Volume XX, Issue (1) (2024) 142-151

International May Conference on Strategic Management

DOI: *10.5937/IMCSM24014J* *Original research*

INTEGRATED PROCESS PLANNING AND SCHEDULING OF PRODUCTION SYSTEMS BASED ON MOUNTAIN GAZELLE OPTIMIZER

Aleksandar Jokić *, Milica Petrović, Zoran Miljković

University of Belgrade, Faculty of Mechanical Engineering, Serbia

Abstract: The mass customization paradigm, in conjunction with high market demands, puts a significant burden on contemporary production systems to output a larger quantity of diversified parts. Consequently, production systems need to achieve even higher flexibility levels through physical and functional reconfigurability. One way of achieving these high levels of flexibility is by utilizing optimization of both scheduling and process planning. In this paper, the authors propose to solve an NP-hard integrated process planning and scheduling optimization problem with transportation constraints regarding one mobile robot. The proposed production environment includes four types of flexibilities (process, sequence, machine, and tool) that can be leveraged to optimize the entire manufacturing schedule. Three metaheuristic optimization algorithms are compared on the nine-problem benchmark based on the makespan metric. The proposed Mountain Gazelle Optimizer (MGO) is compared to the whale optimization algorithm and particle swarm optimization algorithm. The experimental results show that MGO achieves most best results, while it is highly comparable on the average best results.

Keywords: Integrated process planning and scheduling, optimization, mountain gazelle optimizer, metaheuristic algorithms, production systems.

1. INTRODUCTION

The manufacturing industry has undergone significant developments in recent years thanks to the Integrated Process Planning and Scheduling (IPPS) methodology (Phanden et al., 2019). This approach is designed to integrate process planning and scheduling activities, enabling businesses to maximize resource utilization and minimize costs while maintaining customer demands and delivery deadlines. IPPS leverages advanced optimization techniques such as mathematical programming or biologically inspired algorithms to generate efficient

* *Corresponding author: ajokic@mas.bg.ac.rs*

Aleksandar Jokić, ORCID: 0000-0002-7417-4244
Milica Petrović, ORCID: 0000-0002-4950-6518
Zoran Miljković, ORCID: 0000-0001-9706-6134

production plans and schedules that meet multiple objectives and constraints. Besides the increased efficiency of the production systems, another key advantage of IPPS is its ability to provide decision support through tools such as Gant charts visualizations of different planning scenarios. These features allow management to evaluate various alternative schedules and select the optimal solution. Methods IPPS problems are designed to tackle the complexity of these optimization problems, employing sophisticated algorithms such as Mountain Gazelle Optimizer (MGO) (Abdollahzadeh et al., 2022), Genetic Algorithms (GA), Whale Optimization Algorithm (WOA) (Mirjalili & Lewis, 2016), Particle Swarm Optimization (PSO) (Petrović et al., 2016), and Mixed Integer Linear Programming (MILP) (Caumond et al., 2009) to generate feasible scheduling plans.

The Mountain Gazelle Optimizer (MGO) is a novel population-based optimization algorithm that takes inspiration from the social structure, hierarchy, and behavior of mountain gazelles. MGO incorporates four critical behaviors of mountain gazelles, namely territorial solitary males, maternity herds, bachelor male herds, and migration in search of food, into its mathematical framework. Each behavior contributes to the algorithm's exploration and exploitation capabilities, making it excellent for balancing exploration and exploitation while achieving good performance across different optimization problems.

Genetic Algorithms (GA) are a class of optimization algorithms that are inspired by the process of natural selection and genetics. They are well-suited for solving complex problems, as they can efficiently search large solution spaces and handle multiple objectives and constraints. The GA involves generating a population of potential solutions (chromosomes) and then iteratively evolving them through selection, crossover, and mutation operations to find optimal or near-optimal solutions. This iterative process mimics the evolution of species in nature, where the fittest individuals are selected for reproduction. The process continues until a satisfactory solution is obtained.

The Whale Optimization Algorithm (WOA) is an optimization technique inspired by the hunting behavior of humpback whales. This nature-inspired algorithm is utilized to optimize solutions in various domains. The algorithm follows three main phases: encircling prey, bubble-net attacking, and exploring for prey. These phases simulate solution exploitation strategy and diversification of the search, respectively. The WOA is well-suited algorithm in different fields because of its simplicity and adaptability.

The Particle Swarm Optimization algorithm draws inspiration from the collective behavior of birds or fish, and is a robust computational algorithm for optimization problems. Its working principle relies on the process of iteratively refining candidate solutions based on a given measure of quality. PSO begins by generating a group of random particles, and then updates them over time to find the best possible solution. These particles move through the solution space by tracking the current optimal particle while also considering their own historically best positions. Thanks to its efficiency, PSO is a widely used optimization algorithm for a variety of problems.

Mixed-integer linear Programming (MILP) is a powerful mathematical optimization technique that is specifically designed to model and solve optimization problems that have linear or integer constraints. MILP formulations are particularly useful for complex IPPS problems that require resource allocation, precedence constraints, and multiple objectives. To solve these formulations, commercially available software such as Gurobi (Achterberg, 2019) is highly efficient and effective, and can provide optimal or near-optimal solutions to even the most challenging IPPS problems.

1.1. Optimization algorithm in IPPS

Numerous research studies have been carried out to analyze optimization algorithms on different benchmarks for IPPS, due to the no free lunch theory of optimization, which states that there is no one best optimization algorithm for all problems (Wolpert & Marcredy, 1997). Therefore, in the following paragraphs, we analyze different research studies regarding IPPS optimization. The paper (Petrović et al., 2019) introduces an innovative methodology that applies the Whale Optimization Algorithm to the IPPS with constraints regarding a single mobile robot. The authors propose numerous objective functions, and different datasets to test the enhanced version of the WOA algorithm. The experimental results demonstrate that WOA achieves better results compared to other optimization algorithms. In the paper (Homayouni & Fontes, 2019) the authors focus on developing an integrated formulation to address the joint production and transportation scheduling problem in flexible manufacturing environments. The study emphasizes the necessity of simultaneous scheduling of machines and Automatic Guided Vehicles (AGVs), as they are closely interconnected in manufacturing systems where parts need to be transported across various machines for different operations. The authors propose a novel MILP model that incorporates two sets of chained decisions: one for machine scheduling and another for AGV scheduling. These sets are linked through completion time constraints for both machine operations and transportation tasks. Computational experiments conducted using the Gurobi commercial software on benchmark problem demonstrate the effectiveness of the proposed model in finding optimal solutions.

The study detailed in (Homayouni et al., 2020) explores the utilization of a multistart biased random key genetic algorithm augmented with a greedy heuristic to discover high-quality solutions for the job shop scheduling problem with transportation constraints. This approach synchronizes four interconnected aspects: job routing, machine scheduling, vehicle allocation, and transportation timing, with the goal of reducing the total completion time, or makespan. The experimental findings underscore the method's effectiveness across various scheduling problem categories, validated through testing on over 60 cases across two problem sets. Building on this, the authors (Homayouni & Fontes, 2021) introduced a late acceptance hill-climbing strategy to prevent the algorithm from stagnating at local maxima. This technique underwent testing on five datasets, demonstrating its efficacy in optimizing smaller and medium-sized problem instances and delivering competitive outcomes for larger scenarios. The authors of the paper (Petrović et al., 2022) present an innovative methodology using the multi-objective Grey Wolf Optimizer (GWO) to efficiently perform IPPS with material transport systems in intelligent manufacturing systems. The methodology includes a comprehensive analysis, mathematical formulation of 13 novel fitness functions, and a strategy for optimal exploration of the multi-objective search space. The effectiveness of the enhanced GWO algorithm is quantitatively compared with other metaheuristic methods across 25 benchmarks. The experimental results indicate that the enhanced GWO algorithm outperforms the other algorithms in terms of convergence, coverage, and robustness in finding optimal Pareto solutions. The paper (Utama et al., 2024) introduces a novel application of the MGO algorithm for optimizing the no-wait flow shop scheduling problem with the aim of minimizing industrial energy consumption. The MGO algorithm is implemented with the Large Rank Value procedure. The MGO is compared to GWO, GA, PSO, Coati Optimization Algorithm, and Fire Hawk Optimizer, on three different experimental setups. The One-Way ANOVA statistical tests were performed to show the statistical significance of the obtained results, which show the advantages of the proposed MGO algorithm.

Different from other approaches in this paper, the Mountain Gazelle Optimizer (MGO) algorithm is utilized for IPPS with single transportation vehicle constraints. The MGO was selected due to its advantages regarding diverse strategies for exploitation and exploration, which are necessary for finding the optimal solution for optimization problems with such vast solution space, such as IPPS.

2. THE MOUNTAIN GAZELLE OPTIMIZER

The Mountain Gazelle Optimizer represented one of the newly developed nature-inspired population-based optimization algorithms, which found its inspiration behind the social hierarchical structure of mountain gazelle herd (Fig. 1).

Figure 1. Social structure of mountain gazelles with male in the middle

The mathematical framework of the MGO algorithm includes four behaviors of mountain gazelles: 1. Territorial Solitary Males (TSM), 2. Maternity Herds (MH), 3. Bachelor Male Herds (BMH), and 4. Migration in Search for Food (MSF). Each gazelle in the population represents a solution to the optimization problem (*X*) with *D* solution parameters. Many random numbers are defined within the MGO algorithm, and their notations are as follows. The *r* defines random numbers that undergo uniform distribution within [0, 1] range, vectors of random numbers drawn from normal distribution with zero mean and standard deviation of one are defined as *N*(*D*), with D being number of elements, and random integers in [1, 2] range are defined as r_i. In order to mathematically define four behaviors, firstly, four coefficients need to be defined (1):

$$Cof = \begin{cases} a + 1 + r_1 \\ a \cdot N_1(D) \\ r_2(D) \\ N_2(D) \cdot N_3(D)^2 \cdot cos\big(2r_3 \cdot N_4(D)\big) \end{cases}, \quad (1)$$

where $a = -1 + iter \cdot \left(\frac{-1}{max\_iter}\right)$. Afterward, vector *F* is defined as (2):

$$F = N_5(D) \cdot exp\left(2 - iter \cdot \frac{2}{max\_iter}\right). \quad (2)$$

The second part of the multiplication of F starts with values larger than 1 (depending on the maximum number of interactions) and exponentially converges to 1 with iterations, leaving a simple normal random vector in the last iteration. Now, all relevant values are defined to calculate the young male heard coefficient vector defined as *BH* (3):

$$BH = X_{ra} \cdot r_1 + M_{pr} \cdot r_2, \tag{3}$$

where X_{ra} is a randomly selected solution from the last third of the population; since the solutions are sorted in the acceding order, these represent the worst 33% of the solutions in the entire population. M_{pr} is the mean value for the selected 33% of the population, averaged for each dimension in the input vector. The TSM (4) aspect of the algorithm models the behavior of adult male gazelles that establish and defend territories. It is used in the algorithm to enhance the exploitation ability, allowing the optimizer to search intensively around the best solutions found so far:

$$TSM = X_1 - |(r_{i1} \cdot BH - r_{i2} \cdot X_t) \cdot F| \cdot Cof_r, \tag{4}$$

where X_1 is the best solution obtained so far, X_t is the currently updated agent, and Cof_r is the randomly selected coefficient from (1).

The second behavior, MH (5), consists of females and their offspring, reflecting a balance between exploration and exploitation in the algorithm. This mechanism ensures diversity in the solution space and prevents premature convergence:

$$MH = BH + Cof_r + (r_{i3} \cdot X_1 - r_{i4} \cdot X_{rand}) \cdot Cof_r, \tag{5}$$

where X_{rand} represented a randomly selected solution from the population.

The parameter *Dist* (6) needs to be calculated to model Bachelor Male Herds behavior:

$$Dist = |X_t - X_1| \cdot (2r_6 - 1). \tag{6}$$

The third behavior, BMH (7), represents the young male gazelles, and it is used to explore new areas in the search space, contributing to the algorithm's exploration capabilities.

$$BMH = X_t - Dist + (r_{i5} \cdot X_1 - r_{i6} \cdot BH) \cdot Cof_r, \tag{7}$$

Finally, Migration in Search for Food (8) is modeled with a random search mechanism, that allows algorithm to avoid local optima and ensure comprehensive exploration of the search space:

$$MSF = (lb - ub) \cdot r_7 + lb, \tag{8}$$

where *lb* and *ub* are the lower and upper bounds of the parameter space. As it can be seen, MSF is a uniform random sampling of values in the parameter space, which allows MGO to search the entire parameter space even if the initial solutions are not generated well.

2.1. The MGO algorithm for IPPS

Process plans within the production environment are characterized by different types of flexibilities that can influence the final scheduling plan. This paper considers the following flexibilities: process plan, sequencing, machine, and tools. Process plan flexibility means that each job can be done by employing different manufacturing operations or sequences of operations. Sequencing refers to the ability to change the order of manufacturing operations within a job. Machine and tool flexibilities refer to the possibility of selecting alternative machines and tools for each manufacturing operation. Moreover, jobs are transported between machines by a single mobile robot. For the IPPS problem, each valid solution contains four strings (Petrović et al., 2019) that are used to represent a selected sequence of operations, process plans, selected machines and tools for each operation. Therefore, the optimization process aims to select the optimal values for all strings, which results in the minimal time required for the machining of all jobs, i.e., minimizing the makespan cost function. Finally, the implementation of MGO optimization for the IPPS problem is given in Table 1.

Table 1. Pseudo-code of MGO algorithm implemented for IPPS problem

1.	**Input**: Data for (i) the set of jobs, (ii) the set of alternative process plans for each job, (iii) the set of available machines and tools for each operation, processing times for all operations, and transport times between all machines. Definition of algorithm parameters: population size (*N*), maximum number of generations.
2.	Initialization of all strings for the whole population
3.	Calculation of cost function value for each gazelle
4.	**for #1** every generation
5.	Select a leader
6.	**for #2** every gazelle
7.	Generate new solution for all IPPS strings according to equations (4), (5), (7), (8)
8.	**endfor #2**
9.	Calculate the cost function value for the entire new population and select *N* best solutions for the new generation
10.	**endfor #1**
11.	Save results

3. EXPERIMENTAL RESULTS

An experimental evaluation of the proposed algorithm is carried out on a benchmark containing 9 problems, where each problem contains 6 jobs manufactured on 10 machines using 20 different tools. The proposed MGO algorithm is compared to two state-of-the-art algorithms optimization algorithms, namely WOA and PSO. Each algorithm is evaluated five times on each problem. The results of the experimental evaluation are shown in Table 2. The algorithms are compared based on two metrics, the best and average value achieved within five experimental evaluations. As it can be seen, the MGO algorithm achieves most (4/9) best results, which demonstrates the advantage of the MGO's convergence properties. On the other hand, both MGO and PSO achieved the same number of average best results, indicating that MGO can get stuck in local optima in certain experimental evaluations.

Table 2. Experimental results

Problem	best			average		
	WOA	PSO	MGO	WOA	PSO	MGO
1	**133**	163	146	**172**	201	173
2	169	192	**149**	179	230	**172**
3	288	**240**	248	309	**265**	**265**
4	143	**126**	140	168	**133**	159
5	248	**238**	262	277	**251**	286
6	**157**	202	170	**170**	227	230
7	287	271	**254**	329	293	**282**
8	145	135	**129**	155	161	**135**
9	150	137	**108**	175	**151**	175

The convergence curves of the best evaluation of all three algorithms for problems 2, 8, and 9 can be seen in Figure 2. For problems 2 and 8, all the algorithms converge to their optimal values within the first 40 iterations, indicating that their exploration capabilities are sufficient for the considered problems. On the other hand, for problem 9, PSO still manages to achieve a better solution even after 47 generations, indicating that it can benefit from a larger number of generations.

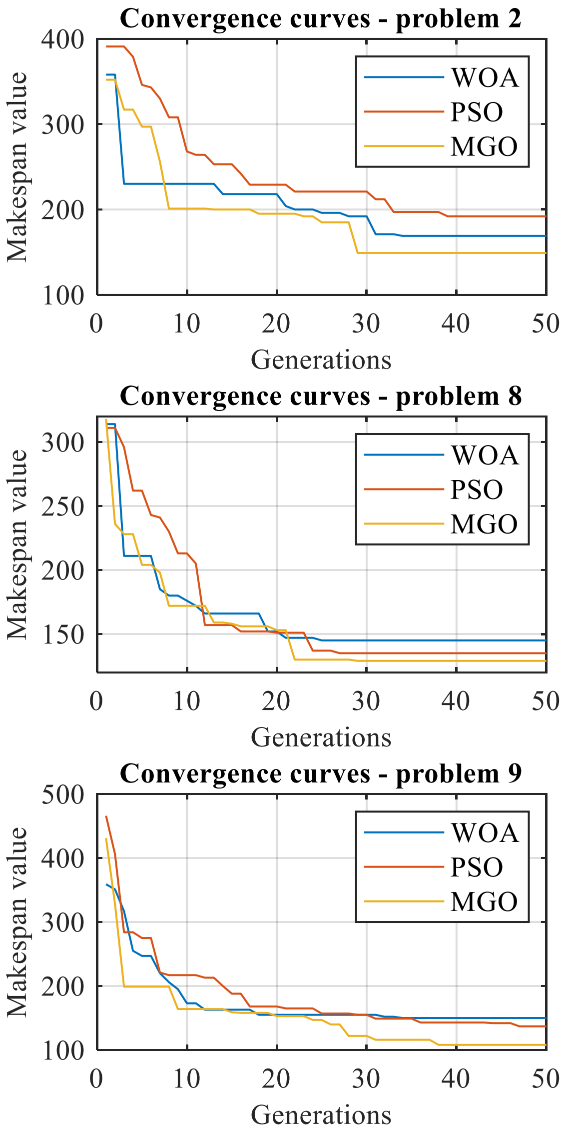

Figure 2. Convergence curves for the best run of three analyzed algorithms

Figure 3. shows the Gantt charts for the best experimental evaluation of the MGO algorithm, again for problems 2, 8, and 9. Each Gantt chart is utilized to represent the sequence of manufacturing operations, the machines that are used, operation duration, and actions the mobile robot needs to perform to transport the jobs from machine to machine. Mobile robot has three actions that it can perform: moving jobs to the machine for the subsequent operation, moving to the machine where the previous operation of the job is manufactured, and waiting for the machine to finish the current operation of the job.

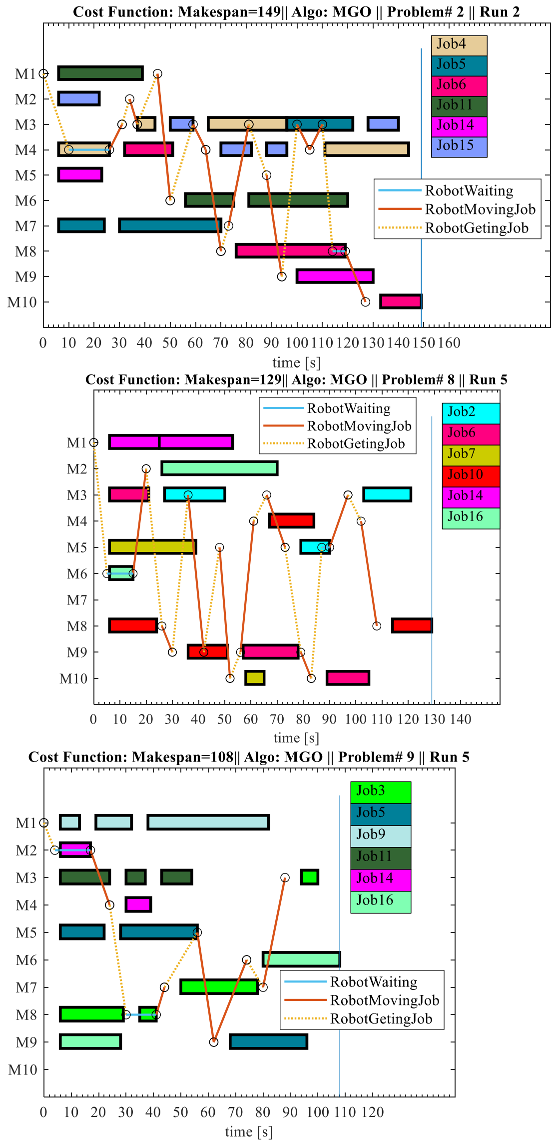

Figure 3. Gantt charts of three problems

4. CONCLUSION

In this paper, the authors propose a novel approach for integrated process planning and scheduling problems based on the mountain gazelle optimization algorithm. The optimization of the production process planning and scheduling is performed based on the makespan metric. The proposed algorithm is compared to two state-of-the-art optimization algorithms. After experimental evaluation with nine different problems containing six jobs, 10 machines, and 20 tools, the proposed MGO algorithm has shown the best convergence properties. However, the MGO achieved the same number of average best results as the PSO. Therefore, improvements in the exploration capabilities of the MGO algorithm can be investigated in the future.

ACKNOWLEDGMENT

This research has been financially supported by the Ministry of Education, Science and Technological Development of the Serbian Government, through the project "Integrated research in macro, micro, and nano mechanical engineering – Deep learning of intelligent manufacturing systems in production engineering" (contract No. 451-03-65/2024-03/200105, 05/02/2024).

REFERENCES

Abdollahzadeh, B., Gharehchopogh, F. S., Khodadadi, N., & Mirjalili, S. (2022). Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. *Advances in Engineering Software, 174*, 103282.

Achterberg, T. (2019). What's new in Gurobi 9.0. *Webinar Talk Url: Https://Www. Gurobi. Com/Wp-Content/Uploads/2019/12/Gurobi-90-Overview-Webinar-Slides-1. Pdf*, *5*(9), 97–113.

Caumond, A., Lacomme, P., Moukrim, A., & Tchernev, N. (2009). An MILP for scheduling problems in an FMS with one vehicle. *European Journal of Operational Research*, *199*(3), 706–722. https://doi.org/10.1016/j.ejor.2008.03.051

Fontes, D. B. M. M., & Homayouni, S. M. (2019). Joint production and transportation scheduling in flexible manufacturing systems. *Journal of Global Optimization*, *74*(4), 879–908. https://doi.org/10.1007/s10898-018-0681-7

Homayouni, S. M., Fontes, D. B. M. M., & Gonçalves, J. F. (2020). A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation. *International Transactions in Operational Research*, 1–29. https://doi.org/10.1111/itor.12878

Homayouni, S. M., & Fontes, D. B. M. M. (2021). Production and transport scheduling in flexible job shop manufacturing systems. *Journal of Global Optimization*, *79*(2), 463–502. https://doi.org/10.1007/s10898-021-00992-6

Mirjalili, S., & Lewis, A. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Petrović, M., Jokić, A., Miljković, Z., & Kulesza, Z. (2022). Multi-objective scheduling of a single mobile robot based on the grey wolf optimization algorithm. *Applied Soft Computing*, *131*, 109784. https://doi.org/10.1016/j.asoc.2022.109784

Petrović, M., Miljković, Z., & Jokić, A. (2019). A novel methodology for optimal single mobile robot scheduling using whale optimization algorithm. *Applied Soft Computing*, *81*, 105520. https://doi.org/10.1016/j.asoc.2019.105520

Petrović, M., Vuković, N., Mitić, M., & Miljković, Z. (2016). Integration of process planning and scheduling using chaotic particle swarm optimization algorithm. *Expert Systems with Applications*, *64*, 569–588. https://doi.org/10.1016/j.eswa.2016.08.019

Phanden, R. K., Jain, A., & Davim, J. P. (2019). *Integration of Process Planning and Scheduling: Approaches and Algorithms*. CRC press.

Utama, D. M., Sanjaya, B. D., & Nugraha, A. (2024). A Modified Mountain Gazelle Optimizer for Minimizing Energy Consumption on No-Wait Permutation Flow Shop Scheduling Problem. *International Journal of Intelligent Engineering and Systems*, *17*(2), 609–620. https://doi.org/10.22266/ijies2024.0430.49

Wolpert, D., & Marcredy, W. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67–82. https://doi.org/10.1145/1389095.1389254