture Resources. *Engineering Proceedings*, *45*(1), 37. https://doi.org/10.3390/engproc2023045037

[19] Rossi, A., & Dini, G. (2007). Flexible job-shop scheduling with routing flexibility and separable setup times using ant colony optimisation method. *Robotics and Computer-Integrated Manufacturing*, *23*(5), 503-516. https://doi.org/10.1016/j.rcim.2006.06.004

[20] Qin, W., Zhuang, Z., Liu, Y., & Tang, O. (2019). A two-stage ant colony algorithm for hybrid flow shop scheduling with lot sizing and calendar constraints in printed circuit board assembly. *Computers & Industrial Engineering*, *138*, 106115. https://doi.org/10.1016/j.cie.2019.106115

[21] Chen, Z., Zheng, X., Zhou, S., Liu, C., & Chen, H. (2020). Quantum-inspired ant colony optimisation algorithm for a two-stage permutation flow shop with batch processing machines. *International Journal of Production Research*, *58*(19), 5945-5963. https://doi.org/10.1080/00207543.2019.1661535

[22] Riahi, V., & Kazemi, M. (2018). A new hybrid ant colony algorithm for scheduling of no-wait flowshop. *Operational Research*, *18*(1), 55-74. https://doi.org/10.1007/s12351-016-0253-x

[23] Chen, R. M., Lo, S. T., Wu, C. L., & Lin, T. H. (2008, June). An effective ant colony optimization-based algorithm for flow shop scheduling. In *2008 IEEE Conference on Soft Computing in Industrial Applications* (pp. 101-106). IEEE. https://doi.org/10.1109/SMCIA.2008.5045943

[24] Alaykýran, K., Engin, O., & Döyen, A. (2007). Using ant colony optimization to solve hybrid flow shop scheduling problems. *The international journal of advanced manufacturing technology*, *35*(5), 541-550. https://doi.org/10.1007/s00170-007-1048-2