RESEARCH PAPER

# An Empirical Study of Quantum Annealing Methods for Robust Time Dependent Job Shop Scheduling Optimization

**V.R.R Lakshmi Kumari[1], Dr. Animoni Nagaraju[2]**

[1]Research Scholar, Dept of Computer Science and Engineering, Malla Reddy Technical Campus, Malla Reddy Vishwavidyapeeth (Deemed to be University), Hyderabad, India. Email: srirajeswari.10@gmail.com

[2]Professor, Dept of Computer Science and Engineering, Malla Reddy Technical Campus, Malla Reddy Vishwavidyapeeth (Deemed to be University), Hyderabad, India. Email: dranagaraju.cse@gmail.com

***ABSTRACT***

A Robust Time Dependent Job Shop Scheduling Problem (RTDJSSP) is a complicated optimization problem for simulating real-world process scheduling tasks with competing objectives. To address RTDJSSPs, approximation approaches are used to ensure solutions are completed within acceptable timelines with varying sectional durations. Quantum Annealing, a metaheuristic that uses quantum mechanical effects, produces better solution quality in less time than conventional algorithms. However, due to hardware restrictions in quantum annealers, hybrid methods are required for solving bigger RTDJSSPs. This work explores the threshold issue sizes for which quantum annealers are sufficient and when hybrid algorithms are necessary, focusing on the distribution of computing resources in hybrid approaches.

**Keywords:** Quantum Annealing; Robust Time Dependent job shop scheduling; process scheduling; meta-heuristic; optimization, Time Dependent.

**How to cite this article:** Kumari VRRL, Nagaraju A. An Empirical Study of Quantum Annealing Methods for Robust Time Dependent Job Shop Scheduling Optimization. Int J Drug Deliv Technol. 2026;16(17s): 466-472.

DOI: 10.25258/ijddt.16.17s.54

**Source of support:** Nil.

**Conflict of interest:** None

## I. Introduction

The Production planning and control (PPC) includes scheduling, capacity and quantity planning, and monitoring manufacturing and assembly operations within a manufacturing system. PPC seeks to assure on-time production and delivery, consistent capacity utilization, shorter lead times, lower inventory levels, and greater flexibility [1]. However, the landscape of global crises resulted in rapid and major changes in the market dynamics of industrial firms. These shifts can cause swings in client demand or short-term material shortages. As a result, it is necessary to establish strategies that ensure manufacturing enterprises' economic sustainability. As a result, an effective PPC becomes a need for a production system's viability. Process scheduling, which deals with the sequencing and distribution of jobs, is very crucial within PPC [2]. The goal of process scheduling is to assign work to available resources within the industrial system in accordance with its objectives, hence increasing efficiency [3]. However, due to the high number of aspects to consider, such as multi-objective constraints and dynamic rescheduling, this is a complex optimization problem known as the job shop scheduling problem (JSSP). Although traditional methods such as precise algorithms are successful, they frequently demand lengthy calculation periods, limiting the system's flexibility and capacity to respond to unexpected occurrences [4].

As a result, there is a need for computational approaches that may produce efficient results in a short period. Recent advances in quantum annealing (QA) have showed promise in closing this research gap. This study demonstrated QA's capacity to provide quality solutions for static, dynamic, and multi-objective JSSPs. However, because to the intricacy of the problem, hybrid solver configurations and iterative approaches were used. Hybrid solutions, while effective, inherit some of the disadvantages of traditional procedures, such as slower computer times. Iterative approaches, while narrowing the solution space, may overlook potentially better alternatives. To fully realize the potential of QA, it is critical to understand the limitations in terms of problem sizes when QA is feasible, when hybrid approaches are required, and when an iterative approach is required to find solutions within an acceptable timescale. To overcome these problems, many problem instances are constructed and computed. Based on the findings, an effort is made to provide insights into the limitations of QA's application to JSSPs.

The paper is organized as follows: Section 2 summarizes the current state of JSSP solving and introduces QA. Furthermore, the current QA-based approaches to JSSP will be explored. After identifying the research gap, Section 3 describes the structure of the QA-based algorithms. The use cases are then offered, which serve as the foundation for the investigations into the QA-based methodologies. The experimental findings are then detailed. Ends with a view and summary.

## II. Related Work

*A Job shop scheduling*

Job shop scheduling (JSS) is an optimization issue in operations research that is widely used for manufacturing process scheduling. The formulation of JSS challenges varies according to the limitations and objectives. Classical JSSP attempt to allocate provided jobs, which consist of consecutive processes, to manufacturing system machines while taking processing times into account in order to fulfill particular objectives . These objectives include a variety of parameters such as time, employment quantity, cost, revenue, energy, and environmental considerations. Notably, the most common purpose is to reduce job lead times. In addition, restrictions play an important role in mapping production system conditions to problem formulations . Procedure, overlapping, and processing limitations are all examples of common constraints . Assumptions include consistent machine availability, the absence of job priorities and due dates, the disregard of transport and setup times, and fixed operation-machine assignments . However, in order to better depict real-world events, several JSSPs vary from these assumptions.

Robust job shop scheduling (FJSSP) posits that machines can complete identical jobs, allowing for flexible operation assignments and machine-specific processing durations . Dynamic job shop scheduling (DJSSP) considers time-based events such as machine failures and probabilistic job arrivals . The intricacy of JSSPs, combined with their NP-hard nature, presents considerable hurdles to scheduling algorithms. Consequently, obtaining optimal solutions is time-consuming. for medium-sized situations, and frequently impractical for bigger cases. Brucker et al.'s findings highlight this difficulty by demonstrating the processing costs of branch and bound algorithms applied to various JSSP situations Thus, approximation approaches are used to efficiently compute solutions for bigger optimization problems. Approximation techniques include constructive methods, artificial intelligence approaches, local search algorithms, and meta-heuristics . These optimization techniques are used either singly or in combination to take use of their unique benefits in dealing with various JSSP scenarios. The choice and adaptation of these strategies are determined by the size of the challenge and the modeling methodologies used. Many techniques to addressing JSSP suffer a common trade-off: balancing the complexity of the problem or aims with the necessity for fast computation, which frequently limits their applicability to small-scale situations. Exact approaches are impractical due to high computing periods. While approximation approaches can handle bigger issue sizes, they either increase processing time or reduce solution quality owing to premature termination. As a result, strategies are being investigated to manage the trade-off between computation time and solution quality . Recent research in the field of QA-based algorithms addresses this trade-off by investigating novel techniques to improve efficiency while maintaining solution accuracy and robustness .

*B. Quantum Annealing*

The first implementation of QA happened in 2011, with the creation of a cloud-based quantum annealer, also known as an adiabatic quantum computer . Since then, the qubit count, which is a measure of hardware capacity, has increased exponentially, from the D-Wave One in 2011 with 128 qubits to the most recent D-Wave Advantage with 5640 qubits . QA uses quantum mechanics principles to efficiently locate optimal or near-optimal solutions in a short period of time by searching for energy-minimal states of an optimization issue . The usage of Hamilton function notation is one way for converting an optimization issue into an energy minimization problem appropriate for quality assurance.The Hamilton formulation assigns an energy value to each state of the optimization problem, utilizing binary variables $x_i$ , $x_j$ , $x_k$ and corresponding sclar weights

$$H = \sum QQ_{ii}\mathrm{k}x_i + \sum QQ_{ijj}\mathrm{k}x_i\, x_{jj}\mathrm{k}$$

To solve the Hamilton function with QA, it is necessary to map it onto the quantum processing unit, which is a key step in any quantum method that uses qubits. However, this change, known as embedding, can be difficult for quantum annealers since it often necessitates the use of more qubits and takes longer to complete. As a result, a variety of embedding approaches are used, including auto embedding and external embedding . As a result, the need to overcome hardware restrictions encourages the use of hybrid solvers and iterative strategies to successfully manage bigger issue sizes while solving the Hamilton function with QA. This trend is visible in the literature on JSSP calculations with QA, where iterative and hybrid solution techniques are already widely used.

*C. Job shop scheduling using Quantum Annealing*

In 2015, Venturelli et al. performed the first implementation of a JSSP on a quantum annealer. For implementation, the D-Wave two with 509 qubits and the Chimera topology are employed. This study investigates the D-Wave machine's computing efficiency in terms of task complexity. To circumvent hardware limits, the researchers used variable pruning approaches, which reduced issue sizes and allowed for the computation of bigger problem instances. Nonetheless, the results suggest that the probability of optimal solutions falls as problem size increases.

Furthermore, no explicit upper bound is specified for the largest problem size that the quantum annealer can handle. Even in this investigation, the superiority of QA over classical algorithms in terms of solution quality and efficiency could not be definitively proved, as only modest issue sizes were used. Nonetheless, this work serves as a watershed moment for many future research attempts in the discipline. Further research has investigated the difficulties associated with constraints on benchmark problems such as ft06, adopting window shaving techniques to permit mapping onto the D-Wave 2000 with the Chimera topology, hence repeatedly lowering variables. However, no specific attention to variable boundaries was addressed . More study has looked at the JSSP bounds for quadratic instances, finding up to 15 tasks, operations, and machines on Chimera topologies and up to 26 jobs, operations, and machines on Pegasus topologies on D-Wave Advantage . These studies provide preliminary indicators of the limitations of classical QA, however they do not take into account classical JSSP with varied processing times and flexible machine assignments in FJSSP. The computation of FJSSP with hybrid solvers was investigated in .

Hybrid iterative approaches outperformed non-iterative hybrids in terms of computational time, but at the expense of slightly inferior solution quality. However, the question of choosing the best solution approach for various application circumstances remains unsolved. Previous research has focused on dynamic and multi-criteria objectives , with iterative and hybrid solution approaches used due to the complexity of the problem. However, the limitations of the solvers utilized have mostly gone unresolved.

*D. Research gap*

It is unknown when a FJSSP requires iterative or hybrid solver approaches to computation. Underutilization of The potential of quantum annealers may result in slower computation times, yet the premature adoption of iterative solution approaches may limit the solution space and likely lead to inferior solution quality. As a result, investigating key limits and providing insights into performance is critical for making educated judgments.

## III. System Model

Investigations into the limitations of employing Quantum Annealing for robust job shop scheduling.

### 3.1 Framework and algorithmic structure

The proposed approach investigates the boundaries of using QA to solve FJSSPs. The FJSSP's machines ($M$) and tasks ($\boldsymbol{V}$) are initially merged with the supplied time range ($T$) to create binary variables. These variables form the basis for setting mathematical limits and objectives for the Binary Quadratic Model (BQM). The mathematical model of the FJSSP suitable for QA may be found in and , thus they will be briefly addressed. The BQM formulation uses binary variables $x_{iiikt} \in X$ equal to 1 if an operation $o_{iit}$ from job $i$ starts at time $t$ and is processed on machine $m_k$. Otherwise, the variable is equal to zero. By linking the variables, The BQM is constructed using the corresponding weights (Lagrange parameters) and factors such as processing times. It is then utilized to generate the Hamiltonian function. The values of Lagrange parameters are determined by the priorities of the objectives and constraints, with constraints often receiving more weight than objective functions. The FJSSP has three constraints: a processing constraint that ensures no duplicate starting times for a single operation, a procedure constraint that ensures the prescribed sequence of operations is followed, and an overlapping constraint that prevents multiple operations from starting on the same machine.

These limitations are enforced by penalty functions, which impose greater energy values in the

Hamiltonian for noncompliance. Furthermore, the makespan objective is met by punishing deviations from a minimum preceding time. The minimal time is calculated by adding the minimum processing times of the preceding processes. Finally, the Hamiltonian function is mapped to quantum hardware and solved utilizing QA-based techniques. This paper uses the following solver configurations:

- Conventional D-Wave quantum processing unit (QPU) samplers.
- Hybrid algorithm combines QA and simulated annealing. and Tabu search in parallel (HQPU).
- IHQPU-based iterative algorithm

Given that the most recent QA topologies, Pegasus and Zephyr, differ in terms of available qubits and connection, which may influence outcomes, both topologies are applied to each setup. The iterative solver setup utilized is based from and illustrated in Figure. The iterative technique starts with the Lagrange parameters of the constraints and objective.
$\alpha\alpha$, $\beta\beta$, $\gamma\gamma$, $\delta$ and a time range $T$ are determined as well as a bottleneck factor $\delta 0^{(LOOP)}$ for each job. Jobs are prioritized based on bottleneck criteria. Makespan jobs are prioritized with long total processing times and a limited number of machines. An initial set of jobs J is separated into smaller subsets of jobs Js based on bottleneck criteria. For each loop, an appropriate number of operations (Os) is chosen. The BQM for the associated subproblems is then created and solved using the chosen solver. This is performed for each subproblem, with the bottleneck factors and time intervals changed correspondingly. When all subproblems have been solved, the solutions to all loops are combined.

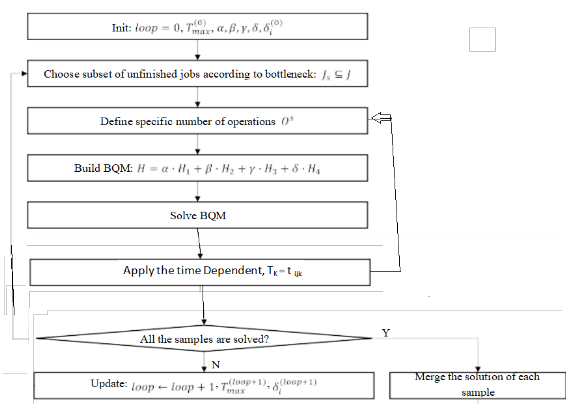



Figure 1: Iterative workflow

## IV. Quantum Annealing for Job Shop Scheduling Use Cases

The solver configurations are applied to FJSSPs to determine their limitations. These studies are divided into three experimental sets, each aiming at investigating a different element of these limits. The first experimental configuration, inspired by Carugno et al.'s work, uses quadratic instances. In these cases, JSSPs are distinguished by scaling the number of operations per job in accordance with the number of machines and workloads. Furthermore, the time range provided, including the discrete intervals when each operation can be performed, is variable. Each operation's processing time ($p_{ijt}$) is standardized to 1. As a result, optimal solutions can be identified, allowing for an analysis of the maximum number of processable variables in the BQM while also taking into account the computing times for each solver. The second experimental setup uses FJSSP to adjust the number of viable machines, k, for each operation while keeping constant processing times and $T_r$.

This architecture allows for a study into how the amount of quadratic interactions in the BQM influences solution quality and processable issue sizes. In the third experimental setting, processing times ($pij$ and Tr.) are changed. This variation allows for an investigation of how variations in processing times affect the amount of processable jobs and operations. Table 1 provides detailed information on the experimental setups. The computations are carried out on an XEON_SP_6126 with 20 GB RAM and the corresponding quantum gear. A time constraint of 15 minutes is established for completing a solution using the appropriate solver settings.

Table 1: Experimental setup

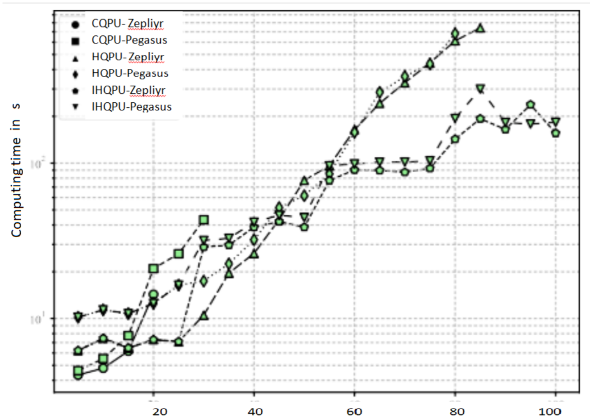
| Experimental Setup | J = O = M | k | p | T | Tr |
|---|---|---|---|---|---|
| 1 | 1-80 | 1 | 2 | 3-90 | 1-4 |
| 2 | 5-40 | 5-40 | 2 | 6-45 | 3 |
| 3 | 10-60 | 10-60 | 1 | 12-70 | 2-6 |
| 4 | 3-30 | 3-30 | 3 | 5-35 | 4 |

## V Simulation and Results

Table 2 summarizes selected computational outcomes with a focus on the restrictions. A variety of parameters are documented for evaluation reasons. These include the individual solver setups as well as computation times Tc, which represent the time it takes to retrieve a solution from a problem submission. Additionally, solution quality can be evaluated using the achieved makespan ms. Furthermore, the properties of each problem are provided, including the number of variables (nv) and quadratic interactions (nq) within the BQM, as well as the amount of qubits used for embedding. Each line has six numbers for Tc and ms, representing the results of CQPU-Zephyr and CQPU.

Pegasus, HQPU-Zephyr, HQPU-Pegasus, IHQPU-Zephyr, and IHQPU-Pegasus, with a dash indicating that no solution could be discovered within the specified time span or owing to hardware restrictions. Only CQPU-Zephyr and CQPU-Pegasus have Ne data collected. It is crucial to note that, depending on the magnitude of the problem, the solutions provided by IHQPU and HQPU are only partially equivalent. This is because the parent problem is only partitioned when it reaches a specified size threshold. Figure 2-4 shows the findings of experimental setup 1 with Tr = 2, experimental setup 2 with J = O = M = k, and experimental setup 3 with Tr = 2 and J = O = M = k = p. Additional findings can be found in the supplemental resources.

The computational findings demonstrate the benefit of the Zephyr topology over the Pegasus topology. In experimental setup 1 with Tr=2, both CQPU-Zephyr and CQPU-Pegasus attain optimal makespans, however CQPU-Zephyr has quicker calculation times. This difference is most likely due to the larger availability of connected qubits in Zephyr topologies, which results in fewer substantial increases in issue size during embedding than Pegasus. However, Zephyr's potential is constrained by its fewer available qubits, which limits its solvable problem size to 20 for CQPU-Zephyr. In contrast, CQPU-Pegasus can tackle issues up to size 32. The HQPU results show advantages in manageable issue sizes due to the enhanced CPU resources for computation. In this case, HQPU-Zephyr can handle problem sizes of up to 86, whereas HQPU-Pegasus can find workable solutions for up to 96 in 15 minutes. This is because the HQPU-Zephyr configuration consistently exhibits shorter computing times than HQPU-Pegasus due to the lower embedding effort.




Problem size

Figure 2: Experimental setup 1 with J=O=M,T

| Ci | Total | Cumtotal | max | Cummax |
|---|---|---|---|---|
| 0.05 | 9 | 9 | 25 | 25 |
| 0.2 | 14 | 23 | 13 | 38 |
| 0.35 | 28 | 51 | 17 | 55 |
| 0.5 | 36 | 87 | 21 | 76 |
| 0.65 | 8 | 95 | 9 | 85 |
| 0.8 | 3 | 98 | 6 | 91 |
| 0.95 | 1 | 99 | 4 | 95 |
| 1.1 | 1 | 100 | 2 | 97 |
| 1.25 | 0 | 100 | 2 | 99 |
| 1.4 | 0 | 100 | 1 | 100 |

Table 2: The total and max are shown in the following figure to show how the objectives perform.

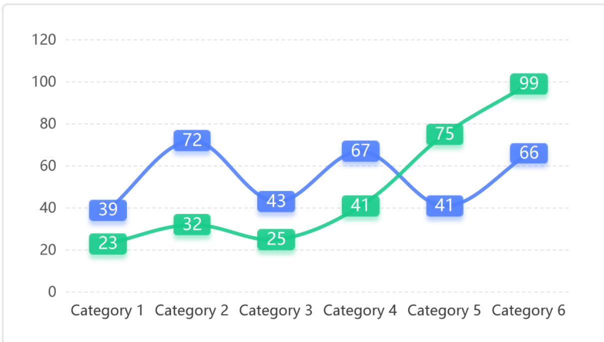



Figure 3: Variation of class intervals versus cumulative frequency

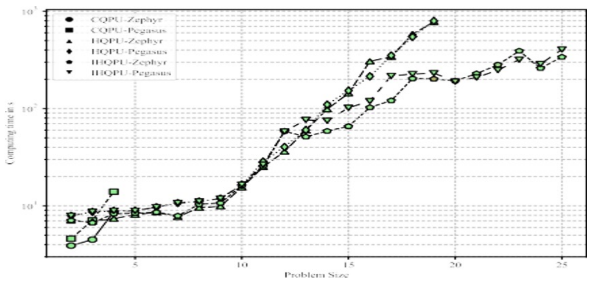


Figure 4: Experimental setup 3 with J=O=M=k=p ,T

## VI Result Analysis:

| J = O = M | k | p | T | Tr | nv | nq | Tc in s[1] | ms[1] | ne[2] |
|---|---|---|---|---|---|---|---|---|---|
| 18 | 1 | 1 | 19 | 2 | 720 | 1040 | 12.5/18.6/7.2/10.4/7.2/10.4 | 18/18/18/18/18/18 | 980/1205 |
| 30 | 1 | 1 | 31 | 2 | 1800 | 2600 | -/180.2/12.4/16.8/12.4/16.8 | -/30/30/30/30/30 | -/2800 |
| 70 | 1 | 1 | 71 | 2 | 9800 | 14500 | -/-/650.2/540.1/200.5/140.3 | -/-/70/70/70/70 | -/- |
| 90 | 1 | 1 | 91 | 2 | 16200 | 24000 | -/-/900.6/-/180.4/260.7 | -/-/90/-/90/90 | -/- |
| 40 | 1 | 1 | 41 | 2 | 3200 | 4700 | -/-/40.2/48.6/38.5/52.3 | -/-/40/40/40/40 | -/- |
| 52 | 1 | 1 | 53 | 2 | 5400 | 8000 | -/-/70.4/75.1/55.2/60.8 | -/-/52/52/52/52 | -/- |
| 22 | 1 | 1 | 23 | 2 | 900 | 1300 | 10.2/14.8/7.1/9.5// | 22/22/22/22/22/22 | 1100/1350 |

Experimental setting three gives similar results, since altering processing time increases variables and quadratic interactions, resulting in a smaller computable problem size. In experimental setup 3, however, non-optimal solutions are more common during the iterative phase. This higher frequency is most likely due to an enlarged solution space, which results in smaller variances in the weighting of individual solutions. In general, only a small percentage of computations fail to find the optimal makespan. However, given the prevalence of non-optimal solutions in previous investigations, this shows a link between effective solutions and issue structures. As a result, the next phase will investigate the behavior of individual solver configurations in industrial settings, highlighting the importance of hybrid and iterative techniques in tackling industrial-scale challenges.

## VI. Conclusion

This research examines the performance of various QA-based solver setups on many instances of JSSP and RTDFJSSP issues. These configurations use QPU topologies, specifically Pegasus and Zephyr, for CQPU, HQPU, and IHQPU. It was discovered that configurations using the Zephyr topology have shorter computing times than those using Pegasus, albeit limited by hardware limits in addressing bigger problem sizes. The performance of the embedding process suffers significantly, especially when the problem size approaches the hardware's capacity limits. In certain cases, hybrid approaches outperform conventional methods. However, as problem sizes grow, the HQPU's processing times increase as well. As a result, adopting IHQPUs becomes useful beyond a certain problem size threshold, but at the risk of reducing solution quality.

However, because the topics under study are theoretical, it makes sense to broaden the research. Future research aims to extend these investigations into industrial settings, encompassing diverse multi-criteria objectives and significant disparities in processing times. We have also compared the objective of this problem by obtaining the total and minmax time individually computed and discovered that the objective gives a better optimum.

## References


[1] Schuh G, Brosze T, Brandenburg U, Cuber S, Schenk M, Quick J et al. Grundlagen der Produktionsplanung und -steuerung. In: Schuh G, Stich V, editors. Produktionsplanung und -steuerung 2. Berlin, Heidelberg: Springer Berlin Heidelberg; 2012, p. 9–293.

[2] Zäpfel G, Missbauer H. New concepts for production planning and control. Eur J Oper Res 1993;67(3):297–320. https://doi.org/10.1016/0377-2217(93)90287-W.

[3] Pinedo ML. Scheduling. Cham: Springer International Publishing; 2016.

[4] Zhang J, Ding G, Zou Y, Qin S, Fu J. Review of job shop scheduling research and its new perspectives under Industry 4.0. J Intell Manuf 2019;30(4):1809–30. https://doi.org/10.1007/s10845- 017-1350-2.

[5] Schworm P, Wu X, Glatt M, Aurich JC. Solving flexible job shop scheduling problems in manufacturing with Quantum Annealing. Prod Eng Res Devel 2022. https://doi.org/10.1007/s11740-022- 01145-8.

[6] Schworm P, Wu X, Glatt M, Aurich JC. Responsiveness to sudden disturbances in manufacturing through dynamic job shop scheduling using Quantum Annealing. Procedia CIRP 2023;120:511–6. https://doi.org/10.1016/j.procir.2023.09.028.

[7] Schworm P, Wu X, Klar M, Glatt M, Aurich JC. Multi-objective Quantum Annealing approach for solving flexible job shop scheduling in manufacturing. Journal of Manufacturing Systems 2024;72:142–53. https://doi.org/10.1016/j.jmsy.2023.11.015.

[8] Xiong H, Shi S, Ren D, Hu J. A survey of job shop scheduling problem: The types and models. Comput Oper Res 2022;142:105731. https://doi.org/10.1016/j.cor.2022.105731.

[9] Chaudhry IA, Khan AA. A research survey: review of flexible job shop scheduling techniques. Intl Trans in Op Res 2016;23(3):551–91. https://doi.org/10.1111/itor.12199.

[10] Mohan J, Lanka K, Rao AN. A Review of Dynamic Job Shop Scheduling Techniques. Procedia Manuf 2019;30:34–

9. https://doi.org/10.1016/j.promfg.2019.02.006.

[11] Sotskov Y, Shakhlevich NV. NP-hardness of shop-scheduling problems with three jobs. Discret Appl Math 1995;59(3):237–66. https://doi.org/10.1016/0166-218X(95)80004-N.

[12] Brucker P, Jurisch B, Sievers B. A branch and bound algorithm for the job-shop scheduling problem. Discrete Applied Mathematics 1994;49(1-3):107–27. https://doi.org/10.1016/0166-218X(94)90204-6.

[13] Gao K, Cao Z, Le Zhang, Chen Z, Han Y, Pan Q. A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. IEEE/CAA J. Autom. Sinica 2019;6(4):904–16. https://doi.org/10.1109/JAS.2019.1911540.

[14] Genova K, Kirilov L, Guliashki V. A Survey of Solving Approaches for Multiple Objective Flexible Job Shop Scheduling Problems. Cybernetics and Information Technologies 2015;15(2):3–22. https://doi.org/10.1515/cait-2015-0025.

[15] Johnson MW, Amin MHS, Gildert S, Lanting T, Hamze F, Dickson N et al. Quantum annealing with manufactured spins. Nature 2011;473(7346):194–8. https://doi.org/10.1038/nature10012.

[16] McGeoch C, Farré P. The d-wave advantage system: An overview. D-Wave Systems Inc., Burnaby, BC.

[17] Crispin A, Syrichas A. Quantum Annealing Algorithm for Vehicle Scheduling. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics. IEEE; 2013, p. 3523–3528.

[18] Zbinden S, Bärtschi A, Djidjev H, Eidenbenz S. Embedding Algorithms for Quantum Annealers with Chimera and Pegasus Connection Topologies. High Performance Computing 2020;12151:187–206. https://doi.org/10.1007/978-3-030-50743-5_10.

[19] Venturelli D, Marchand DJJ, Rojo G. Quantum Annealing Implementation of Job-Shop Scheduling; 2015.

[20] **Nagaraju Animoni**, Dr. Y. Rama Devi., Dynamic Trade Variant of Time Dependent Scheduling in Cloud Compute Based on Parallel Load Balancing, International Journal of Advanced Science and Technology(2020),**Vol. 29 No. 04 (2020): Vol 29 No. 4 (2020)**

[21] Kurowski K, Węglarz J, Subocz M, Różycki R, Waligóra G. Hybrid Quantum Annealing Heuristic Method for Solving Job Shop Scheduling Problem. In: Krzhizhanovskaya VV, Závodszky G, Lees MH, Dongarra J, Sloot P, Brissos S et al., editors. Computational Science – ICCS 2020: 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part VI, 1st ed. Cham: Springer International Publishing; Imprint Springer; 2020, p. 502–515.

[22] Carugno C, Ferrari Dacrema M, Cremonesi P. Evaluating the job shop scheduling problem on a D-wave quantum annealer. Sci Rep 2022;12(1):6539. https://doi.org/10.1038/s41598-022-10169-0.