Production Systems and Information Engineering
Volume 13 (2), pp. 16-26
doi: https://doi.org/10.32968/psaie.2025.2.2

Application of Harmony Search on the Taillard Dataset

ANITA AGÁRDI[1]
University of Miskolc, Hungary
Institute of Information Technology
agardianita@iit.uni-miskolc.hu

Abstract. This article examines the effectiveness of Harmony Search on a common production scheduling problem, the Flow Shop Scheduling. During the task, a given number of jobs must be performed by a given number of machines. The goal is to determine the order of the jobs so that the makespan is minimized. Harmony Search is a metaheuristic algorithm that maintains a population of solutions. The algorithm is inspired by musical composition. The algorithm continuously improves its solutions and returns with the best solution when the stopping condition is met. The paper presents Harmony Search, the Flow Shop Scheduling Problem, and then the running results. The running results are also compared with the results published by other researchers.

Keywords: Harmony Search, Taillard dataset, Flow Shop Scheduling

EARLY ACCESS

1. Introduction

The aim of this paper is to analyze the efficiency of Harmony Search [1] on the Taillard [2] dataset, which is a Flow Shop Scheduling [3] task. Harmony Search is a metaheuristic algorithm that maintains a population of solutions. It models the improvement of musical harmony.

Flow Shop Scheduling [3] is one of the most common problems in production scheduling. The task involves processing multiple workpieces in a predetermined order on consecutive machines. Each workpiece passes through the same machine line, and only one operation can be performed on each machine. The goal is usually to minimize the makespan, optimize machine utilization, or reduce delays. Such scheduling problems are particularly important in the automotive industry, electronics manufacturing, and other mass production systems.

Flow Shop Scheduling (FSS) [3] is a scheduling problem in which a given number of workpieces are to be processed on a predetermined line of machines, where each machine performs a specific operation. Each workpiece passes through the machines in the same order. FSS problems are particularly important in manufacturing and logistics systems. Optimal scheduling can significantly reduce production time, increase capacity utilization, and reduce costs.

Several heuristic and metaheuristic algorithms have been developed to solve them,

---

[1] The author used ChatGPT (OpenAI) to assist in the textual formulation of several section of the manuscript. The author reviewed and edited the AI-generated content for clarity, accuracy and coherence, and take full responsibility for the final version.

such as Genetic Algorithms (GA) [4], Tabu Search (TS) [5], Bee Colony Optimization (ABC) [6], and Harmony Search (HS) [7].

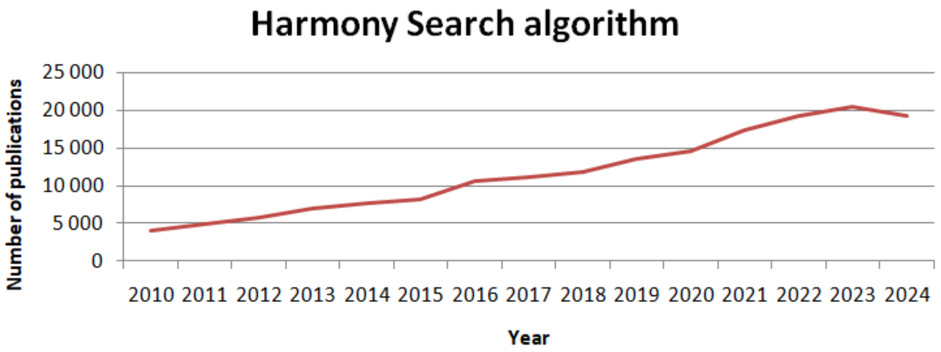


Figure 1. Publication frequency of the Harmony Search algorithm

The Harmony Search algorithm was first published by Geem, Kim, and Loganathan in 2001. Title of the article: A new heuristic optimization algorithm: Harmony search. Place of publication: Proceedings of the 2001 Congress on Evolutionary Computation (CEC 2001)
The Harmony Search algorithm has shown a steady growth over the years. In 2010, the number of published articles was less than 5,000, but in 2023, this increased to more than 20,000.

2. Harmony Search

Harmony Search [8] is a metaheuristic algorithm inspired by musical harmonies. The algorithm was developed by Geem et al. in 2001. It models the process of improving musical harmonies. Musicians combine individual notes to create a harmonious melody. The algorithm is an evolutionary algorithm, maintaining a population of solutions. It continuously improves musical harmonies (i.e., the elements of the population, the possible solutions) in each iteration.

The algorithm has the following important parameters:

- HM (Harmony Memory): Harmony memory, the population of solutions.
- HMS (Harmony Memory Size): Harmony memory size, the number of elements in the population.
- HMCR (Harmony Memory Considering Rate): The probability of choosing from the harmony memory.
- PAR (Pitch Adjustment Rate): The probability of fine-tuning a solution.

1. **Initialization**: Initialization of the population (initial harmony). Either with randomly generated possible solutions or with solutions given by a construction algorithm. Initialization of the HM (Harmony Memory) according to the following formula:

$$\mathrm{HM} = \begin{bmatrix} \mathrm{x}_{1,1} & \cdots & \mathrm{x}_{1,\mathrm{D}} \\ \vdots & \ddots & \vdots \\ \mathrm{x}_{\mathrm{HMS},1} & \cdots & \mathrm{x}_{\mathrm{HMS},\mathrm{D}} \end{bmatrix} i = 1, \ldots, HMS, J = 1, \ldots D$$

EARLY ACCESS

2. **Harmony improvement**: The creation of the new sound (solution) occurs in this phase. Here the algorithm takes into account the harmony memory and the pitch adjustment rate.
    - Selecting from Harmony Memory (HMCR)

      $$x_j^{new} \in \{x_{1,j}, x_{2,j}, \dots, x_{HMS,j}\} \text{ with probability of HMCR}$$

    - Pitch adjustment (PAR): fine-tuning the selected element:

      $x_j^{new} = x_j^{new} + \delta$, where $\delta \backsim U(-b, b)$ where b a predetermined step size, with probability PAR

    - Choosing a randomly generated value: if the HMCR and PAR conditions are not met, the new solution is randomly generated:

      $$x_j^{new} = \begin{cases} x_j^{new}, from\ HM\ with\ probability\ HMCR \\ x_j^{new} + \delta, \text{with probability PAR} \\ random\ U\left(x_j^{min}, x_j^{max}\right), otherwise \end{cases}$$

    -
3. **Updating:** New solutions are entered into the harmony memory, and the best solutions are carried forward by the algorithm to the next iteration.
4. **Termination condition:** the termination condition can be reaching a certain number of iterations, convergence, or reaching a certain running time.

3. Test results

This section presents the results of the Harmony Search run. First, the maximum, average and minimum fitness values for each data set of the algorithm runs. Then, the results are compared to the results published by individual researchers. Here, the results are also presented with a bar chart and a heat map.

Table 1. Test results of the Harmony Search algorithm

Instance	**HS**		
	Max	**Avg**	**Min**
Ta001	1359	1343.4	1309
Ta002	1402	1393	1383
Ta003	1242	1232.8	1227
Ta004	1472	1452.4	1428
Ta005	1357	1341.6	1324
Ta006	1382	1322.6	1276
Ta007	1335	1322.6	1312
Ta008	1382	1357.4	1334
Ta009	1398	1375.4	1352
Ta010	1277	1251.6	1233

EARLY ACCESS

Ta011	1833	1806.2	1782
Ta012	1885	1857.2	1829
Ta013	1737	1710.8	1670
Ta014	1606	1584.2	1558
Ta015	1683	1644.6	1577
Ta016	1602	1570	1554
Ta017	1682	1665.4	1650
Ta018	1757	1748	1743
Ta019	1792	1767.4	1728
Ta020	1823	1794	1762

Table 2 shows the results of Harmony Search according to the following test values: maximum (Max), average (Avg), and minimum (Min).

Smaller benchmark data (Ta001–Ta010): For the first ten test examples, the difference between the average and minimum values ranges from 10 to 60. For example: for Ta006, the maximum is 1382, the average is 1322.6, and the minimum is 1276. For Ta001, the difference is smaller but still significant (1359 → 1343.4 → 1309).

Larger examples (Ta011–Ta020): These datasets also show high variability. For example: for Ta015, the difference between the minimum and maximum value is 106. For Ta019, this is 64 fitness units (1792 → 1767.4 → 1728). However, for example Ta018, the differences are minimal.

The data in the table shows that for the Harmony Search algorithm, there is a greater variability between different runs. The performance of the algorithm is less stable across different runs.

Table 2. The minimum values of the test results of the Harmony Search algorithm

Instance	HS	HMM-PFA %	HGA %	IIGA %	DSOMA %	HGSA %
Ta001	1309	13.52	10.70	13.52	4.97	1.15
Ta002	1383	10.48	5.57	10.48	1.81	4.27
Ta003	1100	32.73	26.00	32.73	16.36	-0.18
Ta004	1362	16.59	11.67	16.59	6.31	7.86
Ta005	1298	11.63	8.09	11.63	3.31	-0.54
Ta006	1250	18.48	14.40	18.48	9.04	11.28
Ta007	1257	17.98	16.23	17.98	9.86	3.34
Ta008	1240	19.52	15.56	19.52	11.21	4.19
Ta009	1255	17.05	11.39	17.05	9.40	4.06
Ta010	1161	18.60	14.04	18.60	10.51	6.20
Ta011	1649	23.95	18.56	21.95	2.97	3.88
Ta012	1810	19.67	17.29	19.67	1.27	-5.08
Ta013	1627	19.24	17.52	19.24	3.01	-4.43
Ta014	1487	21.79	19.84	21.79	3.97	1.95
Ta015	1536	25.85	25.85	25.85	5.27	2.41
Ta016	1466	29.06	24.62	29.06	8.46	-0.61
Ta017	1555	26.24	25.02	26.24	4.31	4.31
Ta018	1651	24.59	21.50	24.59	4.85	5.94
Ta019	1676	17.72	13.84	17.72	4.24	-3.10
Ta020	1686	21.65	18.68	21.65	5.69	2.14

Table 3 presents the Harmony Search values (maximum) and compares them with the fitness values published by the researchers [9-10]. The table shows that in some cases the best Harmony Search result was worse than the results published by the

EARLY ACCESS

researchers. However, in most cases the HS result was even 10-20% better than the results published by the researchers.

According to the data in the table, in most cases the HS algorithm found solutions 10-20% better than the best results published by the researchers. This suggests that the algorithm has an efficient search strategy. In some cases its results were weaker than the published values. In particular, a decline is observed in the following examples: Ta012 (-5.08%), Ta013 (-4.43%), Ta016 (-0.61%), Ta019 (-3.10%)

These results show that for certain problems, Harmony Search may not find the optimal solution or may perform with greater variability.

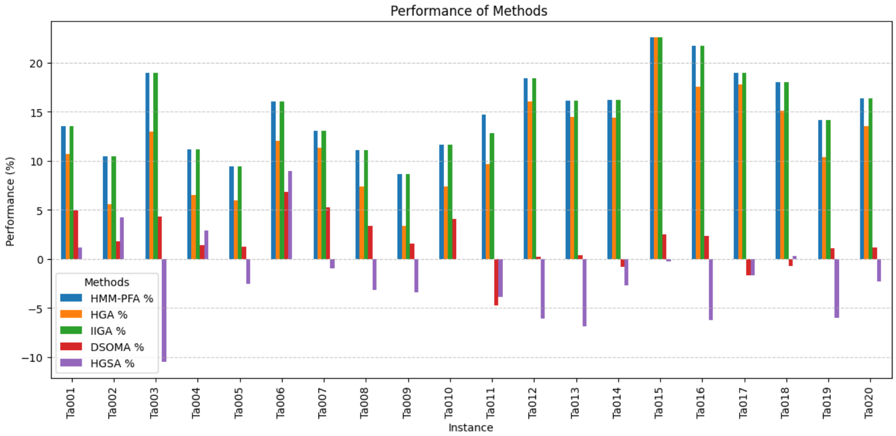
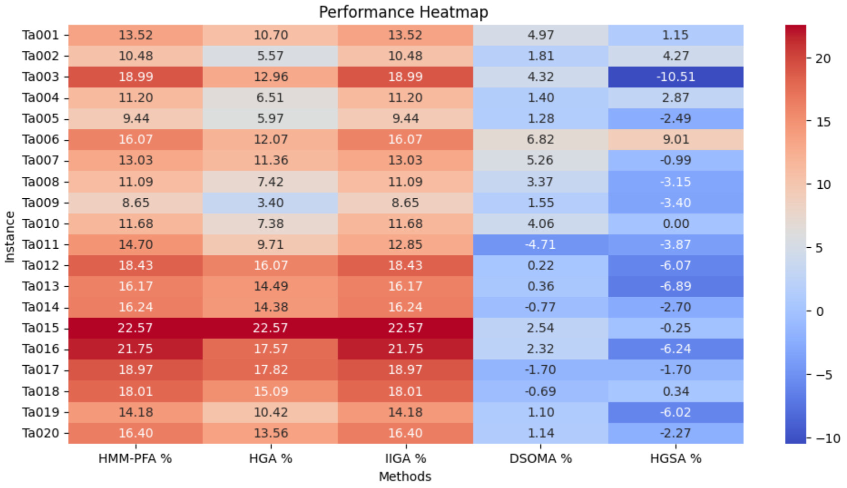


Figure 2. The minimum values of the test results of the Harmony Search algorithm

Figure 2 shows the minimum running values of the Harmony Search in the form of a bar chart. It can be seen that in some cases the HGSA and DSOMA algorithms were better than HS.



	HMM-PFA %	HGA %	IIGA %	DSOMA %	HGSA %
Ta001	13.52	10.70	13.52	4.97	1.15
Ta002	10.48	5.57	10.48	1.81	4.27
Ta003	18.99	12.96	18.99	4.32	-10.51
Ta004	11.20	6.51	11.20	1.40	2.87
Ta005	9.44	5.97	9.44	1.28	-2.49
Ta006	16.07	12.07	16.07	6.82	9.01
Ta007	13.03	11.36	13.03	5.26	-0.99
Ta008	11.09	7.42	11.09	3.37	-3.15
Ta009	8.65	3.40	8.65	1.55	-3.40
Ta010	11.68	7.38	11.68	4.06	0.00
Ta011	14.70	9.71	12.85	-4.71	-3.87
Ta012	18.43	16.07	18.43	0.22	-6.07
Ta013	16.17	14.49	16.17	0.36	-6.89
Ta014	16.24	14.38	16.24	-0.77	-2.70
Ta015	22.57	22.57	22.57	2.54	-0.25
Ta016	21.75	17.57	21.75	2.32	-6.24
Ta017	18.97	17.82	18.97	-1.70	-1.70
Ta018	18.01	15.09	18.01	-0.69	0.34
Ta019	14.18	10.42	14.18	1.10	-6.02
Ta020	16.40	13.56	16.40	1.14	-2.27

Figure 3. The minimum values of the test results of the Harmony Search algorithm

Figure 3 compares the test values of the Harmony Search algorithm with the results published by the researchers in the form of a heat map. In many cases, the results are red, which also indicates that Harmony Search was better than the results published by the researchers.

Table 3. The average values of the test results of the Harmony Search algorithm

Instance	HS	HMM-PFA %	HGA %	IIGA %	DSOMA %	HGSA %
Ta001	1343.4	10.61	7.86	10.61	2.28	-1.44
Ta002	1393	9.69	4.81	9.69	1.08	3.52
Ta003	1232.8	18.43	12.43	18.43	3.83	-10.93
Ta004	1452.4	9.34	4.72	9.34	-0.30	1.14
Ta005	1341.6	8.01	4.58	8.01	-0.04	-3.77
Ta006	1322.6	11.98	8.12	11.98	3.05	5.17
Ta007	1322.6	12.13	10.46	12.13	4.42	-1.78
Ta008	1357.4	9.18	5.57	9.18	1.59	-4.82
Ta009	1375.4	6.81	1.64	6.81	-0.17	-5.05
Ta010	1251.6	10.02	5.78	10.02	2.51	-1.49
Ta011	1806.2	13.17	8.24	11.34	-5.99	-5.16
Ta012	1857.2	16.63	14.31	16.63	-1.30	-7.50
Ta013	1710.8	13.40	11.76	13.40	-2.03	-9.11
Ta014	1584.2	14.32	12.49	14.32	-2.41	-4.31
Ta015	1644.6	17.54	17.54	17.54	-1.68	-4.35
Ta016	1570	20.51	16.37	20.51	1.27	-7.20
Ta017	1665.4	17.87	16.73	17.87	-2.61	-2.61
Ta018	1748	17.68	14.76	17.68	-0.97	0.06
Ta019	1767.4	11.63	7.96	11.63	-1.15	-8.11
Ta020	1794	14.33	11.54	14.33	-0.67	-4.01

Table 3 compares the averages of Harmony Search runs with the results published by the researchers. Here, the negative percentage is higher, so the results published by the researchers were better in more places.

In several cases, the results published by the researchers were better. The number of negative deviations in the table increased, which indicates that the average Harmony Search performance was weaker in some cases than the previously published solutions.

For example, in the cases Ta011 (-5.99%), Ta012 (-1.30%), Ta013 (-2.03%), Ta014 (-2.41%), Ta015 (-1.68%), Ta017 (-2.61%), the HS algorithm did not achieve the results published by the researchers.

Harmony Search was still better in many cases, but with a smaller advantage. Although in the majority of cases Harmony Search achieved better results than those published by the researchers. There is still an improvement of around 20% in some cases, e.g. Ta016: +20.51%, Ta016 (+20.51%), Ta015 (+17.54%), Ta017 (+17.87%), Ta012 (+16.63%).

Moderate improvements can be observed in the following cases: Ta003 (+18.43%), Ta018 (+17.68%), Ta014 (+14.32%), Ta020 (+14.33%).

HGSA (Hybrid Genetic Simulated Annealing) was often better than the Harmony Search algorithm. In the case of the HGSA algorithm, many negative deviations are visible. For example, in the case of Ta003 (-10.93%), Ta012 (-7.50%), Ta013 (-9.11%), Ta016 (-7.20%), HGSA produced significantly better results.

DSOMA also often produced better results. Cases Ta011 (-5.99%), Ta012 (-1.30%), Ta013 (-2.03%) show that the DSOMA algorithm is better than Harmony Search in certain situations.

EARLY ACCESS

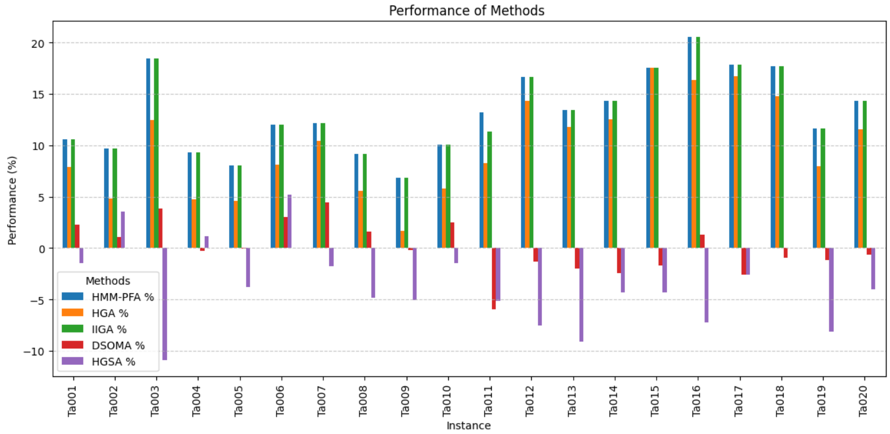
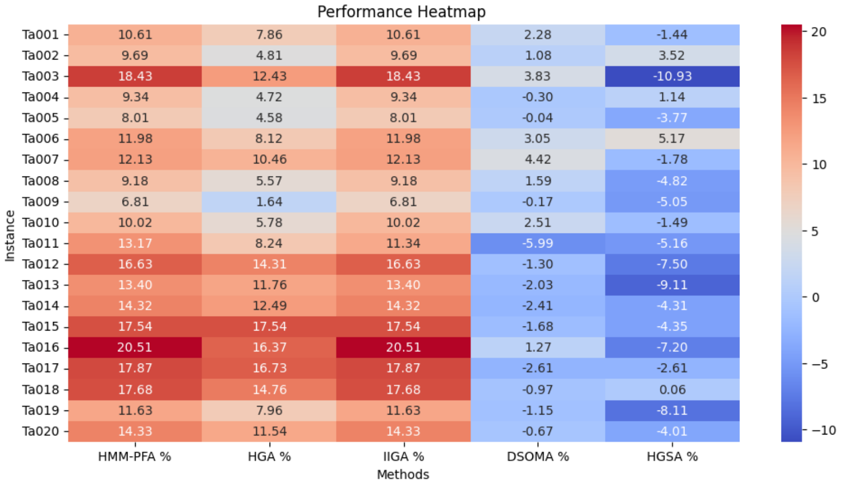


Figure 4. The average values of the test results of the Harmony Search algorithm

Figure 4 shows the average of Harmony Search test results, comparing them with the results published by the researchers in the form of a bar chart.



Instance	HMM-PFA %	HGA %	IIGA %	DSOMA %	HGSA %
Ta001	10.61	7.86	10.61	2.28	-1.44
Ta002	9.69	4.81	9.69	1.08	3.52
Ta003	18.43	12.43	18.43	3.83	-10.93
Ta004	9.34	4.72	9.34	-0.30	1.14
Ta005	8.01	4.58	8.01	-0.04	-3.77
Ta006	11.98	8.12	11.98	3.05	5.17
Ta007	12.13	10.46	12.13	4.42	-1.78
Ta008	9.18	5.57	9.18	1.59	-4.82
Ta009	6.81	1.64	6.81	-0.17	-5.05
Ta010	10.02	5.78	10.02	2.51	-1.49
Ta011	13.17	8.24	11.34	-5.99	-5.16
Ta012	16.63	14.31	16.63	-1.30	-7.50
Ta013	13.40	11.76	13.40	-2.03	-9.11
Ta014	14.32	12.49	14.32	-2.41	-4.31
Ta015	17.54	17.54	17.54	-1.68	-4.35
Ta016	20.51	16.37	20.51	1.27	-7.20
Ta017	17.87	16.73	17.87	-2.61	-2.61
Ta018	17.68	14.76	17.68	-0.97	0.06
Ta019	11.63	7.96	11.63	-1.15	-8.11
Ta020	14.33	11.54	14.33	-0.67	-4.01

Figure 5. The average values of the test results of the Harmony Search algorithm

Figure 5 compares the average of Harmony Search test result in the form of a heat map with the results published by the researchers.

Table 4. The maximum values of the test results of the Harmony Search algorithm

Instance	HS	HMM-PFA %	HGA %	IIGA %	DSOMA %	HGSA %
Ta001	1359	9.35	6.62	9.35	1.10	-2.58
Ta002	1402	8.99	4.14	8.99	0.43	2.85
Ta003	1242	17.55	11.59	17.55	3.06	-11.59
Ta004	1472	7.88	3.33	7.88	-1.63	-0.20
Ta005	1357	6.78	3.39	6.78	-1.18	-4.86
Ta006	1382	7.16	3.47	7.16	-1.37	0.65
Ta007	1335	11.09	9.44	11.09	3.45	-2.70

EARLY ACCESS

Ta008	1382	7.24	3.69	7.24	-0.22	-6.51
Ta009	1398	5.08	0.00	5.08	-1.79	-6.58
Ta010	1277	7.83	3.68	7.83	0.47	-3.45
Ta011	1833	11.51	6.66	9.71	-7.36	-6.55
Ta012	1885	14.91	12.63	14.91	-2.76	-8.86
Ta013	1737	11.69	10.07	11.69	-3.51	-10.48
Ta014	1606	12.76	10.96	12.76	-3.74	-5.60
Ta015	1683	14.85	14.85	14.85	-3.92	-6.54
Ta016	1602	18.10	14.04	18.10	-0.75	-9.05
Ta017	1682	16.71	15.58	16.71	-3.57	-3.57
Ta018	1757	17.07	14.17	17.07	-1.48	-0.46
Ta019	1792	10.10	6.47	10.10	-2.51	-9.38
Ta020	1823	12.51	9.76	12.51	-2.25	-5.54

Table 4 contains the maximum running values of Harmony Search and compares them with the results published by the researchers. Here, the values are negative in many cases. This means that the results published by the researchers were better. Compared to the previous table, there are more negative deviations. The DSOMA and HGSA algorithms often performed better.

Harmony Search generally showed an improvement of between 0-10%, if it performed better. However, in some cases, it was still possible to achieve outstanding improvements (e.g. Ta016: +18.10%). Which shows that Harmony Search can still be very good for certain types of problems.

Much better results: Ta016 (+18.10%), Ta018 (+17.07%), Ta017 (+16.71%), Ta012 (+14.91%), Ta015 (+14.85%).

Moderate improvement can be observed in the following cases: Ta003 (+17.55%), Ta014 (+12.76%), Ta020 (+12.51%).

EARLY ACCESS

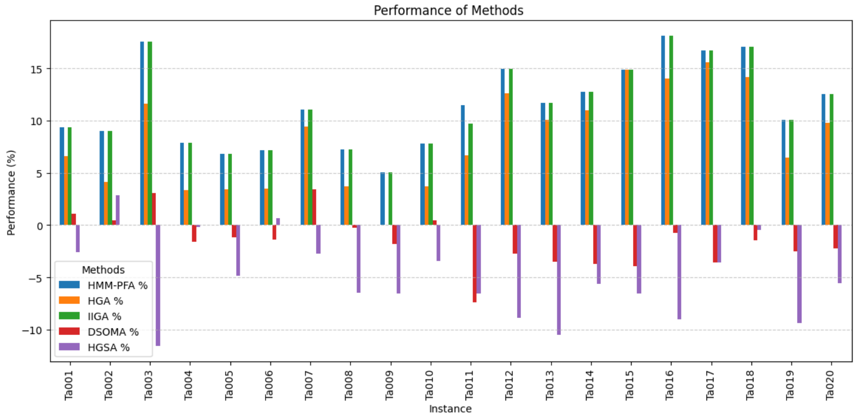


Figure 6. The average values of the test results of the Harmony Search algorithm

Figure 6 compares the maximum of Harmony Search running values with the results published by the researchers in the form of a bar chart.

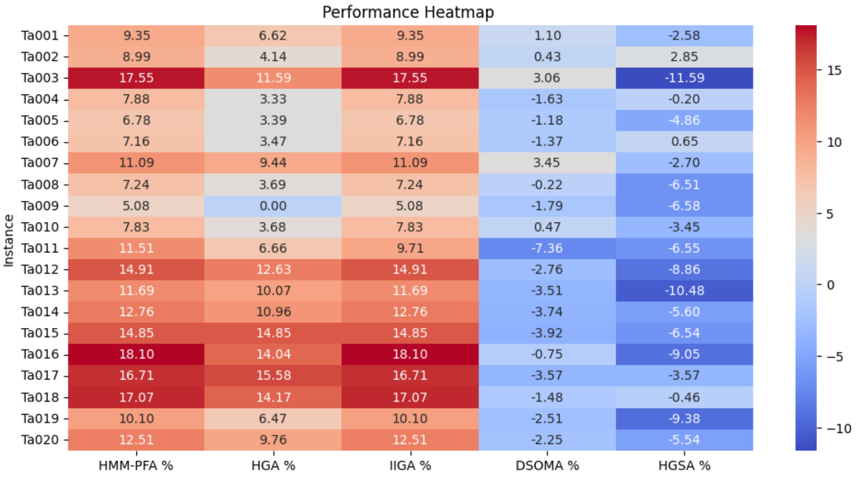


Figure 7. The average values of the test results of the Harmony Search algorithm

Figure 30 compares the maximum of Harmony Search's running values with the results published by the researchers in the form of a heat map.

Test results summary

The performance of the Harmony Search (HS) algorithm was evaluated on 20 Taillard test cases (Ta001–Ta020). The first table shows the best (Max), average (Avg), and worst (Min) values of HS based on 30 independent runs. The results indicate that HS finds stably good solutions with relatively low variance. For example, for Ta018, the difference between the best and worst values is only 14 units (Max = 1757, Min = 1743), indicating robust convergence of the method. In contrast, examples Ta006 or Ta015 show a larger deviation, suggesting that parameter tuning or additional local search could further improve stability. The following tables compare the performance of HS with five other modern metaheuristic algorithms (HMM-PFA, HGA, IIGA, DSOMA, HGSA), in the form of relative percentage deviations. The individual tables were prepared taking into account the minimum, maximum and average results of the HS runs. HS clearly outperforms its competitors in several examples. For example, in the case of Ta003, the value achieved by HS is 1100, while the other algorithms show a significantly larger deviation (e.g. HMM-PFA: 32.73%, HGA: 26.00%). A similar trend can be observed in the case of Ta016, where HS performs 29% better. However, in some examples (e.g. Ta012, Ta013, Ta019) the DSOMA or HGSA algorithm achieved a smaller deviation, which shows that they can be competitive with HS under certain conditions.

4. Conclusions and future research directions

Harmony Search (HS) is a metaheuristic optimization algorithm originally inspired by musical improvisation. The algorithm is often applied to combinatorial optimization problems. The research tested the efficiency of the algorithm on the Taillard dataset.

The results were compared with other metaheuristic algorithms published in the literature:

EARLY ACCESS

- HMM-PFA (Hybrid Multi-Mutation Population-based Firefly Algorithm),
- HGA (Hybrid Genetic Algorithm),
- IIGA (Improved Iterated Greedy Algorithm),
- DSOMA (Dynamic Self-Organizing Migrating Algorithm),
- HGSA (Hybrid Genetic Simulated Annealing).

It can be observed that Harmony Search was competitive in many cases, because it gave better results by around 10-20% in some cases.

A future research direction could be the development of Hybridized Harmony Search, for example with Simulated Annealing, Genetic Algorithm, or Tabu Search. Another research area could be adaptive parameter tuning, such as self-learning parameter updating. During the process, the algorithm continuously modifies its own settings according to the characteristics of the search space. Another direction could be the application of fuzzy logic or machine learning, in which the optimal parameters are selected based on prior learning or search patterns.

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.”

EKÖP
Egyetemi Kutatói Ösztöndíj Program
Ministry of Culture and Innovation

EARLY ACCESS

References

[1] Yang, X. S. (2009). Harmony search as a metaheuristic algorithm. Music-inspired harmony search algorithm: theory and applications, 1-14. https://doi.org/10.1007/978-3-642-00185-7_1

[2] E. Taillard, "Benchmarks for basic scheduling problems", EJOR 64(2):278-285, 1993. https://doi.org/10.1016/0377-2217(93)90182-M

[3] 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

[4] Etiler, O., Toklu, B., Atak, M. E. H. M. E. T., & Wilson, J. (2004). A genetic algorithm for flow shop scheduling problems. Journal of the Operational Research Society, 55(8), 830-835. https://doi.org/10.1057/palgrave.jors.2601766

[5] Saidi-Mehrabad, M., & Fattahi, P. (2007). Flexible job shop scheduling with tabu search algorithms. The international journal of Advanced Manufacturing technology, 32, 563-570. https://doi.org/10.1007/s00170-005-0375-4

[6] Zhang, R., Song, S., & Wu, C. (2013). A hybrid artificial bee colony algorithm for the job shop scheduling problem. International Journal of Production Economics, 141(1), 167-178. https://doi.org/10.1016/j.ijpe.2012.03.035

[7] Yuan, Y., Xu, H., & Yang, J. (2013). A hybrid harmony search algorithm for the flexible job shop scheduling problem. Applied soft computing, 13(7), 3259-3272. https://doi.org/10.1016/j.asoc.2013.02.013

[8] Gao, X. Z., Govindasamy, V., Xu, H., Wang, X., & Zenger, K. (2015). Harmony search method: theory and applications. Computational intelligence and neuroscience, 2015(1), 258491. https://doi.org/10.1155/2015/258491

[9] Qu, C., Fu, Y., Yi, Z., & Tan, J. (2018). Solutions to no-wait flow shop scheduling problem using the flower pollination algorithm based on the hormone modulation mechanism. Complexity, 2018 https://doi.org/10.1155/2018/1973604

[10] Wei, H., Li, S., Jiang, H., Hu, J., & Hu, J. (2018). Hybrid genetic simulated annealing algorithm for improved flow shop scheduling with makespan criterion. Applied Sciences, 8(12), 2621 https://doi.org/10.3390/app8122621

EARLY ACCESS