ISSN 2063-5346

COMPARATIVE STUDY OF CONVENTIONAL & NONCONVENTIONAL SCHEDULING RULES IN AN FLEXIBLE MANUFACTURING SYSTEMS ENVIRONMENT

Vivekanand S Gogi[1], Dr. C. K. Nagendra Gupta[2]

[1]Research Scholar, Dept. of Industrial Engineering & Management, RV College of Engineering, Bangalore, India

[2]Associate Professor & HoD, Dept. of Industrial Engineering & Management, RV College of Engineering, Bangalore, India

vivekanands@rvce.edu.in nagendragupta@rvce.edu.in

Article History: Received: 12.05.2023 **Revised:** 25.05.2023 **Accepted:** 05.06.2023

Abstract

FMS is a cutting-edge area for implementing manufacturing systems. The emergence of microcomputers and a wide range of high-speed microprocessors has provided organizations with an excellent opportunity to capitalize on these advancements, leading to enhanced performance, efficiency, and cost-effectiveness. FMS is regarded as the optimal approach to attain capabilities that are on par with industry leaders while maintaining high production volumes.

Given the rapid technological progress, achieving optimal performance becomes crucial. Efficient scheduling plays a central role in the functioning of an FMS, as it involves organizing and prioritizing job assignments to machines. The significance of effective scheduling cannot be emphasized enough, particularly in a context where delays result in substantial penalties.

The paper focuses on two different approaches to scheduling: traditional rules such as shortest processing time (SPT), earliest due date (EDD), and critical ratio (CR), and unconventional methods like genetic algorithms (GA) and a newer technique called Bacterial Foraging Optimization Algorithm (BFOA). The research compares the performance of these scheduling approaches using simulations and computations carried out in MATLAB, utilizing its built-in functions and tools. The results indicate that while EDD serves as a decent fundamental rule, the performance enhancements provided by GA and BFOA (especially the latter) greatly surpass those achieved by the more simplistic EDD approach.

Key Words: Flexible Manufacturing System (FMS), Scheduling Approaches, Optimization Algorithm.

Introduction

The investment required for establishing FMS is considerable, and any failure can have several implications for business continuity, resulting in significant reputational damage, missed opportunities, and substantial losses. Consequently, FMS remains a highly discussed topic in industry debates and a dynamic field of research with a wealth of knowledge available. Despite its extensive research, FMS still lacks comprehensive understanding due to a limited scope of thinking. The commonly employed methodologies include optimization models, multi-criteria decision making, heuristics-based approaches, simulation methods, and artificial intelligence techniques.

Historically, research efforts have primarily concentrated on long-term concerns, prioritizing planning problems while neglecting real-time control issues. However, there is a promosing inclination to employ replication not only for validation purposes but also for algorithm formulation. Additionally, advanced algorithms are being utilized as expert and knowledge systems to aid in problem-solving. The use of genetic algorithms (GA), knowledge-based systems (such as KBGA), petri-nets, and Distributed Problem Solving (DPS) is gaining momentum to address the gaps in heuristic research.

An increasing amount of research is being conducted to leverage artificial intelligence (AI) alongside other methodologies to enhance the practical applicability of results in industrial environments. The assessment of existing practices in the industry regarding FMS scheduling aims to propose an enhanced approach. Substantial effort will be dedicated to part-type selection and the algorithms associated with it, encompassing various techniques such as GA, KGBA, DPS, and other self-improving algorithms that learn from experience.

Literature Review: The following table presents the level of academic focus on different types of scheduling problems:

Table 1: Comparison of different technique

Sl	Authors	Type of scheduling problem					Type of AI technique used				
		1	2	3	4	5	1	2	3	4	5
1	Wu and Wysk (1988)	Yes	--	--	--	--	--	Yes	--	--	--
2	Wu and Wysk (1989)	Yes	--	--	--	--	--	Yes	--	--	--
3	Zimmerman et al. (1990)	Yes	Yes	--	--	--	Yes	--	--	--	--
4	Watanabe (1990)	Yes	--	--	--	--	Yes	--	--	--	--
5	Chandra et al. (1991a)	--	Yes	--	--	--	--	Yes	--	--	--
6	Nakasuka and Yoshida (1992)	Yes	--	--	--	--	--	--	--	--	Yes
7	Rao and O'Keefe (1992)	Yes	--	--	--	--	Yes	--	--	--	--
8	Kovacs et al. (1994)	--	--	--	--	Yes	--	Yes	--	--	--
9	Nagarur (1994)	Yes	--	--	Yes	--	--	--	Yes	--	--
10	Wang (1995)	Yes	--	--	--	Yes	Yes	--	--	Yes	--
11	Wen (1996)	Yes	--	--	--	--	--	--	--	--	Yes
12	Wu (1997)	--	--	Yes	--	--	--	Yes	--	--	--
13	Fang and Xi (1997)	Yes	--	--	--	Yes	Yes	--	--	--	--
14	Jawahar et al. (1998a)	Yes	--	--	--	--	--	--	--	Yes	--

15	Jawahar et al. (1998b)	--	Yes		--	--	--	--	--	--	Yes
16	Min et al. (1998)	Yes	--	--	--	Yes	--	--	--	Yes	--
17	Kim et al. (1998)	Yes	--	--	--	--	--	--	--	--	Yes
18	Chen et al. (1999)	Yes	--	--	--	Yes	Yes	--	--	--	--
19	Yu et al. (1999)	Yes	--	--	Yes	--	--	--	Yes	Yes	--
20	Qi et al. (2000)	Yes	--	--	--	--	--	--	--	--	Yes
21	O’Kane (2000)	Yes	--	--	--	Yes	--	Yes	--	--	--
22	Chandra and Patel (2001)	--	--	Yes	--	--	Yes	--	--	--	Yes
23	Krishna and Malley (2001)	Yes	--	--	--	--	--	--	--	Yes	--
24	Babu et al. (2001)	Yes	--	--	--	Yes	--	Yes	--	--	--
25	Fu and Li (2001)	Yes	--	--	Yes	--	--	--	--	--	Yes
26	Chen and Ram (2003)	Yes	--	--	Yes	--	--	--	--	Yes	--
27	Davis and Shore (2003)	Yes	--	--		Yes	Yes	--	--	--	Yes
28	Klabinsky et al. (2003)	Yes	--	--	Yes	--	--	Yes	--	--	--
29	Kumar and Dev (2003)	Yes	--	--	--	Yes	--	--	Yes	--	--
30	O’Reilly et al. (2004)	--	--	Yes	--	--	--	--	--	Yes	--
31	Summers (2004)	Yes	--	--	--	--	Yes	--	--	--	Yes
32	Sears and Denn (2005)	Yes	--	Yes	--	--	--	Yes	--	--	--
33	Fisher (2005)	--	Yes	Yes	--	--	--	--	Yes	Yes	--
34	Flannelly (2006)	Yes	--	--	--	--	--	--	--	--	--
35	Connor and D’Souza (2007a)	Yes	--	--	--	--	Yes	--	--	--	--
36	D’Souza (2008b)	Yes	--	--	Yes	--	--	Yes	--	--	--
37	Brandis and Riley (2009)	Yes	--	--	--	--	--	--	--	--	Yes
38	Riley et al. (2009)	--	Yes	--	--	--	--	--	--	Yes	--
39	Williams (2009)	--	--	Yes	--	Yes	Yes	--	Yes	Yes	--
40	Raine (2009)	Yes	--	--	--	--	--	--	--	--	--
41	Moore and Sharma (2010)	Yes	--	--	--	Yes	--	Yes	--	--	--
42	Prive (2011)	Yes	--	--	--	--	--	--	Yes	--	--
43	Swami and Krishna (2011)	Yes	--	--	--	Yes	Yes	--	--	Yes	--
44	Ernes and Burne (2011)	Yes	--	--	--	--	--	Yes	--	--	Yes
45	Li Chen et al. (2013)	Yes	--	--	--	Yes	--	--	Yes	--	Yes
46	Krishnan et al. (2012)	--	Yes	Yes	Yes	--	Yes	--	--	Yes	--
	TOTAL	25	6	7	8	13	13	12	7	12	12

The analysis indicates a notable inclination towards Part Dispatching (PD) problems, which researchers and academics consider to be broad and highly relevant. Additionally, there has been a recent surge in interest in AGV problems, with an increasing number of authors addressing the challenges posed by AGV systems. This can be attributed to the growing ease of use and maintenance of AGV systems and industries' willingness to adopt them in their operations. While PD problems have consistently garnered significant attention from researchers, there has been limited research of GA-based techniques for problem-solving. The analysis of 46 papers reveals that only 7 of them employed GA-based methodologies, while other approaches were more popular. Interestingly, GA-based techniques have been applied to solve PD problems in just 5 out of the 46 surveyed papers. This significant underrepresentation in the literature is one of the reasons why this paper places substantial emphasis on Genetic Algorithms and related algorithms.

Problem Description

The analysis indicates a notable inclination towards Part Dispatching (PD) problems, which researchers and academics consider to be broad and highly relevant. Additionally, a recent surge in interest in AGV problems, with an increasing number of authors addressing the challenges posed by AGV systems. This can be attributed to the growing ease of use and maintenance of AGV systems and industries' willingness to adopt them in their operations. While PD problems have consistently garnered significant attention from researchers, there has been limited study of GA-based techniques for problem-solving. The analysis of 46 papers reveals that only 7 of them employed GA-based methodologies, while other approaches were more popular. Interestingly, GA-based techniques have been applied to solve PD problems in just 5 out of the 46 surveyed papers. This significant underrepresentation in the literature is one of the reasons why this paper places substantial emphasis on Genetic Algorithms and related algorithms.

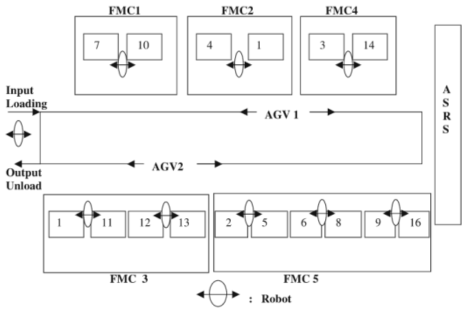

Fig. 1 Configuration of system considered

The problem environment and assumptions in this study, as stated by Jerald (2005), are outlined as follows:

i. There are approximately 40 to 50 different product varieties that correspond to specific combinations of tools stored in the tool magazines.

ii. A product variety has its processing sequence, batch size, deadline, and associated penalty cost if the deadline is not met.

iii. Each processing step is associated with a specific machine and has a predetermined processing time.

iv. The schedule's objective is to minimize both machine idle time and the total penalty cost.

v. The orders and types of parts to be produced are provided in Table 3. The table presents the machines (M1, M2, etc.) and the part types (P2, P2, P3, etc.). Each part requires a specific processing time (measured in hours) and arrives in batches of size T. Additionally, each part type incurs a penalty cost (PC) if the due date (DD) is exceeded.

Table 2: Order & Type of parts

	M 1	M 2	M 3	M 4	M 5	M 6	M 7	M 8	M 9	M 10	M 11	M 12	M 13	M 14	M 15	M 16	T	P. C	D. D
P1	0	0	0	0	0	1	1	1	0	2	0	0	0	0	0	0	150	1	17
P2	0	1	0	0	0	1	0	2	2	0	0	0	0	4	0	2	200	1	17
P3	0	0	0	0	0	0	0	1	0	0	3	0	4	0	0	0	800	1	14
P4	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	700	2	26
P5	0	0	0	5	3	0	0	0	0	0	0	0	0	0	4	0	150	1	11
P6	0	0	0	0	0	5	0	0	0	0	0	0	0	1	0	0	700	1	16
P7	0	0	5	0	0	3	0	0	0	0	0	0	0	0	0	5	250	2	26
P8	0	0	0	0	4	5	0	1	0	0	0	0	0	0	0	0	850	2	26
P9	0	0	0	1	5	0	0	1	0	0	1	0	0	0	0	0	100	0	1
P10	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	4	150	2	20
P11	0	0	0	0	0	0	0	4	0	0	0	2	0	0	0	0	250	1	1
P12	0	0	0	0	0	2	0	4	0	1	0	0	0	0	0	0	1000	3	19
P13	0	0	0	0	0	1	5	0	0	4	0	0	0	0	0	0	700	4	25
P14	0	0	0	2	3	2	0	0	0	0	0	0	0	0	2	0	1000	4	22
P15	0	0	0	0	4	0	0	3	0	0	0	0	0	0	0	0	700	5	15
P16	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	750	3	27
P17	0	0	1	0	0	4	0	0	0	0	0	0	0	1	0	0	650	4	20
P18	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	3	250	5	24
P19	0	0	0	1	5	2	0	2	0	0	0	0	0	0	5	0	450	1	5
P20	0	0	0	0	0	0	0	2	0	0	4	0	0	0	0	0	50	5	11
P21	0	0	0	5	5	0	0	4	0	0	0	0	0	0	4	0	850	3	16
P22	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	200	5	24
P23	0	0	0	2	1	5	0	4	0	0	0	0	0	0	0	0	50	4	14
P24	0	0	0	0	0	0	0	4	0	0	4	5	4	0	0	0	200	5	7
P25	0	0	0	0	0	0	3	0	0	2	0	0	0	0	0	0	350	1	24
P26	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	450	0	27
P27	0	0	0	0	0	0	0	5	0	0	5	4	0	0	0	0	400	1	22
P28	0	1	0	0	0	0	0	1	2	0	0	0	0	0	0	0	950	5	3
P29	0	0	0	1	5	0	0	0	0	0	0	0	0	0	0	0	700	1	7
P30	0	0	0	0	0	0	0	0	0	0	3	5	0	0	0	0	1000	1	18
P31	0	0	0	0	0	0	0	2	0	2	0	0	0	0	0	0	800	2	2
P32	0	3	0	0	0	4	0	0	3	0	0	0	0	0	0	0	800	1	15
P33	0	0	0	0	4	5	0	0	0	0	0	0	0	0	3	0	500	4	27
P34	0	0	2	0	0	2	0	0	0	0	0	0	0	0	0	0	300	4	12
P35	0	0	4	0	0	0	0	0	0	0	0	0	0	1	0	0	900	2	9
P36	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	700	2	20
P37	5	2	0	0	0	3	0	3	2	0	0	0	0	0	0	4	250	4	22
P38	0	4	0	0	0	0	0	3	2	0	0	0	0	0	5	0	50	1	8
P39	0	0	0	0	0	5	0	0	0	5	0	0	0	0	0	0	500	1	9

P40	**0**	**2**	**0**	**0**	**0**	**4**	**0**	**0**	**4**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**250**	**5**	**7**
P41	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**800**	**4**	**22**
P42	**0**	**5**	**0**	**0**	**0**	**4**	**0**	**0**	**3**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**400**	**2**	**19**
P43	**3**	**0**	**0**	**0**	**2**	**2**	**0**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**3**	**0**	**550**	**3**	**15**

Objectives

1. To prepare a schedule for the system using various techniques.
2. To identify the best technique among those considered.

Methodology

This paper involves a comparative analysis of different scheduling rules and techniques commonly employed in the field. The study assesses the performance of simple rules such as shortest processing time (SPT), earliest due date (EDD), and critical ratio (CR) as scheduling parameters. It compares their conclusiveness with more sophisticated approaches, specifically utilizing a basic genetic algorithm (GA) and bacterial foraging optimization algorithm (BFOA), for scheduling a given part-machine setup provided as input. MATLAB is utilized, employing a GUI-based program to visualize and analyze the obtained results. The performance evaluation focuses on measures such as idleness and penalties incurred due to order non-fulfillment, with the intention being to minimize a Combined Objective Function (COF).

Scheduling objective

The primary goal of the schedule is to achieve a balance between minimizing the idle time of the machines and reducing the overall penalty cost. In essence, the objective is to minimize a weighted objective function, which combines the weighted values of the total penalty cost and the total idle time of the machines.

$$COF = w_1 * \frac{\text{Total penalty cost}}{\text{Maximum permissible penalty}} w_1 * \frac{\text{Total penalty cost}}{\text{Maximum permissible penalty}} + w_2 * \frac{\text{Total machine idle time}}{\text{Maximum total machine elapsed time}} w_2 * \frac{\text{Total machine idle time}}{\text{Maximum total machine elapsed time}}$$

$$\text{Total penalty cost} = \sum (CT_i - DD_i) * UPC_i * BS_i$$

The allocations of weights to each objective function are denoted as w_1 and w_2. In the formulas, CT_i represents the completion time of job i, DDi represents the due date of job i, UPCi represents the unit penalty cost of job i, and BSi represents the batch size of job i.

A deliberate choice was made to prioritize methodologies that offer practical applications and the potential for expanding current knowledge. As a result, the emphasis of this paper will be on two primary categories of scheduling rules: traditional scheduling rules such as Shortest Processing Time, Earliest Due Date, and Critical Ratio, and non-conventional scheduling rules including Genetic Algorithm and Bacterial Foraging Optimization Algorithm.

Results & Discussion

The outcomes achieved by the different conventional rules are presented in Figure 15. Additionally, Table 4 displays the techniques employed along with their corresponding penalty, combined objective function (COF), and idleness values.

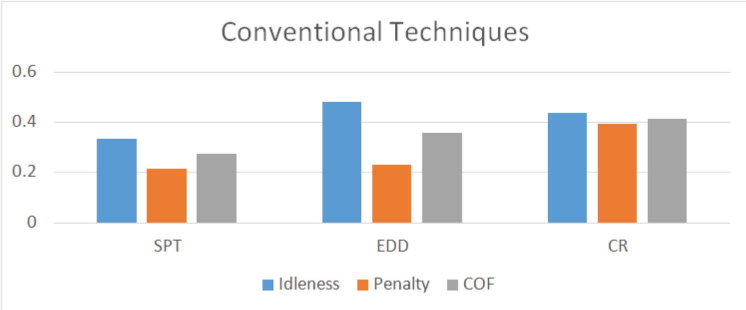

Fig. 2 Comparison of results from conventional techniques

Table 3: Results from conventional rules

Technique	Sequence	Idleness	Penalty	COF
SPT	20 23 38 1 9 26 22 10 34 18 36 11 25 5 16 2 40 4 31 41 7	0.2453	0.2138	0.2296
	24 28 17 6 29 35 37 15 39 42 27 33 3 43 19 12 13 30 32			
	8 14 21			
EDD	3 4 17 43 1 2 41 15 12 7 39 21 5 25 31 34 28 10 6 23 32	0.4826	0.2325	0.3576
	16 27 8 9 24 19 22 13 14 30 35 37 40 18 38 11 29 36 33			
	20 42 26			
CR	20 26 22 23 1 18 10 36 25 16 38 34 4 7 2 41 5 17 37 33	0.4362	0.3952	0.4157
	27 6 42 13 15 8 40 12 14 43 30 24 35 32 39 29 9 21 28			
	19 11 31			

To summarize, the results reveal that the Shortest Processing Time (SPT) rule outperforms the other two conventional rules considered in this study. The idleness achieved with SPT is approximately half that of the Earliest Due Date (EDD) rule. It is also worth noting that the difference in penalty between SPT and EDD is minimal. Therefore, if strict adherence to due dates is of utmost importance, the EDD rule may be more suitable. However, it is important to note that the Critical Ratio (CR) rule performs poorly, as it exhibits higher values for two out of three performance measures. Therefore, based on the conditions examined in this paper, it is not advisable to consider CR as a favorable scheduling rule.

For the widely discussed genetic algorithm (GA) and bacterial foraging optimization algorithm (BFOA). The results generated by these algorithms are showcased in Table 5, where the performance change is demonstrated by comparing them with the results obtained using the Shortest Processing Time (SPT) rule from the conventional rules.

Table 4: Results from GA and BFOA for various data sets

Method	Operations	Sequence	Idleness	Penalty	COF
GA	100	33 9 15 25 39 19 17 32 30 27 28 43 13 21 29 3 6 16 22 24 23 41 5 37 4 36 10 12 34 44 14 40 26 7 20 8 35 11 31 18 38 42 1 2	0.3590	0.2235	0.2913
	112	23 12 39 24 32 3 40 35 22 30 9 41 6 26 29 43 16 34 7 37 36 4 1 31 11 2 15 33 14 25 21 10 38 8 17 13 5 20 27 28 19 18 44 42	0.3481	0.2359	0.2920
	124	6 34 35 39 11 24 9 15 29 5 12 23 28 36 43 1 18 26 3 17 21 20 31 22	0.3342	0.2488	0.2915

		7 13 25 37 41 30 2 40 27 14 32 8 10 33 4 15 38 19 42			
	136	5 11 42 18 33 29 26 37 2 35 19 21 14 27 16 12 34 20 39 25 13 24 3 4 32 43 38 10 41 44 15 30 1 17 31 7 36 6 8 9 22 40 28 23	0.3219	0.2635	0.2927
	148	6 22 32 37 25 39 19 21 2 34 9 40 27 43 41 7 15 28 16 33 5 14 35 3 12 23 30 38 36 13 24 10 26 4 31 1 18 17 44 8 11 29 20 42	0.3142	0.2703	0.2923
BFOA	100	10 26 3 27 8 23 41 17 15 32 2 39 16 20 13 29 9 21 18 34 37 5 28 6 30 33 7 25 40 36 31 24 44 38 1 22 43 4 11 35 14 12 42 19	0.1453	0.1624	0.1539
	112	10 1 18 13 8 33 2 31 27 16 28 9 15 37 21 22 44 7 42 41 5 4 34 25 23 43 32 3 36 20 14 39 35 30 17 12 19 24 40 38 6 11 29 26	0.1296	0.1734	0.1515
	124	19 41 36 5 38 3 43 2 15 31 8 37 1 10 33 11 29 18 39 7 40 13 35 12 27 42 30 23 6 22 17 21 34 25 4 24 20 14 32 28 9 16 26	0.1184	0.1977	0.1581
	136	37 4 27 23 16 42 11 15 5 31 32 1 14 33 30 19 40 25 24 38 13 10 20 28 29 3 2 36 18 43 21 22 41 17 7 44 12 39 6 34 9 26 35 8	0.1013	0.2087	0.1550
	148	17 9 27 39 29 22 13 7 28 1 30 24 20 8 40 35 16 5 18 15 32 10 36 21 34 6 38 12 3 31 23 42 14 2 41 26 19 4 11 44 43 25 37 33	0.0821	0.2193	0.1507

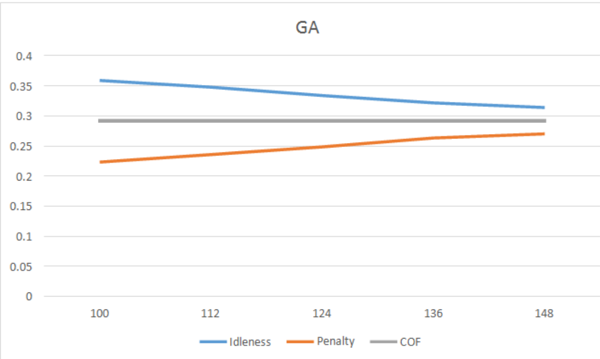

Fig 3 Variation of results for GA

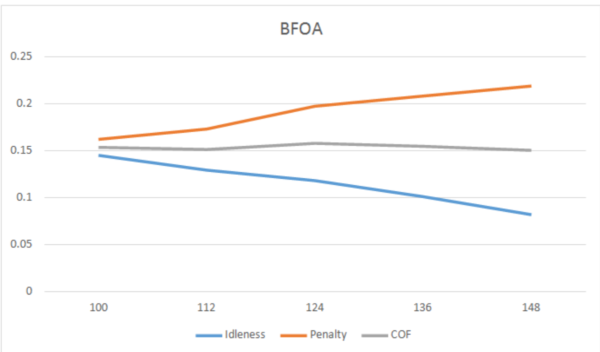

Fig 4 *Variation of results for BFOA*

In this analysis, it is observed that as the different operations handled by the system increases, the value of the combined objective function (COF) does not exhibit significant changes. However, the corresponding idleness and penalty values do fluctuate. This is attributed to the algorithm stabilizing the COF value around its final values after approximately 80 generations of offspring, which aligns with the known property of the bacterial foraging optimization algorithm (BFOA) causing rapid convergence. Consequently, a careful opinion has been made in this study to consider 124 operations, as specified in Table 3 as part of the methodology.

The outcomes presented in Table 6 demonstrate that the basic genetic algorithm does not provide any enhancement compared to the simpler Shortest Processing Time (SPT) rule. Across all the performance measures utilized in this study, it is evident that the basic genetic algorithm results in higher penalty and idleness when compared to SPT. Hence, it is concluded that the basic genetic algorithm does not offer improvements in terms of these performance measures compared to SPT.

Table 5: Compiled results

Technique	Sequence	Penalty	Idleness	COF
SPT	20 23 38 1 9 26 22 10 34 18 36 11 25 5 16 2 40 4 31 41 7 24 28 17 6 29 35 37 15 39 42 27 33 3 43 19 12 13 30 32 8 14 21	0.2453	0.2138	0.2296
EDD	3 4 17 43 1 2 41 15 12 7 39 21 5 25 31 34 28 10 6 23 32 16 27 8 9 24 19 22 13 14 30 35 37 40 18 38 11 29 36 33 20 42 26	0.4826	0.2325	0.3576
CR	20 26 22 23 1 18 10 36 25 16 38 34 4 7 2 41 5 17 37 33 27 6 42 13 15 8 40 12 14 43 30 24 35 32 39 29 9 21 28 19 11 31	0.4362	0.3952	0.4157
GA	21 5 32 19 4 20 42 35 23 11 6 22 41 1 12 25 33 24 43 2 13 26 17 28 31 14 3 29 7 30 9 36 18 38 10 37 16 40 8 34 15 27 39	0.2995	0.2548	0.2772
BFOA	9 20 42 3 5 8 10 14 18 19 23 28 31 33 35 36 40 43 6 15 17 24 27 32 38 1 4 7 11 16 25 34 37 41 2 12 22 26 29 39 13 21 30	0.1034	0.2019	0.1527

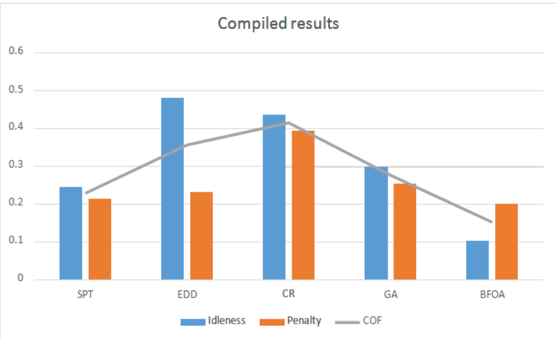

Fig 5: Compiled results of all techniques

In contrast, the application of Bacterial Foraging Optimization Algorithm (BFOA) yields substantial changes in the results. The penalty value, which was 0.2453 with SPT, is significantly reduced to 0.1034, representing a 43% decrease. Likewise, the idleness is reduced by approximately 5% to 20.1% of the original duration. Furthermore, the combined objective function is reduced by 33% to a value of 0.1527. These findings highlight the significant improvements achieved by employing BFOA, resulting in reduced penalty, idleness, and overall objective function values compared to the SPT rule.

Conclusion

Significant advancements in Flexible Manufacturing Systems (FMS) have been witnessed in recent years, largely attributed to the integration of ideas and innovations from various scientific and technological fields. The findings show that Bacterial Foraging Optimization Algorithm (BFOA) may soon join the list of influential advancements in FMS. While Genetic Algorithm (GA) falls short of expectations, the reliability of Shortest Processing Time (SPT) is reaffirmed as the preferred method for simpler systems. BFOA emerges as the focal point of this study, showcasing substantial improvements over traditional methods by reducing the combined objective function (COF) by one-third and more than halving the incurred penalty. While these results show promise, the demanding nature of BFOA poses a significant challenge, as it takes a considerable amount of time to complete a single run, even on high-end computers.

Consequently, BFOA's usability is limited, making it unsuitable for real-time scheduling. However, within the conditions and assumptions of the model used in this study, BFOA proves to be the most effective technique for part type scheduling, surpassing other methods by a considerable margin.

References:

1. Babu, A., 2001, A Heuristics Based Framework for machine interactions, International Journal of Production Management, 39, 29-33
2. Basnet, C.B. and Mize, J.H., 1992, An object-oriented framework for operating flexible manufacturing systems. Proceedings: International Conference on Object-Oriented Manufacturing Systems, Calgary, Canada, 346-351.
3. Basnet, C.B. and Mize, J.H., 1992, Scheduling and Control of FMS: A Critical Review, Advanced Manufacturing, 5, 1442-1474
4. Brandis, D., 2009, A comparison of advanced FMS scheduling Techniques, Advanced Manufacturing, 23, 233-245
5. Chan, F.T. and Chan, H.K., 2002, A comprehensive survey and future trend simulation study on FMS scheduling, Journal of Intelligent Manufacturing, 15, 87-102
6. Chandra, S., 1991, A Heuristic-Based Machine Selection Model, Indian Journal of Manufacturing Technology, 33, 78-84
7. Chandra R. and Patel, H., 2001, A Fuzzy Framework for studying AGV interactions, Indian Journal of Production Sciences, 9, 122-125
8. Chen, F. and Ram, M., 2003, A Comparison of Neural Network Systems, Indian Journal of Operations Management, 22, 211-215
9. Connor, A. and D’Souza, P., 2007, A Simplified fuzzy algorithm for manufacturing control, Proceedings: International conference on sustainable manufacturing, Lisbon, Portugal, 38-42
10. Davis, S. and Shore, P., 2003, Optimizing for multiple objectives with Fuzzy-Petri Net Hybrids, Proceedings of Conference of British Operations Researchers, Cork, 222-225
11. D’Souza, P., 2008, A GUI-based multi-objective heuristic function for optimizing FMS scheduling, International Journal of Advanced Manufacturing, 34-39
12. Fang, A., Xi, P., 1997, An Advanced Fuzzy OS Technique, Xinhua Journal of Industrial Research, 18, 33-38
13. Fisher, D., 2005, A Genetic-neural hybrid optimization technique for shop floor use, American Journal of Manufacturing Engineering, 48, 34-38
14. Flanelly, M., 2006, A survey of part dispatching techniques, International Journal of Current Engineering and Technology, 3, 5-7
15. Fu, L. and Li, C., 2001, Multi-Level Scheduling Algorithm using Swarm Optimization, Taiwan Production Sciences Journal, 2, 67-72