Faculty of Manufacturing Engineering

HYBRID-DISCRETE MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION FOR MULTI-OBJECTIVE JOB-SHOP SCHEDULING

Nurul Izah binti Anuar

Doctor of Philosophy

2022

HYBRID-DISCRETE MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION FOR MULTI-OBJECTIVE JOB-SHOP SCHEDULING

NURUL IZAH BINTI ANUAR

A thesis submitted
in fulfillment of the requirements for the degree of Doctor of Philosophy

Faculty of Manufacturing Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this thesis entitled "Hybrid-Discrete Multi-Objective Particle Swarm Optimization for Multi-Objective Job-Shop Scheduling" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : Nurul Izah binti Anuar

Date : 7 September 2022

APPROVAL

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of Doctor of Philosophy.

Signature : ....

Supervisor Name : Assoc. Prof. Dr. Muhammad Hafidz Fazli bin Md Fauadi

Date : 7 September 2022

DEDICATION

To my beloved family

ABSTRACT

Many real-world production scheduling problems involve the simultaneous optimization of multiple conflicting objectives that are challenging to solve without the aid of powerful optimization techniques. This includes the multi-objective Job-shop Scheduling Problem (JSP), which is among the most difficult to solve owing to the existence of an intractably large, highly complex solution space. Particle Swarm Optimization (PSO) is a population-based metaheuristic that possesses many advantages compared to other metaheuristics in solving scheduling problems. However, due to the complex nature of the multi-objective JSP, a single approach like PSO is not sufficient to explore the search space effectively owing to its shortcoming such as the tendency to become trapped in local optima. Besides, since PSO operates in the continuous domain, it cannot be applied directly to solve a discrete problem like the JSP efficiently. This research first proposes an improved continuous MOPSO to address the rapid clustering problem that exists in the basic PSO algorithm using three improvement strategies: re-initialization of particles, systematic switch of best solutions and mutation on global best selection. In order to establish an efficient mapping between the particle's position in the continuous MOPSO and the scheduling solution in the JSP, this research proposes the JSP to be adopted within a discrete MOPSO through a modified solution representation using the permutation-based representation and a modified setup of the particle's position and velocity. The discrete MOPSO also includes the modified maximin fitness function to promote solution diversity in the selection of global best solutions. In order to accomplish better performance by improving the search quality and efficiency of the discrete MOPSO, this research proposes a hybrid with the Diversification Generation Method in Scatter Search, the non-dominated sorting mechanism in non-dominated sorting Genetic Algorithm II (NSGA-II) and the local search mechanism in Tabu Search. The experimentations of the proposed algorithm are conducted using existing benchmark instances and a published case study on an energy-efficient job-shop model. The computational results are evaluated against other optimization techniques published in the literature. From the results, it is found that the proposed improved algorithm is effective in solving the benchmark instances compared to when no improvement is implemented and with a reasonable increase in computational costs. It is also discovered that the hybrid-discrete MOPSO (HD-MOPSO) algorithm manages to obtain higher values in the performance metrics consisting of non-dominance ratio and hypervolume compared to the competing algorithms. For the non-dominance ratio, HD-MOPSO is able to contribute 89% to 100% of solutions to the reference Pareto front. For the hypervolume values, HD-MOPSO manages to obtain a minimum of 1.0172 to 1.2862 out of the optimum value of 1.44. As higher values of metrics indicate better performance, HD-MOPSO thus outperforms the competing algorithms in solving the benchmark instances and the published case study. For these types of problems, the proposed algorithm is demonstrated to be capable of producing higher percentages of solutions in the overall non-dominated set with better quality in terms of convergence and diversity than those obtained by the competing algorithms.

PENGOPTIMUMAN KAWANAN ZARAH BERBILANG OBJEKTIF HIBRID-DISKRIT UNTUK PENJADUALAN BENGKEL KERJA BERBILANG OBJEKTIF

ABSTRAK

Banyak masalah penjadualan pengeluaran dunia sebenar melibatkan pengoptimuman serentak berbilang objektif yang bercanggah dan mencabar untuk diselesaikan tanpa bantuan teknik pengoptimuman yang berkuasa. Ini termasuk Masalah Penjadualan Bengkel Kerja (MPBK) berbilang objektif, yang merupakan antara yang paling sukar untuk diselesaikan kerana kewujudan ruang penyelesaian yang sangat besar dan sangat kompleks. Pengoptimuman Kawanan Zarah (PKZ) ialah metaheuristik berasaskan populasi yang mempunyai banyak kelebihan berbanding metaheuristik lain dalam menyelesaikan masalah penjadualan. Walau bagaimanapun, disebabkan sifat kompleks MPBK berbilang objektif, pendekatan tunggal seperti PKZ tidak mencukupi untuk meneroka ruang carian dengan berkesan kerana kelemahannya seperti kecenderungan untuk terperangkap dalam optima tempatan. Selain itu, oleh kerana PKZ beroperasi dalam domain berterusan, ia tidak boleh digunakan secara langsung untuk menyelesaikan masalah diskret seperti MPBK dengan cekap. Penyelidikan ini pertamanya mencadangkan PKZ berbilang objektif (PKZBO) Berterusan dipertingkatkan untuk menangani masalah pengelompokan pantas yang wujud dalam algoritma PKZ asas melalui tiga strategi penambahbaikan: pembentukan semula zarah, pertukaran penyelesaian terbaik secara sistematik dan mutasi pada pemilihan terbaik global. Untuk mewujudkan pemetaan yang efisien antara posisi zarah dalam PKZBO Berterusan dan penyelesaian penjadualan dalam MPBK, penyelidikan ini mencadangkan MPBK diguna pakai dalam PKZBO Diskret melalui pengubahsuaian representasi penyelesaian menggunakan representasi berasaskan permutasi serta pengubahsuaian tetapan posisi dan halaju zarah. PKZBO Diskret juga mengandungi fungsi kecergasan maximin yang diubah suai untuk menambah kepelbagaian penyelesaian dalam pemilihan penyelesaian terbaik global. Untuk mencapai prestasi yang lebih baik dengan meningkatkan kualiti dan kecekapan carian PKZBO Diskret, penyelidikan ini juga mencadangkan hibrid dengan Kaedah Penjanaan Kepelbagaian dalam Carian Taburan, mekanisme penyusunan tidak didominasi dalam Algoritma Genetik Penyusunan Tidak Didominasi II dan mekanisme carian tempatan dalam Carian Tabu. Ujikaji algoritma dilakukan menggunakan masalah penanda aras sedia ada dan kajian kes yang telah diterbitkan mengenai model cekap tenaga bengkel kerja. Hasil pengiraan telah dinilai berbanding teknik pengoptimuman sedia ada. Dari keputusan tersebut, didapati algoritma yang dicadangkan berkesan menyelesaikan masalah penanda aras berbanding apabila tiada penambahbaikan dengan peningkatan munasabah dalam kos pengiraan. Ia juga mendapati algoritma PKZBO hibrid-diskrit (PKZBO-HD) berjaya memperoleh nilai lebih tinggi dalam ukuran prestasi yang terdiri daripada nisbah bukan dominasi dan hipervolume berbanding algoritma pesaing. Untuk nisbah bukan dominasi, PKZBO-HD menyumbang 89% hingga 100% penyelesaian kepada rujukan sempadan Pareto. Untuk nilai hipervolume, PKZBO-HD memperoleh minimum 1.0172 hingga 1.2862 daripada nilai optimum 1.44. Oleh kerana nilai ukuran yang lebih tinggi menunjukkan prestasi lebih baik, PKZBO-HD mengatasi prestasi algoritma pesaing dalam menyelesaikan masalah penanda aras dan kajian kes. Untuk jenis masalah ini, algoritma yang dicadangkan mampu menghasilkan peratusan penyelesaian yang lebih tinggi dalam set keseluruhan tidak didominasi dengan kualiti yang lebih baik dari segi penumpuan dan kepelbagaian berbanding yang diperoleh algoritma pesaing.

ACKNOWLEDGEMENTS

My deepest gratitude to Allah, the God Almighty for his blessings that allows and enables me to complete this research work and prepare this thesis successfully.

I wish to express my fullest appreciation and most sincere thanks to my supervisor, Associate Professor Dr. Muhammad Hafidz Fazli bin Md Fauadi, for his patience, unwavering support, continuous encouragement, invaluable guidance and excellent advice throughout my studies. His insightful comments and ideas have greatly improved the work until it turns into real success.

I would also like to express my deepest gratitude to my previous supervisor, Professor Dr. Adi Saptari, for his patience, never-ending support and ideas, especially for his confidence in me. My sincere appreciation also extends to Universiti Teknikal Malaysia Melaka for providing this opportunity to do my PhD studies and my employer Multimedia University Malaysia for the study leave granted during the final year of my studies.

Last but not least, I am deeply in debt of gratitude to my family, especially my children; I have sacrificed the time, attention and care towards them to pursue my dream. For their love, trust, patience and understanding, I dedicate this achievement to them.

TABLE OF CONTENTS

			PAGE
DECLARATION			
DEDICATION			
ABSTRACT			**i**
ABSTRAK			**ii**
ACKNOWLEDGEMENTS			**iii**
TABLE OF CONTENTS			**iv**
LIST OF TABLES			**vii**
LIST OF FIGURES			**ix**
LIST OF APPENDICES			**xii**
LIST OF PUBLICATIONS			**xiii**
CHAPTER			
1.	**INTRODUCTION**		**1**
	1.1	Background	1
	1.2	Problem Statement	5
		1.2.1 Energy-Efficient Scheduling	8
	1.3	Research Gap	10
	1.4	Research Objectives	12
	1.5	Research Contributions	12
	1.6	Research Scope	15
	1.7	Organization of the Thesis	16
2.	**LITERATURE REVIEW**		**17**
	2.1	Production Scheduling	17
		2.1.1 Classes of Scheduling Problems	20
		2.1.2 Job-shop Scheduling Problem (JSP)	22
		2.1.3 Objective Functions	26
	2.2	Energy-efficient Scheduling	29
		2.2.1 Electricity Saving Methods	31
		2.2.2 Related Works on Energy-efficient JSP	32
	2.3	Multi-objective Optimization	37
	2.4	Particle Swarm Optimization (PSO)	40
		2.4.1 PSO for Discrete Optimization Problems	42
		2.4.2 PSO for Multi-objective Optimization	46
	2.5	Related Works of MOPSO in solving JSP	49
	2.6	Other Metaheuristic Approaches in solving JSP	56
		2.6.1 Scatter Search	57
		2.6.2 Non-dominated Sorting Genetic Algorithm II (NSGA-II)	59
		2.6.3 Tabu Search	61
	2.7	MOPSO Improvement Strategies	64
		2.7.1 Improvements in diversity of swarm solutions	69
		2.7.2 Improvements in exploitation/exploration mechanisms	70
		2.7.3 Improvements in premature convergence	75
	2.8	MOPSO model for solving JSP	76
	2.9	Discussion	78
	2.10	Summary	81

3. RESEARCH METHODOLOGY **82**
3.1 Flow of Methodology and General Framework 82
3.2 Continuous MOPSO 88
3.2.1 Solution Representation Procedure 89
3.2.2 Pareto-dominance Strategy for Two Objectives 90
3.2.3 Maintenance and Diversification of Repositories 92
3.3 Improved Continuous MOPSO 95
3.3.1 Re-initialization of Particles 97
3.3.2 Systematic Switch of Best Solutions 98
3.3.3 Mutation on Global Best Selection 105
3.4 Improved Discrete MOPSO 109
3.4.1 Modified Solution Representation Procedure 111
3.4.2 Modified Setup of Position and Velocity 114
3.4.3 Global Best Selection using Modified Maximin Fitness Function 118
3.5 Hybrid-Discrete MOPSO 122
3.5.1 Particle Initialization using DGM in Scatter Search 124
3.5.2 Personal Best Selection using Non-dominated Sorting Mechanism in NSGA-II 126
3.5.3 Local Search Strategy using Tabu Search mechanism 128
3.6 Experimental Setup 131
3.6.1 Job-shop Models 131
3.6.1.1 Conventional JSP 133
3.6.1.2 Energy-Efficient JSP 134
3.6.2 Experimental Design 137
3.7 Performance Metrics 141
3.8 Summary 143

4. RESULTS AND DISCUSSIONS **145**
4.1 Parameter Settings 145
4.2 Experimentations on Improved Continuous MOPSO 146
4.2.1 1st Experiment 147
4.2.1.1 Performance Statistics 147
4.2.1.2 Percentage of Improvements 150
4.2.1.3 Pareto Fronts 150
4.2.1.4 Non-dominance Ratio 153
4.2.1.5 Hypervolume 153
4.2.1.6 Average Computational Times 154
4.2.1.7 Concluding Remarks 155
4.2.2 2nd Experiment 156
4.2.2.1 Performance Statistics 156
4.2.2.2 Percentage of Improvements 158
4.2.2.3 Pareto Fronts 159
4.2.2.4 Non-dominance Ratio 161
4.2.2.5 Hypervolume 162
4.2.2.6 Concluding Remarks 164
4.3 Experimentations on Improved Discrete MOPSO 165
4.3.1 Performance Statistics 166
4.3.2 Percentage of Improvements 167
4.3.3 Pareto Fronts 168
4.3.4 Non-dominance Ratio 170

4.3.5 Hypervolume 171
4.3.6 Average Computational Times 172
4.3.7 Concluding Remarks 173
4.4 Experimentations on Hybrid-Discrete MOPSO 175
4.4.1 Performance Statistics 176
4.4.2 Percentage of Improvements 177
4.4.3 Pareto Fronts 178
4.4.4 Non-dominance Ratio 180
4.4.5 Hypervolume 181
4.4.6 Average Computational Times 181
4.4.7 Concluding Remarks 182
4.5 Case Study on Energy-Efficient JSP 184
4.5.1 Performance Statistics 184
4.5.2 Percentage of Improvements 186
4.5.3 Pareto Fronts 186
4.5.4 Non-dominance Ratio 189
4.5.5 Hypervolume 190
4.5.6 Concluding Remarks 191
4.6 Conclusion 191
4.7 Summary 194

5. CONCLUSION AND RECOMMENDATIONS 195
5.1 Research Summary 195
5.2 Research Conclusions 197
5.3 Limitations of Study and Recommendations for Future Research 201
5.4 Summary 202

REFERENCES 204
APPENDICES 228

LIST OF TABLES

TABLE	TITLE	PAGE
2.1	Example of JSP with 3 jobs and 3 machines (Yamada, 2003)	24
2.2	Studies related to energy-efficient JSP	35
2.3	Studies related to MOPSO in solving JSP	52
2.4	List of MOPSO improvements applied in JSP	66
3.1	Example of position and velocity initialization for discrete MOPSO	116
3.2	Due dates and weights of jobs for modified FT10 instance (Liu et al., 2014)	136
3.3	Idle power of machines for modified FT10 instance (in kW) (Liu et al., 2014)	136
3.4	Experimental design for each stage of proposed algorithm	138
3.5	Test instances and objectives implemented in each stage of proposed algorithm: (a) Improved Continuous MOPSO, (b) Improved Discrete MOPSO and (c) HD-MOPSO	140
4.1	Results of improved continuous MOPSO in comparison with basic MOPSO	148
4.2	Percentage of improvements in objective values achieved by improved MOPSO in comparison with basic MOPSO	150
4.3	Results of improved MOPSO in comparison with SPEA and CMOEA	157
4.4	Percentage of improvements in objective values achieved by improved MOPSO in comparison with SPEA and CMOEA	158
4.5	Results of improved discrete MOPSO in comparison with competing algorithms	167
4.6	Percentage of improvements in objective values achieved by discrete MOPSO in comparison with competing algorithms	168
4.7	Results of HD-MOPSO in comparison with competing algorithms	176
4.8	Percentage of improvements in objective values achieved by HD-MOPSO in comparison with competing algorithms	177
4.9	Results of HD-MOPSO in comparison with NSGA-II	185

4.10 Percentage of improvements in objective values achieved by HD-MOPSO in comparison with NSGA-II 186
4.11 Compilation of computational results 193

LIST OF FIGURES

FIGURE	TITLE	PAGE
2.1	Interaction of scheduling with other functions in a generic manufacturing environment (Pinedo, 2016)	19
2.2	A disjunctive graph representation of the 3 × 3 problem (Yamada, 2003)	25
2.3	A Gantt chart representation of a scheduling solution for the 3 × 3 problem (Yamada, 2003)	26
2.4	The simplified power input of a machine tool when working on one operation (Liu et al., 2014)	30
2.5	Basic procedure of PSO algorithm	41
2.6	Priority-based representation where each particle represents the priority values of activities (Zhang et al., 2005)	44
2.7	Permutation-based representation where each particle represents the permutation or sequence of activities (Zhang et al., 2005)	45
2.8	Basic procedure of MOPSO algorithm	48
2.9	Basic procedure of Scatter Search	58
2.10	Basic procedure of NSGA-II	60
2.11	Basic procedure of Tabu Search	63
2.12	Example of dominance relationships among four fitness values at time step *t*+1 (Li, 2003)	75
2.13	Proposed MOPSO model to solve JSP	77
3.1	Flow of methodology	84
3.2	Proposed improvement strategies in MOPSO for JSP	85
3.3	Stages of improvements in MOPSO with corresponding research objectives	86
3.4	General framework of the study	88
3.5	Example of random key representation with SPV rule	90
3.6	Example of Pareto-dominance strategy for two objectives	91

3.7 Example of update of global repository 93
3.8 Procedure of the improved continuous MOPSO algorithm 96
3.9 Flowchart of re-initialization of particles during initialization phase 97
3.10 Flowchart of re-initialization of particles during current fitness evaluation phase 98
3.11 Flowchart of systematic switch of best solutions 101
3.12 Flowchart of random selection procedure 102
3.13 Flowchart of preserved selection procedure during (a) personal best selection, and (b) global best selection 104
3.14 Flowchart of updated preserved selection procedure with Tabu Search-based mutation 108
3.15 Procedure of the improved discrete MOPSO algorithm 110
3.16 Example of solution representation procedure for (a) random key representation, and (b) permutation-based representation 113
3.17 Example of velocity normalization for discrete MOPSO 117
3.18 Example of position update for discrete MOPSO 118
3.19 Flowchart of updated preserved selection procedure with modified maximin fitness function 121
3.20 Procedure of the HD-MOPSO algorithm 123
3.21 Example of DGM for particle initialization 125
3.22 Example of ten particles' fitness classified into four successive fronts (Li, 2003) 128
3.23 Pseudo-code of local search 130
3.24 Illustration of the hypervolume 143
4.1 Pareto fronts of basic and improved continuous MOPSO for (a) FT06, (b) FT10 and (c) FT20 instances 152
4.2 Non-dominance ratio of basic and improved continuous MOPSO 153
4.3 Hypervolume of basic and improved continuous MOPSO 154
4.4 Average computational times per run (in seconds) of basic and improved continuous MOPSO 155
4.5 Pareto fronts of SPEA, CMOEA and improved MOPSO for (a) FT06, (b) FT10 and (c) FT20 instances 160
4.6 Non-dominance ratio of SPEA, CMOEA and improved continuous MOPSO 162
4.7 Hypervolume of SPEA, CMOEA and improved continuous MOPSO 163
4.8 Pareto fronts of SPEA, CMOEA, improved continuous and discrete MOPSO for (a) FT10 and (b) FT20 instances 169

4.9 Non-dominance ratio of SPEA, CMOEA, improved continuous and discrete MOPSO 171
4.10 Hypervolume of SPEA, CMOEA, improved continuous and discrete MOPSO 172
4.11 Average computational times per run (in seconds) of improved continuous and discrete MOPSO 173
4.12 Pareto fronts of SPEA, CMOEA, improved discrete MOPSO and HD-MOPSO for (a) FT10 and (b) FT20 instances 179
4.13 Non-dominance ratio of SPEA, CMOEA, improved discrete MOPSO and HD-MOPSO 180
4.14 Hypervolume of SPEA, CMOEA, improved discrete MOPSO and HD-MOPSO 181
4.15 Average computational times per run (in seconds) of improved discrete MOPSO and HD-MOPSO 182
4.16 Pareto fronts of NSGA-II and HD-MOPSO for the case study problem 188
4.17 Non-dominance ratio of NSGA-II and HD-MOPSO 190
4.18 Hypervolume of NSGA-II and HD-MOPSO 190
4.19 Average computational times of each algorithm 194

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	JSP benchmark instances	228
B	Computational Results	230

LIST OF PUBLICATIONS

Anuar, N.I., and Md Fauadi, M.H.F., 2021. A Study on Multi-Objective Particle Swarm Optimization in Solving Job-Shop Scheduling Problems. *International Journal of Computer Information Systems and Industrial Management Applications*, Vol. 13, pp. 51-61.

Anuar, N.I., Md Fauadi, M.H.F., Saptari, A., and Hao, X., 2020. Improved Multi-Objective Particle Swarm Optimization for Job-Shop Scheduling Problems. *Journal of Advanced Manufacturing Technology (JAMT)*, 14(3).

Anuar, N.I., Md Fauadi, M.H.F., and Saptari, A., 2019. Performance Evaluation of Continuous and Discrete Particle Swarm Optimization in Job-Shop Scheduling Problems. In: *Proceedings of the International Conference on Recent Advances in Industrial Engineering and Manufacturing*, Penang, Malaysia, 12–13 December 2018. IOP Conference Series: Materials Science and Engineering (Vol. 530).

Anuar, N.I., and Saptari, A., 2016. Performance evaluation of different types of particle representation procedures of Particle Swarm Optimization in Job-shop Scheduling Problems. In: *Proceedings of the 2nd International Manufacturing Engineering Conference and 3rd Asia-Pacific Conference on Manufacturing Systems*, Kuala Lumpur, Malaysia, 12–14 November 2015. IOP Conference Series: Materials Science and Engineering (Vol. 114).

CHAPTER 1

INTRODUCTION

In today's complex manufacturing setting, addressing crucial issues such as improvement of cost, throughput rate and customer satisfaction are among the challenging tasks encountered in the real-world manufacturing environment. The ability to cope efficiently with this type of situation will boost the firm's competitiveness. This study intends to address the subject by looking at benchmark instances and a published case study of job-shop configurations and introduces an improved hybrid metaheuristic approach to aid decision-makers in an effort to deal with these concerns.

1.1 Background

Scheduling belongs to combinatorial optimization problems and it is generally concerned with the assignment of tasks to limited resources over an interval of time (Amrane et al., 2021). The scheduling problem has widespread applications in a variety of settings. The problem appears in many important fields including hydrothermal systems (Kaur et al., 2020), transportation and logistics (Guo et al., 2018) and cloud computing (Wan and Qi, 2021).

Scheduling has become one of the major issues in the planning and operations of manufacturing systems. It has a direct impact on production efficiency, cost and quality. Issues in production scheduling involve the use of resources efficiently, reduction of production costs and delivery of high-quality products by given deadlines (Birgin et al., 2020). Due to the realistic expectations, production scheduling has gained considerable research effort among scheduling researchers (Fazel Zarandi et al., 2020). The challenge is to find a good assignment of an operation to a machine in order to obtain a schedule that optimizes certain pre-defined objective functions.

The Job-shop Scheduling Problem (JSP) belongs to one of the best known and most studied production scheduling problems (Zhang et al., 2019; Horng and Lin, 2021). It involves a permutation of jobs with the aim of optimizing one or more objectives. The JSP can commonly be described as having n jobs to be scheduled on m machines (Pinedo, 2016). Every job consists of an order of operations with precedence constraints. The operations of the job will follow the assigned processing route, specific for each job. The execution of job i on machine j is denoted as operation O_{ij}, and its duration is p_{ij}.

The JSP is regarded as significant because it reflects the actual operation of several industries and many complex industrial problems are generally modelled as the JSP. In a job-shop production environment, a number of jobs may require scheduling, each with a different processing sequence and a different processing time on the machines. Jobs may have promised delivery dates and the solution procedure differs as the objective of the scheduling changes. A typical example of real-life job-shop scheduling is the wafer fabrications in the semiconductor industry in which an order commonly involves a batch of a particular product type that has to go through the facility according to a given route with specific processing times.

The motivation behind production scheduling is the goal of attaining a job sequence in such a way that one or more objectives are optimized. The objectives can be grouped into either process-oriented or customer-oriented objectives (He et al., 2018). The common process-oriented objective is to find the optimum value of the makespan, i.e. the minimum completion time of the final job to leave the system (Bürgy and Bülbül, 2018). Makespan signifies a good measure of performance for the JSP; a schedule with minimum makespan suggests high machine utilization (Pinedo, 2016). The customer-oriented objectives portray close conformance to prescribed due dates. The due date-related objectives have been studied extensively. At present, many objective functions are developed such as minimum tardiness penalty costs (Kim et al., 2020), minimum total weighted earliness and tardiness (Wei et al., 2021), minimum mean tardiness (He et al., 2021), etc.

The majority of the works carried out for the JSP have concentrated on a single objective and the optimization of makespan. However, real-world scheduling problems are multi-objective by nature and thus it demands decision-makers to take into account a number of different objectives simultaneously. A multi-objective optimization involves a problem with a number of objectives to be achieved and these objectives are generally conflicting (Gunantara, 2018). Hence, the trade-offs involved in considering these conflicting objectives can provide the decision-makers with a better understanding of the problem where all the consequences of a decision with respect to all the objectives can be explored before arriving at any conclusion.

A wide range of solution methodologies has been proposed to solve the optimization of multiple objectives. At the very beginning, the research on single-objective optimization focused on the exact methods. Due to the need for solving large-scale scheduling problems and the deficiency in computational resources, it was soon identified that the exact methods were impractical. Therefore, the research has now centred on metaheuristic techniques.

There are various metaheuristic approaches reported in the literature for solving the JSP, which include Simulated Annealing (Garza-Santisteban et al., 2019), Tabu Search (Xie et al., 2021), Artificial Bee Colony algorithm (Hakim et al., 2019), Ant Colony Optimization (Chaouch et al., 2019), Genetic Algorithm (Mencía et al., 2021) and Particle Swarm Optimization (Anil Kumar and Das, 2020). These algorithms are able to find near-optimal solutions within an acceptable computational time.

Particle Swarm Optimization (PSO) is one of the computational intelligence techniques developed by James Kennedy and Russell Eberhart (Kennedy and Eberhart, 1995). The original algorithm was discovered through a simplified social model simulation. It is a population-based search algorithm and is initialized with a population of random solutions called particles. The particle flies through the search space with a velocity that is dynamically adjusted according to its own flying experience and its companions' flying experiences. Hence, the particles have a tendency to fly towards a better search area over the course of the search process. PSO has been observed to be capable of producing superior solutions at a very low computational cost, where it has performed considerably well in a broad range of applications (Wang et al., 2018).

PSO is initially proposed to solve continuous optimization problems, but it can also be modified to solve problems in discrete spaces, such as combinatorial optimization problems that involve sequencing or permutation. Its modified model, namely the discrete PSO, has been developed to achieve this purpose. Unlike the continuous PSO, each particle in the discrete PSO directly represents the candidate solution. For scheduling problems like the JSP, this means that for the solution representation, each particle is directly mapped to a sequence of operations or a schedule, instead of its position in the continuous search space. This is in order to establish a direct and efficient mapping in handling the JSP as a combinatorial optimization problem that is set in the discrete domain.

There are several approaches to solving the problems of multi-objective optimization using PSO. The most popular approach is based on Pareto dominance, which optimizes all objective functions simultaneously (Yasear and Ku-Mahamud, 2021). According to the notion of Pareto optimality, it produces a set of Pareto optimal solutions that are non-dominated with respect to each other (Gunantara, 2018). The non-dominated solutions represent diverse compromises or trade-offs among the objectives. When considering real-life cases, Pareto-optimal solution sets are commonly desired over single solutions since they are more practical in real-world production systems.

1.2 Problem Statement

The Job-shop Scheduling Problem (JSP) is one of the most difficult production scheduling problems in the industry, with the aim to obtain a sequence of jobs in optimizing one or multiple objectives. Although a single objective like makespan is often used, the achievement of multiple objectives such as the improvement of cost, machine utilization and on-time deliveries are among the greater concerns encountered in real-world production systems (Sha and Lin, 2010; Reza Tavakkoli-Moghaddam et al., 2011; Meng et al., 2018). Nevertheless, research works on solving the JSP with multiple objectives are still limited compared to the single objective (Lei, 2008a; Feng et al., 2010; R. Tavakkoli-Moghaddam et al., 2011; Wisittipanich and Kachitvichyanukul, 2013).

The existing methods used on the standard single-objective model are also impractical to directly be applied to real-world scheduling scenarios in solving multiple objectives simultaneously. Instead of a unique, single solution produced as the output in a single-objective case, there exists a number of solutions in a multi-objective case that correspond to the most feasible compromises among the objectives. A multi-objective case is also more challenging to solve as the objectives are normally in conflict with each other, where one objective cannot be improved without degrading at least another objective (Zitzler and Thiele, 1999).

There have been several methods and algorithms proposed to solve multi-objective problems. More recently, swarm intelligence approaches have been developed for this purpose (Yasear and Ku-Mahamud, 2021), where the success of the PSO algorithm in solving single-objective optimization problems has inspired research works in the extension of this method to problems of multi-objective optimization. In comparison with evolutionary algorithms, PSO has inherent advantages in scheduling problems. For instance, it does not have to devise special mutation or crossover operators to inhibit the presence of illegal individuals. Its structure is simpler, with a memory function to retain the best position of the population, as well as the best location of the individuals. It also contains less complex mathematical calculations and requires fewer parameter adjustments, which furnishes it with high search efficiency. The relative simplicity of PSO, its straightforward implementation and its adaptability to a wide range of domains have rendered it an emerging prospect to be extended for multi-objective optimization (Freitas et al., 2020). However, one main shortcoming of the basic PSO design is that the swarm is inclined to cluster rapidly towards the current best location, resulting in a stagnation of the search process when the swarm becomes stuck at a local optimum (Sengupta et al., 2018). This issue is magnified further when dealing with multi-objective optimization (Wang et al., 2018; Freitas et al., 2020). Thus, improvement strategies need to be implemented in the existing PSO design to address this rapid clustering problem in optimizing the JSP with multiple objectives. Hence, the first research question in this study is:

- **Research Question 1:** What strategies can be implemented to improve the rapid clustering problem in the existing PSO design in solving the JSP with multiple objectives?

Traditionally, PSO has been introduced as an optimization technique in the continuous search space where it works by adjusting trajectories through the manipulation of each coordinate of a particle. Each particle represents a solution encoded as a real variable in a multi-dimensional search space. All the dimensions are typically independent of each other, thus the updates of the velocity and the position are performed independently in each