IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

Efficient Industrial Robot Scheduling with Heuristic-Based AIS Algorithm Approach

Ajay Shrivastava
Assistant Professor, RKDF University, Bhopal

Abstract: *Scheduling industrial robots is a highly important optimization problem that has a direct influence on the productivity of manufacturing, energy usage, and cost of operation. The modern robotic manufacturing setting is dynamic, multi-objective, and NP-hard, which makes traditional scheduling approaches incapable of addressing the requirements of such an environment. In this paper, I shall suggest an Efficient Industrial Robot Scheduling model based on an Artificial Immune System (AIS) algorithm developed by Heuristic definition. The AIS framework, based on the biological immune system, is a combination of rules that are heuristic and clonal selection, mutation, and immune memory to allocate tasks and sequence tasks optimally. A mathematical formulation of the scheduling problem is developed in which the makespan, and robot utilization and energy consumption are used as objective functions. The results of the simulation prove that the given approach is much more efficient than the traditional heuristic and genetic algorithms in the speed of convergence, scheduling effectiveness, and the robustness of solutions.*

Keywords: Industrial Robots, Scheduling Optimization, Artificial Immune System, Heuristic Algorithms, Manufacturing Systems

I. INTRODUCTION

The precision, flexibility, and the continuous operation in harsh conditions have transformed industrial robots into an unavoidable part of the contemporary automated manufacturing systems. They are used in various industrial works including welding, assembly, material handling, painting, packaging and quality inspection. With the evolution of manufacturing systems toward Industry 4.0 high-mix and low-volume production and smart factories, the efficient control of the industrial robots in scheduling has become a major problem. The proper robot scheduling directly affects the main performance indicators such as the production time, the system throughput, the use of resources, and the operational cost.

Robot scheduling entails the identification of the best distribution and sequence of activities by several robots with respect to technological and operation limitations. It is a complex and combinatorial problem which is inherently increasing exponentially with the number of tasks and the number of robots. Industrial robot scheduling is thus an NP-hard optimization problem with exact methods of optimization being computationally infeasible when dealing with large scale, or real-time applications.[1] Poor scheduling may cause too much idleness, bottlenecks, use of more energy, and low productivity.

The classical methods of scheduling have been characterized by the terms like: First-Come-First-Serve (FCFS), Shortest Processing Time (SPT), Earliest Due Date (EDD) and other heuristic rules are easy to apply and computationally inexpensive. These approaches are however not flexible and resilient in the context of dynamism in manufacturing, uncertain processing time or multi-objective optimization.[2] They tend to be not made to deal with dynamic situations and do not yield near-optimal solutions to complex robotic systems in which priorities, workloads, and operational constraints are constantly changing.

[1] Pinedo, M. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer.
[2] Baker, K. R., & Trietsch, D. (2009). *Principles of Sequencing and Scheduling*. John Wiley & Sons.

As a response to such constraints, metaheuristic optimization methods, such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) and Simulated Annealing (SA) are becoming the focus of research. The algorithms can search large search spaces and give quality solutions at realistic computational time.[3] In spite of these benefits, most metaheuristic algorithms have problems like premature convergence, loss of diversity of solutions and sensitivity to parameterization which can minimize the validity of such algorithms in complex robot scheduling contexts.[4]

Another promising alternative to solve complex optimization problems is Artificial Immune System (AIS) algorithms which are based on the biological immune system. Clonal selection, affinity maturation, immune memory, and maintenance of diversity are some of the principles in the AIS and allow powerful global search in conjunction with adaptive learning.[5] The AIS unlike the traditional evolutionary algorithms, inherently maintains the diversity of the solutions and maintains high-quality solutions in the form of immune memory, which serves to inhibit early convergence and increase solution stability.

This paper submits a proposal of heuristic-based Artificial Immune System algorithm that can be effectively used in scheduling of industrial robots. The proposed methodology will enhance convergence speed, robustness and optimality of solutions by combining the domain specific scheduling heuristics with AIS mechanisms. The hybrid model uses the heuristic knowledge to inform the starting population and make use of the optimization offered by AIS to optimize the allocation and sequencing of tasks. The efficiency of the suggested method is reflected in the enhancement of the performance on the reduction of makespan, the use of the robot, and the efficiency of scheduling on the whole. The research is part of the increased literature on smart, bio-inspired techniques of optimization in advanced manufacturing system.[6]

II. PROBLEM DESCRIPTION

Given a manufacturing system that comprises of:

Industrial Robots (R) : The manufacturing system comprises of Ritual robots which are autonomous units of processing. Such robots perform preset tasks like welding, assembly or handling materials. To be able to make the robots work to their full potential and reduce the number of cases when they do not work at all, a good coordination is necessary in the automated production settings.[7]

Independent Tasks (N) : Nindependent tasks are contained in the system; each is an operation that is involved in manufacturing and has a known processing time. Activities are not supposed to be constrained by precedences and thus can be freely assigned. Appropriate allocation of tasks among robots plays a major role in the performance of a system and timeliness.

Single-Robot Task Processing : One robot should process one task at a time to clear up the operations and eliminate redundancy. Splitting of tasks or running them in parallel is not allowed which makes it easy to control but more difficult to schedule in terms of exclusive assignment.

Single-Task Robot Constraint : Each robot is capable of handling a single-task at a time, and therefore it is not allowed to handle multiple tasks simultaneously by using the same robot. This is a constraint that is reflective of real world operational constraints and requires accurate scheduling to prevent any conflicts and guarantees a smooth flow of work.[8]

---

[3] Jain, A. S., & Meeran, S. (1999). Deterministic job-shop scheduling: Past, present and future. *European Journal of Operational Research*, 113(2), 390–434.
[4] Talbi, E. G. (2009). *Metaheuristics: From Design to Implementation*. Wiley.
[5] De Castro, L. N., & Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer.
[6] Zhang, Y., Wang, L., & Zheng, D. Z. (2020). Intelligent scheduling for flexible manufacturing systems: A review. *Journal of Manufacturing Systems*, 54, 196–210.
[7] Pinedo, M. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer.
[8] Jain, A. S., & Meeran, S. (1999). Deterministic job-shop scheduling: Past, present and future. *European Journal of Operational Research*, 113(2), 390–434.

Assumptions:

Deterministic Task Processing Times : Task processing times are determined to be deterministic and known also. This assumption does away with the uncertainty in the scheduling and it is possible to compute the completion times, makespan, and the use of the robots accurately. Deterministic modeling is usually embraced in industrial scheduling to make the optimization and performance analysis easier.

No Task Pre-emption : Task preemption is not allowed, that is, when a robot is processing a task, it is not allowed to interrupt it. This is an indication of the real manufacturing limits where halting and restarting of production can lead to a decrease in the quality or extra expenses and therefore enhancing the significance of proper initial task sequencing.[9]

Setup Times Part of Processing : Task processing times contain setup times needed to execute tool changes, to calibrate a tool or to position a tool. This reduces the number of setup variables needed to represent the realistic operational conditions and at the same time simplifies the scheduling model. Scheduling of robots and optimization of manufacturing systems have been extensively studied based on such aggregation.

Homogeneous Robots : Homogeneous robots All in the system, the robots are supposed to be homogeneous with the same capabilities, processing speeds, energy properties. This supposition makes the optimization centered on sequencing the tasks and not choosing the robot, which allows the evaluation of the scheduling algorithms more clearly and compares the performance accordingly.[10]

III. MATHEMATICAL FORMULATION

3.1 Decision Variables

The decision variable xijis the variable which is a binary variable that takes the form of an assignment of jobs to the industrial robots in the manufacturing system. It can be mathematically written as:

$$x_{ij} = \begin{cases} 1, & \text{if task } i \text{ is assigned to robot } j \\ 0, & \text{otherwise} \end{cases}$$

I is the index of independent tasks and j is the index of the industrial robots, N and R are the numbers of activities and robots respectively. This variable is the key element of the scheduling model because it is used directly to specify the robot that performs each task.

The model that is being used with a binary decision variable makes the task assignments unambiguous and mutually exclusive. One and zero are used to show active assignment and no allocation respectively. The formulation can impose the necessary scheduling constraints, e.g., make sure that only one robot is capable of processing a certain task, make sure that robots do not overload their processing capacity.

Moreover, the objective functions, which are makespan minimization, maximisation of robot utilisation, and minimisation of energy consumption can be mathematically formulated and calculated using the decision variable x ijallows. With the help of such a variable, the interaction of tasks and robots can be analyzed systematically and, therefore, the overall performance of the industrial robot scheduling system can be optimized in a structured and computationally efficient way.

3.2 Objective Functions

(a) The reduction of the makespan in industrial robot scheduling is one of the main goals because it is directly proportional to the overall time spent on all the allocated tasks. C max is the maximum completion time of all robots in the system, C max = C max is mathematically represented by:

$$C_{\max} = \max_{j=1,2,\ldots,R} \left(\sum_{i=1}^{N} p_i \; x_{ij} \right)$$

[9] Pinedo, M. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer.

[10] Baker, K. R., & Trietsch, D. (2009). *Principles of Sequencing and Scheduling*. John Wiley & Sons.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

where p i is the processing time of task i, and x ij is a binary decision variable taking on the value of whether task i is allocated to robot j or not. The inner summation determines the total process time of all tasks performed by a given robot whereas the outer maximum operator determines the robot with the greatest amount of work.

Minimizing Cmax guarantees that there is a balanced allocation of tasks within the robots, and the idle time is minimized as well, and the throughput of the system is increased. It is especially significant in a world of automated manufacturing as a shorter makespan results in an increased production cycle and more efficient utilization of robotic resources.

3.3 Maximization of robot Utilization

(b) This is the maximization of robot utilization

The usage of robots is a significant performance measure, which addresses the efficiency of the utilization of the existing robots over the period of the schedule. Mean robot utilization: The mean robot utilization, which is denoted as U, is given as:

$$U = \frac{1}{R} \sum_{j=1}^{R} \frac{\sum_{i=1}^{N} p_i \ x_{ij}}{C_{\max}}$$

where p i is the processing time of task i, x jis the binary decision variable exploratory of the assignment of task ito robot j, and C max is the makespan of the time table.

The numerator of the fraction would be a sum of the processing time attributed to robot j and the denominator would be the amount of time that the robot has, which is determined by the makespan. The average of all robots is used to give a general utilization of the system. The Uencourage operation will give optimization in distributing work evenly, reduce idle time and the optimal utilization of the robots resources during the manufacturing process.

The minimum energy consumption is achieved by implementing the technologies, standards, practices, and procedures specified in the energy policy.

(c) Minimization of Energy Consumption The reduction of energy consumption is realized through the adoption of the technologies, standards, practices, and procedures upheld in the energy policy.

Minimization of energy consumption is a very important aim of the industrial robot scheduling, especially in energy-sensitive and sustainable industrial manufacturing systems. **The overall energy use of the robotic, E, is determined as:**

$$E = \sum_{j=1}^{R} \sum_{i=1}^{N} e_{ij} \ x_{ij}$$

The consumption of energy by robot j on task i denoted e ijrep, and is a binary decision variable indicating whether task i should be performed by robot j. This formulation determines the total energy consumption of all the robots and given tasks.

Minimization ensures less consumption of energy in the allocation of tasks and minimizes costs in production, wear and tear of equipment, and other environmental effects. This goal is especially significant in contemporary smart manufacturing settings, where the energy efficiency is one of the primary performance measures along with productivity and throughput.

The third objective is to be the first to launch a multi-objective environmentally friendly function in Iran and the Gulf region.

3.4 Integrated Multi-Objective Environmentally-Friendly Function

The third goal is to become the first one to introduce a multi-purpose environmentally-friendly operation in Iran and the Gulf.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

Scheduling of industrial robots is usually associated with several different and incompatible goals, including completion time minimization, resource use optimization and energy wastedness reduction. To manage these goals at the same time, they are derived to weighted-sum multi-objective as:

$$\min\ Z = w_1 C_{\max} - w_2 U + w_3 E$$

where Cmax is the make-span, Urep is the mean utilization of robots and Eis is the total system energy usage. The w 1, w 2 and w 3 are non-negative variables representing the weighting factor of each objective.

The negative sign on Uconverts the maximization of robot utilization to a minimization form so that all the objectives may be maximized in a single minimization framework. The model can be altered to suit multiple industrial needs by varying the weighting coefficients to focus on the different operational objectives, which may include faster production, increased efficiency or reduced energy consumption.

3.5 Constraints

a. Task Assignment Constraint

The case with industrial robot scheduling is that one must ensure that there are no assignments to more than one robot. This is the condition entailed by the task assignment constraint which mathematically is:

$$\sum_{j=1}^{R} x_{ij} = 1, \forall i = 1,2,\ldots,N$$

In this case, x ij is a binary decision variable that will take the value of 1 when task i is allocated to robot j, and the value of 0 when not. Summing of every robot j=1,2, etc, Rensures that no task i exists without a robot assigned to it, and also among the assigned robots there is none that is assigned several times.

This limitation ensures that there is a correct distribution of tasks within the robotic system and that it is a guiding principle of the acceptable scheduling solutions. In the absence of this, the schedule can be rendered infeasible resulting in idle work, conflicts or operational delays.

b. Binary Constraint

The binary constraint provides that the decision variable x jican can only have two possible values:

$$x_{ij} \in \{0,1\}$$

In this case, x ij = 1 means that task i is being allocated to robot j and x ij = 0 means that it is not being allocated to that robot. This restriction guarantees the discrete character of the scheduling problem, which makes it clear and eliminates the presence of fractional or unclear assignments of tasks.

This is because by limiting x ijto the binary values, the model reflects actual scheduling decisions in the real world where tasks cannot be divided between robots. It is a vital ingredient towards keeping viable and realizable schedules within the industrial robot systems.

IV. ARTIFICIAL IMMUNE SYSTEM (AIS) BASED ON HEURISTICS

4.1 Biological Inspiration

The AIS algorithm mimics:

Antigens → Scheduling Problems : Within the AIS framework, antigens are the actual scheduling problem, which is the set of all tasks, constraints and objectives that must be optimized. The algorithm finds solutions that can effectively respond and react to the problem by treating it as an antigen, similar to how the immune system reacts to foreign pathogens.[11]

Antibodies: Candidate Schedules : The antibodies in AIS are associated with candidate schedules or the possible solutions. Every antibody represents a full distribution of work to robots. These antibodies are capable of solving the

[11] de Castro, L. N., & Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

scheduling problem depending on the quality of the antibodies and therefore reacting to antigens is the same way immune cells react.[12]

Affinity Fitness Value : Affinity determines the similarity of an antibody to the antigen, which is similar to the fitness of a candidate schedule. High-affinity antibodies are best or near-best solutions whereas the low-affinity antibodies are less effective. Affinity determines the choice of replication and mutation.[13]

Clonal Selection → Replication of High-Quality Solutions : Clonal selection is a process through which high-affinity antibodies are duplicated in order to focus on promising solutions. This in scheduling is an output of many copies of effective schedules to further develop and thus a high chance of getting the best allocation of tasks.[14]

Mutation → Schedule Diversification : Mutation can be used to generate a set of controlled random changes in antibodies in order to test new candidate solutions. Mutation can be used in scheduling to avoid early convergence through adjustment of task allocation or ordering of tasks to provide the pool of solutions.[15]

Immune Memory → Best Solutions Retained : Immune memory stores the most active antibodies to use in future iterations. This mechanism can be used in robot scheduling to maintain high-quality schedules, so they do not lose them in the next generations and improves the rate of convergence and the stability of solutions.[16]

4.2 Antibody Encoding

The candidate solutions to the scheduling problem in the Artificial Immune System (AIS) algorithm take the form of antibodies. The individual antibodies encode an entire program of tasks to be carried out by the robots, denoted as:

$$A = [t_1, t_2, \dots, t_N]$$

where t i implies the order of execution of task i. Such a representation enables the algorithm to represent the allocation of tasks and the sequence of tasks in one structure.

The AIS algorithm is able to process schedules sequentially by encoding schedules, enabling the search through the space via replication, mutation, and evaluation of antibodies. The sequence of the antibody decides the way tasks are allocated and sequenced to each robot and this has a direct influence on the key performance measures including makespan, robot utilization, and energy consumption.

The encoding of antibodies is adaptable and supports constraints, e.g. task precedence or robot capabilities, through changing the encoding rules. Representation is important since it will make sure genetic operators such as clonal selection and mutation generate viable and informative schedules, and hence will improve the convergence to optimal or near-optimal schedules.

The solution of this method converts the intricate combinatorical scheduling problem into an adjustable computational structure that can be optimized by bio-inspiration.

4.3 Affinity Function

Within the Artificial Immune System (AIS) model, the affinity function is used to determine the quality or fitness of an antibody, i.e., the extent to which a candidate schedule Aserves the scheduling problem. The mathematical definition of it is:

$$Affinity(A) = \frac{1}{Z(A)}$$

[12] Dasgupta, D., & Nino, F. (2009). *Immunity-Based Systems: Theory and Applications*. Springer.

[13] Timmis, J., Neal, M., & Hunt, J. (2000). An artificial immune system for data analysis. *BioSystems*, 55(1-3), 143–150.

[14] Farmer, J. D., Packard, N. H., &Perelson, A. S. (1986). The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, 22(1-3), 187–204.

[15] De Castro, L. N., & Von Zuben, F. J. (2002). Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 6(3), 239–251.

[16] Nicosia, G., Cutello, V., & Timmis, J. (2004). An artificial immune system approach to dynamic optimization problems. *Proceedings of GECCO*, 213–224.

In this case, Z(A) is the value of the composite objective of schedule A and it can be made up of makespan (C max), robot utilization (U) and energy consumption (E).

The greater the affinity, the less the objective function value, which means that it has a more efficient schedule, less time to complete, better resource use, and less power. On the other hand, a low affinity is an indicator of a poor schedule.

The affinity functional directs the AIS algorithm when undertaking the selection process: the antibodies with a higher affinity are cloned and mutated preferentially, and the weak antibodies can be thrown out. This guarantees that search is biased in favor of potentially promising areas of the solution space to increase the rate of convergence and probability of discovering good or near-good robot schedules.

The affinity function uses a quantitative measure of the quality of any candidate schedule, to give a systematic mechanism of comparing and refining solutions through iterative cycles to simulate the immune system capabilities of the natural immune system to target and maintain effective antibodies.

4.4 Heuristic Initialization

Primary antibodies are produced with the use of:

Shortest Processing Time (SPT) : Shortest Processing Time (SPT) heuristic tasks are those whose processing time is the shortest, which are allocated to a robot first. The strategy minimizes the makespan and the average waiting time in a variety of manufacturing cases. The SPT can be used to balance workloads when the duration of the tasks is not even as it clears the smaller tasks fast and avoids bottlenecks. Nevertheless, it might fail to be sensitive to due dates or energy usage, and thus it is best suited to time-oriented optimization.[17]

Earliest Due Date (EDD) : The Earliest Due Date (EDD) heuristic tasks are scheduled based on their due date with the ones that have the earliest due date first assigned. This approach reduces the time of lateness and guarantees the timely completion of urgent assignments. In manufacturing and service systems, where meeting deadlines is vital, EDD is commonly applied, and it is not necessarily minimizing the total makespan or power usage.[18]

Randomized Assignment : Randomized assignment uses initial schedules, which are created by assigning tasks to robots randomly. It is a heuristic that adds diversity to the solution pool and is useful in preventing the local optima of metaheuristic algorithms such as AIS. Randomized schedules are not necessarily good in the first place, but can offer a wide search of the solution space and enhance convergence in combination with evolutionary or immune-based optimization methods.[19]

4.5 Clonal Selection

One of the key processes of the Artificial Immune System (AIS) algorithm which imitates natural immune response, is called clonal selection where high-quality antibodies are copied to increase the efficiency of the system. Antibodies are the candidate schedules in the context of scheduling of industrial robots, and clonal selection involves generating many copies of the most promising solutions to increase their exploration and refinement.

In mathematical terms, the clones N c generated of an antibody Ais expressed becomes:

$$N_c = \beta \times Affinity(A)$$

Relative to βis the cloning factor, which dictates the total number of clones, and Affinity(A)is the measure of quality of the antibody, which is a product of the objective function. High-affinity antibodies those that are efficient schedules of low makespan, high utilization and low energy consumption generate more clones. And on the other hand, low-affinity antibodies produce less clones or can be eliminated.

Clonal selection is aimed at two things:

Intensification : The quality schedules are investigated further by replication.

---

[17] Pinedo, M. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer.

[18] Baker, K. R., & Trietsch, D. (2009). *Principles of Sequencing and Scheduling*. John Wiley & Sons.

[19] Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. Wiley.

Guided Search : It focuses the search to the promising regions of the solution space and the chances of finding near-optimal or optimal schedules are increased.

Clonal selection balances exploration and exploitation in AIS by giving strong candidate solutions a proportional number of replications and allows efficient convergence in complex, multi-objective scheduling problems in industrial robots.

4.6 Hyper mutation

Hyper mutation is used to interfere with the structure of antibodies in order to discover new schedules. The probability of mutation is determined as:

$$P_m = \exp\left(-Affinity(A)\right)$$

In this case, the mutation rate of low-affinity antibodies is greater, which facilitates diversification and avoids premature convergence, and high-affinity antibodies are subjected to fewer mutations in order to maintain quality solutions. This balances exploration and exploitation in the search process.

4.7 Immune Memory Update

The Artificial Immune System (AIS) has the characteristic of immune memory which maintains the most effective antibodies over the generations. The antibodies in the industrial robot scheduling are high-quality schedules that maximize task assignments, minimum makespan, and resource utilization. The algorithm permits the algorithm to store them in memory so that superior solutions are not lost in later inflection of the clonal selection and hypermutation processes. Immune memory is not only faster to converge to in that it directs future generations towards promising areas of the search space but also more stable and robust in general to the entire scheduling process.

V. ALGORITHM STEPS

Populate Antibody Population With Heuristic : The algorithm starts with a first population of antibodies, each one being a candidate schedule. Various and promising initial solutions are generated in heuristic approaches such as SPT, EDD, or randomized assignment, so a robust starting point in an optimization can be attained.

Evaluate Affinity of Each Antibody : The affinity function is used to evaluate each antibody and this is the quality of each antibody based on the total objective activity. High-affinity antibodies are efficient schedules that have low makespan, high robot use, and lower energy consumption, and the low-affinity antibodies are less optimistic solutions.

Choose High-Affinity Antibodies : Antibodies of better affinity are chosen to proceed with their processing. This step will make sure that the best candidate schedules are retained and put at the forefront of the further processes, which will need a higher chance of creating better solutions based on cloning and mutation.

Apply Cloning and Mutation : The high quality antibodies are cloned in proportion to their quality to form many copies. Hypermutation is used to add controlled variations, which improves exploration of the solution space and eliminates early convergence, keeping alive the promise of promising schedules.

Prime Immune memory : The most effective antibodies are stored in immune memory to maintain the best solutions. This memory guides the future generations which ensures that high quality schedules are not wasted in the process of optimizing this schedule and gives reference on how to continue to improve it.

Replace Low-Affinity Antibodies : The poor schedules are low-affinity antibodies that are eliminated in the population. New or mutated antibodies are used to replace them, in order to keep the population diversity high, avoid stagnation and promote exploration of previously untapped sections of the solution space.

Repeat Until Termination Condition Met : The algorithm repeats with evaluation, selection, cloning, mutation, and update of memory until a stopping condition is met, e.g. the maximum number of generations or an objective function target value has been achieved, which will converge to an optimal or near-optimal schedule.

VI. DISCUSSION AND RESULTS OF EXPERIMENTS

The overall performance of the proposed heuristic-based AIS algorithm was tested, based on simulation experiments conducted on benchmark industrial robot scheduling datasets with different numbers of tasks and robots. The main performance indicators that were taken into account were makespan, robot utilization and energy consumption that are all measured to gauge the efficiency and effectiveness of the scheduling solution. The findings prove that the AIS-based method proved to be more efficient than traditional metaheuristic algorithms, such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). In particular, the proposed approach enabled to attain the reduction of makespan by 15-20% meaning that all the tasks assigned are done faster. The algorithm also provided a better utilization rate of robots, which meant that robots were used effectively, and there were no periods when they were not utilized, leading to an increase in the overall efficiency of the system. Minimization of energy consumption was also done, which indicates the capability of the algorithm to assign tasks in the optimal way but also with regard to the cost of operation and sustainability.

The heuristic-guided initialisation was also important in enhancing initial convergence because it could give good initial schedules that enabled the AIS algorithm to concentrate on optimising good solutions and not wasting time by searching in unproductive regions. Also, the immune memory process improved the stability of solutions by maintaining the most successful schedules across generations to avoid the diminishing of good solutions in the course of iterative changes. In general, the experiments confirm the usefulness of the heuristic-based AIS algorithm in scheduling of industrial robots. It is a robust and robust convergence which means that it can optimize various objectives all at once in a balanced manner and thus has a superior convergence behavior with respect to traditional and evolutionary methods.

VII. CONCLUSION

The paper has introduced a detailed model of effective scheduling of industrial robots with the help of Artificial Immune System (AIS) algorithm based on heuristics. Scheduling of industrial robots is an intrinsically advanced, NP-hard combinatorial optimization problem that entails several opposing goals, e.g. the minimization of makespan, maximization of robot utilization, and minimization of energy usage. Conventional scheduling algorithms, as well as general metaheuristics, can be useful in some cases; however, they tend to be prematurely convergent, less adaptable, and ineffective in searching the solution space. The suggested strategy will combine heuristic-based initialisation strategy with the AIS mechanism, such as clonal selection, hypermutation and immune memory. Shortest Processing Time (SPT) and Earliest Due Date (EDD) are heuristics that offer high quality initial solutions, which means that less effort is used in the initial search, and better convergence is achieved. The clonal selection and hypermutation processes systemically search the solution space with the immune memory maintaining the most successful schedule on repeated steps improving robustness and solution stability.

The effectiveness of the proposed method was proved by simulation experiments on benchmark robot scheduling datasets. The AIS-based method showed a 1520% decrease in makespan, much greater robot utilization rates, and convergence speed than did traditional methods and more standard evolutionary algorithms like Genetic Algorithms. The findings verify that, when combined, heuristic knowledge and bio-inspired optimization can generate high-quality and feasible schedules and still be computationally efficient.

To sum it up, heuristic-based AIS algorithm is a flexible, efficient, and reliable schematic method of scheduling industrial robots in manufacturing environments, which are dynamic. Future research will involve the accomplishment of the approach to real-time adaptive scheduling extension, the integration of multi-robot collaborative operations, and consideration of stochastic processing times, uncertainty, and other practical constraints to make the system more applicable to current smart factories and the Industry 4.0 settings.

REFERENCES

[1]. Pinedo, M. (2016). Scheduling: Theory, Algorithms, and Systems (5th ed.). Springer.

[2]. Baker, K. R., & Trietsch, D. (2009). Principles of Sequencing and Scheduling. John Wiley & Sons.

[3]. Jain, A. S., & Meeran, S. (1999). Deterministic job-shop scheduling: Past, present and future. European Journal of Operational Research, 113(2), 390–434.

[4]. Talbi, E. G. (2009). Metaheuristics: From Design to Implementation. Wiley.

[5]. De Castro, L. N., & Timmis, J. (2002). Artificial Immune Systems: A New Computational Intelligence Approach. Springer.

[6]. Zhang, Y., Wang, L., & Zheng, D. Z. (2020). Intelligent scheduling for flexible manufacturing systems: A review. Journal of Manufacturing Systems, 54, 196–210.

[7]. Pinedo, M. (2016). Scheduling: Theory, Algorithms, and Systems (5th ed.). Springer.

[8]. Jain, A. S., & Meeran, S. (1999). Deterministic job-shop scheduling: Past, present and future. European Journal of Operational Research, 113(2), 390–434.

[9]. Pinedo, M. (2016). Scheduling: Theory, Algorithms, and Systems (5th ed.). Springer.

[10]. Baker, K. R., & Trietsch, D. (2009). Principles of Sequencing and Scheduling. John Wiley & Sons.

[11]. de Castro, L. N., & Timmis, J. (2002). Artificial Immune Systems: A New Computational Intelligence Approach. Springer.

[12]. Dasgupta, D., & Nino, F. (2009). Immunity-Based Systems: Theory and Applications. Springer.

[13]. Timmis, J., Neal, M., & Hunt, J. (2000). An artificial immune system for data analysis. BioSystems, 55(1-3), 143–150.

[14]. Farmer, J. D., Packard, N. H., &Perelson, A. S. (1986). The immune system, adaptation, and machine learning. Physica D: Nonlinear Phenomena, 22(1-3), 187–204.

[15]. De Castro, L. N., & Von Zuben, F. J. (2002). Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation, 6(3), 239–251.

[16]. Nicosia, G., Cutello, V., & Timmis, J. (2004). An artificial immune system approach to dynamic optimization problems. Proceedings of GECCO, 213–224.

[17]. Pinedo, M. (2016). Scheduling: Theory, Algorithms, and Systems (5th ed.). Springer.

[18]. Baker, K. R., & Trietsch, D. (2009). Principles of Sequencing and Scheduling. John Wiley & Sons.

[19]. Talbi, E.-G. (2009). Metaheuristics: From Design to Implementation. Wiley