

## Comparative Analysis of Spiral Dynamic Algorithm and Artificial Bee Colony Optimization for Position Control of Flexible Link Manipulators



N. M. Abdul Ghani\*<sup>ID</sup>, N. M. Nasir<sup>ID</sup>, A. A. Abdullah Hashim<sup>ID</sup>

Faculty of Electrical & Electronics Engineering Technology, Universiti Malaysia Pahang Al-Sultan Abdullah, Pekan 26060, Malaysia

Corresponding Author Email: [normaniha@umpsa.edu.my](mailto:normaniha@umpsa.edu.my)

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.570315>

### ABSTRACT

**Received:** 22 March 2024  
**Revised:** 27 May 2024  
**Accepted:** 4 June 2024  
**Available online:** 25 June 2024

#### Keywords:

*PID controller, DC motor, MATLAB, ABC SDA, flexible link manipulator*

Evolutionary algorithms have significantly advanced robotics by enabling the creation of efficient and intelligent robotic systems. This study aims to evaluate the effectiveness of two optimization algorithms, artificial bee colony (ABC) and spiral dynamic algorithm (SDA), in controlling the position of a flexible-link manipulator. By integrating the ABC algorithm into the manipulator's control system, the goal is to enhance its ability to plan paths and optimize trajectories. Additionally, the spiral dynamics algorithm, which draws on principles from complex adaptive systems and human values, provides a framework for modelling system evolution. The study hypothesizes that combining these two algorithms will improve the flexible link manipulator's adaptability and flexibility in dynamic environments and varied task conditions. The results support this hypothesis, demonstrate that the combined ABC and spiral dynamics approach outperforms conventional methods in several key performance metrics, including PID parameter tuning, overshoot, rise time, settling time, and steady-state error. In industry application such a motoring machinery, it is crucial to achieve these metrics at best, which kept the overshoot below 10%, settling time and rise time within a second. Interestingly, the manipulator's behaviours using the spiral dynamics algorithm for PID controller tuning were superior to those using alternative methods. Specifically, the spiral dynamics approach yielded the lowest overshoot at 5.83%, compared to 16.64% with the heuristic method and 8.82% with the ABC method. The SDA has the fastest rise time and settling time which is 0.03267 s and 0.18445 s respectively. Overall, the simulation results indicate that employing these algorithms for PID parameter optimization effectively enhances the manipulator's transient response and minimizes steady-state error, offering promising implications for real-world robotic applications.

## 1. INTRODUCTION

Flexible-link manipulators, also known as one-degree-of-freedom robot arms, are essential robotic systems characterized by their simple structure and ability to move in a single joint. They consist of a base, a single arm segment, and an end effector, making them less complex than multi-link manipulators. Nonetheless, flexible-link manipulators play a vital role in comprehending the fundamental principles of robotics, encompassing kinematics, manipulability, and control. The study of flexible-link manipulators establishes a solid groundwork for exploring more intricate robotic systems and applications. Through the analysis of their kinematics, researchers can gain insights into the interrelationships between joint variables and the position as well as the orientation of the end effector. This knowledge proves essential for tasks like determining the pose of the end effector, calculating trajectories, and developing control strategies. Furthermore, the evaluation of manipulability aids in assessing the reachability and performance of flexible-link manipulators, facilitating optimization for specific applications [1].

Moreover, flexible-link manipulators find practical applications across various domains. Their simplicity and affordability make them well-suited for tasks such as pick-and-place operations, assembly tasks, and material handling. They are frequently utilized in educational settings to introduce students to the fundamentals of robotics, as their uncomplicated structure allows for clear demonstrations and comprehension of key concepts. Additionally, flexible-link manipulators serve as building blocks for more intricate robotic systems, serving as a starting point for studying multi-link manipulators and advanced robotic functionalities [2, 3].

Overall, the study and practical implementation of flexible-link manipulators constitute a critical component of the broader field of robotics. The problem statement is to develop a position control system for a flexible-link manipulator that accurately and efficiently controls the position of the DC motor. The control system should be capable of moving the manipulator to desired positions within a specified workspace while minimizing errors and disturbances and using the algorithm to minimize the step response characteristics. The objective is to compare and analyze the performances of the

optimized controller for position control of a flexible link manipulator. The control system should also consider constraints such as joint limits and actuator limitations while providing smooth and reliable motion control for the flexible-link manipulator [4, 5].

PID controllers are commonly employed in a wide range of control systems to regulate various process variables such as temperature, pressure, speed, and position. These controllers operate by adjusting their output based on the disparity between the desired setpoint and the measured process variable. The proportional, integral, and derivative components of the PID controller each play distinct roles in influencing the control action, working together to achieve the desired response [6]. Tuning PID controllers involves determining optimal values for the controller gains, including the proportional, integral, and derivative constants, to achieve stable and efficient control performance [7]. This tuning process is essential for achieving the desired system behavior, such as fast response, minimal overshoot, and effective disturbance rejection.

To tune the PID controller, an algorithm is employed where the objective function is utilized, and the parameters are adjusted after each independent run. Data is collected for each run, and the results are organized based on the varying parameter categories. The PID controller gains ( $K_p$ ,  $K_i$ , and  $K_d$ ) with the best objective value are selected and used in the simulation for the hub angle of the flexible-link manipulator. Once the simulation is complete, the step response characteristics, including rise time, settling time, steady-state error, and overshoot, are evaluated.

The performance of the hub angle can be assessed by setting specific conditions to analyze the performance criteria. A PID controller tuning method for the flexible-link manipulator system is implemented, outlining the steps involved in utilizing the ABC algorithm and SDA [8].

The selection of ABC and SDA for comparison in PID parameter tuning warrants further justification based on their suitability for the specific optimization task. ABC and SDA were chosen due to their distinct optimization approaches, each offering unique advantages relevant to PID tuning. ABC mimics the foraging behavior of honeybees, employing a combination of exploration and exploitation to search for optimal solutions. While ABC's stochastic nature enables broad exploration of the solution space, it can struggle with convergence speed and get trapped in local optima. However, its ability to balance exploration and exploitation makes it a candidate worth considering for complex optimization problems like PID tuning [9, 10].

Conversely, SDA utilizes a deterministic spiral search pattern, systematically shrinking the search space towards promising regions [11]. This methodical approach often results in faster convergence and more stable solutions, crucial characteristics for PID parameter optimization where precision and efficiency are paramount. Additionally, SDA's adaptability to various problem landscapes and reduced parameter sensitivity simplifies the optimization process, further enhancing its suitability for PID tuning tasks [12-14]. Thus, in this study, the choice of ABC and SDA for comparison in PID parameter tuning is selected by their complementary strengths in exploration, exploitation, convergence speed, and adaptability, offering a comprehensive assessment of optimization algorithms in addressing the complexities of PID tuning to establish the novelty and significance of this work.

The performance of the hub angle optimized by ABC algorithm and SDA was assessed with suitable performance criteria (error) to achieve the best performance of the hub angle [8, 13, 14].

## 2. METHODOLOGY

### 2.1 Heuristic method

A closed-loop control system is one in which the output influences the subsequent input to the process. If the output deviates from the target benchmark, the input is automatically adjusted to achieve the required result. In other words, the input is modified based on the output's response. The illustration depicts the block diagram of a closed loop system. The controller for this system is a PID controller. The circuit diagram of DC motor position control based PID controller is shown in Figure 1.

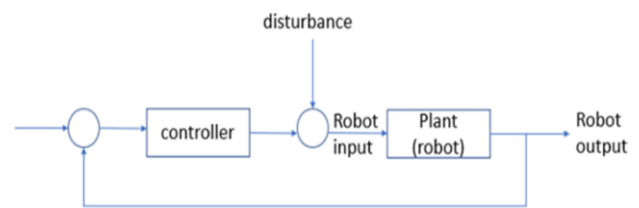


Figure 1. Block diagram of closed loop system

Tuning the PID controller manually entails adjusting three parameters: setting the reset time to infinity, the rate to zero, and the gain to a level that induces the loop to oscillate consistently [15, 16]. Depending on the system's response time, a higher gain may be initially chosen for quicker error correction, with a slight increase preferred for slower responses. Subsequently, the gain is reduced by half the incremental amount, and the reset time is fine-tuned until the error is eliminated. Once overshoot is minimized, the rate of the PID loop can be gradually increased [17].

### 2.2 ABC algorithm

The ABC algorithm, short for artificial bee colony algorithm, is a metaheuristic optimization technique inspired by the foraging behavior of honeybee colonies. It is widely utilized for solving complex optimization problems, including the tuning of PID (Proportional-Integral-Derivative) control systems. The ABC algorithm adopts a swarm-inspired approach to optimize the gains of PID controllers. It mimics the collaborative behavior of honeybees in a colony, where they communicate and share information to collectively search for the best food sources. In the context of the ABC algorithm, candidate PID controller gains are represented as artificial bees, which explore the solution space to find the optimal set of gains. The algorithm operates through three main phases: employed bees, onlooker bees, and scout bees [18].

During the employed bee phase, each bee generates a new solution by modifying its current solution and evaluates its performance using a fitness function that reflects the behavior of the control system. Onlooker bees select solutions based on their fitness values and perform a dance-like mechanism to guide the search towards promising regions. Scout bees randomly explore uncharted territory to prevent the algorithm from getting stuck in local optima. Through iterative processes

and information exchange among bees, the ABC algorithm aims to converge towards the optimal set of PID controller gains that result in the desired control performance. Employed bees share information about the best food sources with onlooker bees, who then use this information to select promising solutions [19]. The flowchart of ABC algorithm is shown in Figure 2.

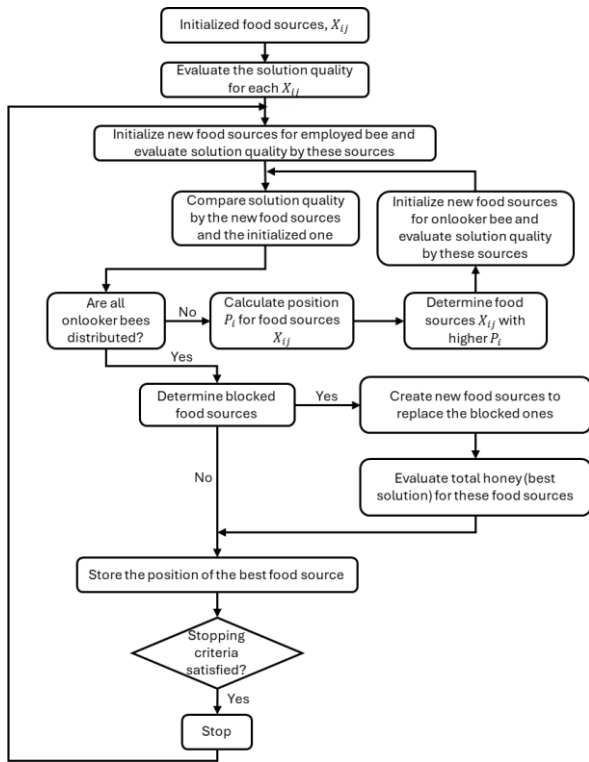


Figure 2. Flow chart ABC

Mathematically, the exploitation processes are performed by employed and Onlooker’s bees by Eq. (1), the selection condition performed by scout bees represented in Eq. (2) as:

$$X_{ij} = x_{ij} + \theta_{ij}(x_{ij} - x_{kj}) \quad (1)$$

$$x_{ij}^{rand} = x_{ij}^{min} + rand(0,1)(x_{ij}^{max} - x_{ij}^{min}) \quad (2)$$

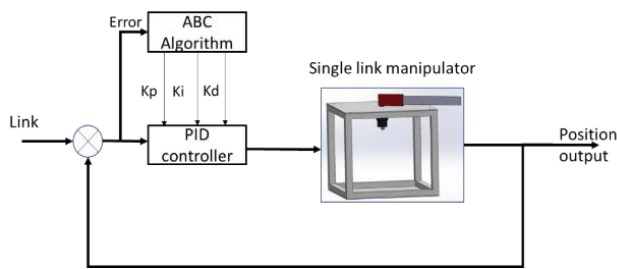


Figure 3. PID parameter tuning block diagram of reformative ABC

The employed and onlookers used same Eq. (1) for exploitation process which cannot guaranty for finding optimal solution. The researchers improved typical ABC algorithm by different strategies such as global and guided bees. The ABC algorithm is used to optimize the PID parameters, which is a parameter optimization problem based on a certain objective function, i.e., finding the optimal values in the parameter space of  $Kp$ ,  $Ki$ , and  $Kd$  variables to optimize

the control performance of the system. the control block diagram is shown in Figure 3.

### 2.3 Spiral dynamic algorithm

Spiral dynamics optimization leverages the insights and principles of the spiral dynamics framework to enhance optimization algorithms and problem-solving processes. By considering the developmental stages, values, and cultural evolution, it aims to create more inclusive, adaptable, and effective optimization approaches that reflect the diverse range of human perspectives and motivations [20, 21].

The SDA algorithm introduces a unique opportunity for agents to explore optimal solutions in locations opposite to their current positions. This strategy enhances the dispersion of the searching agents across the predefined search space, thereby improving the exploration and exploitation capabilities of the SDA. Utilizing a spiral equation, the SDA operates as an agent-based algorithm, facilitating the smooth movement of agents from one location to another. The spiral equation, denoted as Eq. (3), is a fundamental component of the SDA algorithm.

$$x_i(t + 1) = S(r, \theta)x_i(t) - (S(r, \theta) - I)x^*(t) \quad (3)$$

The detailed optimized input-output scaling parameters are depicted in Table 1, while Figures 4 and 5 show the block diagram of the SDA.

Table 1. Detailed optimized input output scaling parameters

Symbols	Description
$\theta_{i,j}$	Search point angular displacement or rotational angle on $x_i - x_j$ plane around point of origin
$r$	Spiral radius or convergence rate
$m$	Total number of search points
$k_{max}$	Maximum iteration number
$x_j(k)$	Position of $i_{th}$ point in $k_{th}$ generation
$R^n$	Composition of rotational $n \times n$ matrix based on combination of all two axes

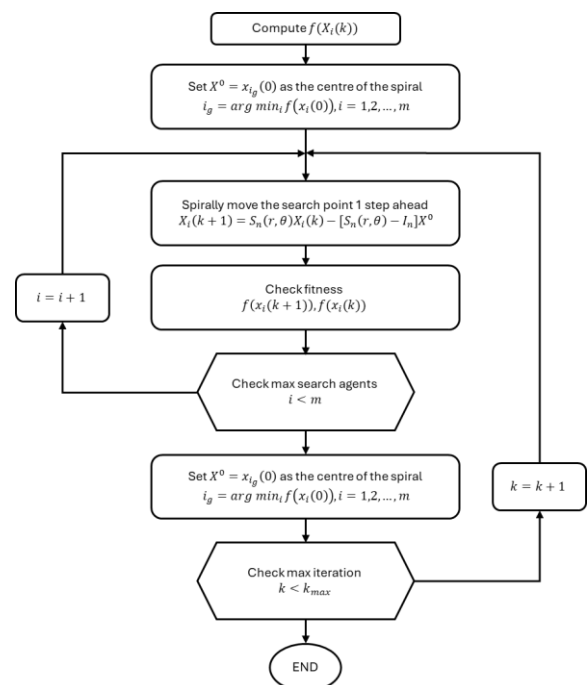


Figure 4. Flowchart of SDA

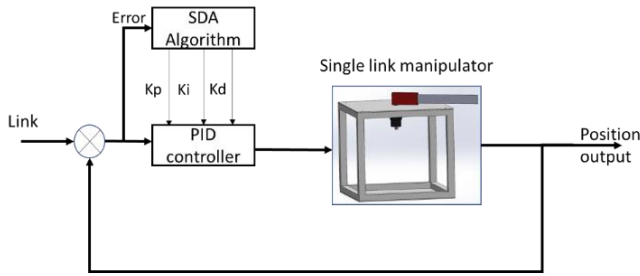


Figure 5. Block diagram of SDA

### 3. SIMULATION AND RESULT ANALYSIS

This project aimed to determine the optimal parameters ( $K_p$ ,  $K_i$ , and  $K_d$ ) by employing the ABC and SDA algorithms, leading to a remarkable response rate. The algorithms were utilized to identify the appropriate PID controller parameters. The combination of  $K_p$ ,  $K_i$ , and  $K_d$  parameters can effectively enhance the system's response and minimize various performance parameters in the time domain. This includes reducing parameters such as stability time ( $T_s$ ), rise time ( $T_r$ ), maximum overshoot (%OS), and steady-state error ( $e_{ss}$ ), thereby improving system performance.

#### 3.1 Convergence test

The proposed methods, ABC and SDA were both used to define the potency of meta-heuristic approached to optimize the parameters of PID control in closed-loop system. The algorithms success can be determined by analyzing the convergence test which were defined as the success rate of minimizing error after multiple iteration of algorithms process in tuning the PID constants ( $K_p$ ,  $K_i$ , and  $K_d$ ). Figure 6 illustrates the comparison of ABC and SDA by their convergence rate.

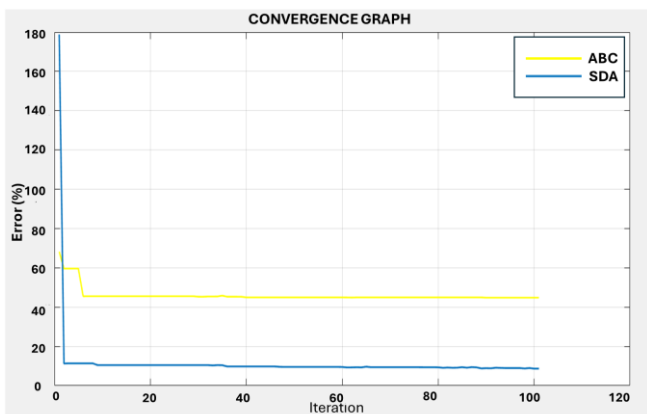


Figure 6. Convergence graph comparison

Table 2. Convergence rate

Meta-Heuristic	ABC	SDA
Convergence's rate of error	42.50%	8.89%

Figure 6 and Table 2 illustrate that the ABC algorithm often struggles with local optima, resulting in an error performance of 42.50%. In contrast, the SDA algorithm significantly outperforms ABC, achieving an error performance of only 8.89%. This difference in performance can be attributed to the

SDA's unique spiral search method, which effectively narrows down the search space and homes in on promising areas, leading to faster and more reliable convergence. On the other hand, the ABC algorithm shows that increasing the number of food sources can reduce the objective value because a larger number of solutions are being evaluated. While this extensive search can eventually improve results, it also means that the algorithm takes more time to process and find the best solution. As a result, although the ABC algorithm can improve its performance with more food sources, it tends to do so more slowly and can still get stuck in suboptimal solutions. In contrast, SDA's structured search pattern allows it to achieve better accuracy more quickly.

#### 3.2 Transient analysis

Table 3 presents the PID controller parameters that were optimized for each performance criterion using the respective method. These optimized parameters are then implemented in the PID controller to assess the optimal performance of the flexible-link manipulator.

Table 3. PID Parameters

	$K_p$	$K_i$	$K_d$
Heuristic	1.14121	15.8238	0.02599
ABC	9.96787	3.86539	0.44742
SDA	21.58463	13.47670	0.02091

The performance of the flexible-link manipulator is evaluated by analyzing the step response of the hub angle in the flexible link manipulator system. The step response characteristics encompass metrics such as rise time, settling time, steady state error, and overshoot. The responses were observed and compared based on the transient response of the three methods. Figure 7 illustrates the transient response of the step response injected into the flexible-link manipulator. Table 4 records the value of those responses.

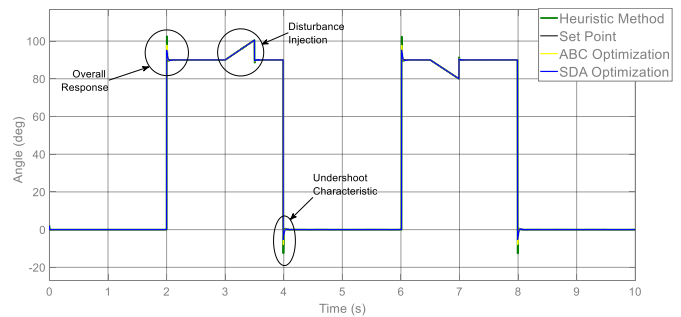


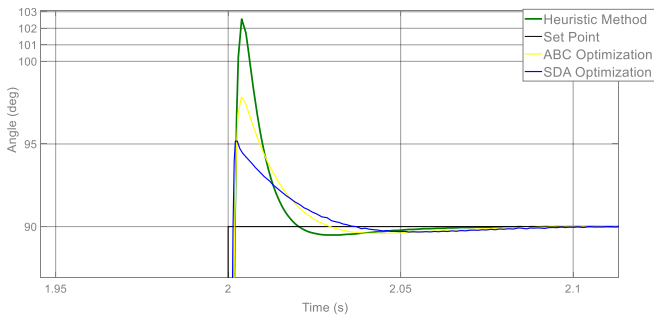
Figure 7. Angular position comparison

Table 4. Transient responses

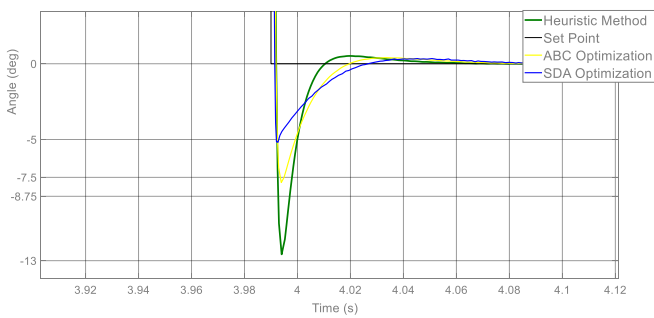
	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady State Error (%)
Heuristic	0.09578	0.51834	16.64	0.81
ABC	0.05589	0.23174	8.82	0.49
SDA	0.03267	0.18445	5.83	0.42

Figure 7 is the comparison of time response characteristic exhibits by heuristic approach, ABC optimization, and SDA optimization to the original responses. Figure 8 is the overall view of the transient response where the recorded data of the response as in Table 4. Figure 9 is the viewing area to study

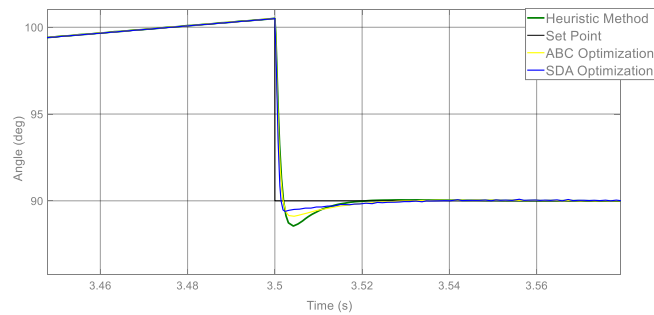
the undershoot characteristics of the system during reverse angular position of the system which can be further observe its value in Table 5. Lastly Figure 10 is the viewing area to test the system robustness against disturbance of ramp input and its analysis recorded in Table 6.



**Figure 8.** Overall transient responses



**Figure 9.** Undershoot characteristics



**Figure 10.** Robustness from disturbances

**Table 5.** Undershoot characteristics

	Rise Time (s)	Settling Time (s)	Undershoot (%)
Heuristic	0.00127	0.13664	13.98
ABC	0.00146	0.11655	8.71
SDA	0.00076	0.09984	5.76

**Table 6.** Robustness from disturbance

	Rise Time (s)	Settling Time (s)	Undershoot (%)
Heuristic	0.00153	0.01035	1.61
ABC	0.00109	0.00199	0.99
SDA	0.00092	0.00182	0.68

The SDA-based PID controller exhibits slightly faster hub angle response in comparison to the heuristic method and ABC. Numerical analysis of the hub angle response as in Table 4

further confirms that the proposed SDA optimization yields faster settling time and rise time when compared to the responses obtained through the ABC algorithm and heuristic method.

Table 4, Table 5, and Table 6 have consistency results in proving SDA as the superior algorithm. The data collected from the observation shows the step response characteristics for the heuristic method, ABC and SDA for different number of foods evaluated in the system. The SDA has the shortest rise time among all the method the performance criteria that evaluated in the system. The settling time and steady state error for the SDA shows the most well performance among all methods. The heuristic method poses the longest settling time which causes the time taken for the system to reach the steady state error to become slower, thus not suitable for application that requires high stability. The performance for the SDA optimization shows the shortest settling time which coherently influences the shorter time taken for the system to reach the steady state error.

The overshoot of the hub angle of the flexible-link manipulator is that response exceeds the steady error. SDA achieves the lowest overshoot, among all the performance criteria that were evaluated. In contrast, the highest overshoot is recorded in the heuristic method which is 16.64% while the ABC shows 8.82%. The SDA shows a better result of 5.83% of overshooting compared to the other two methods. Similarly, during reverse angle operation, the undershoot of the system under the tested controller’s algorithms shows that the SDA achieved the lowest undershoot as shown in Table 5.

The system also has undergone an injection of disturbance in the form of ramp input to test how well the system performs, which determines the system robustness when applied with controllers. As in Table 6, it was shown again that the SDA manages to outperform heuristic approach and ABC algorithm.

The faster hub angle response exhibited by the SDA-based PID controller not only signifies quicker settling and rise times but also hints at a more agile system capable of swift adjustments in angular position. This agility is paramount in dynamic environments where prompt responses are necessary for optimal performance. Moreover, the significant reduction in overshoot and undershoot achieved by the SDA highlights its ability to mitigate oscillations and ensure smoother transitions between states, indicating enhanced stability and precision in control. The observed improvements in system robustness against disturbances, particularly with the SDA, underscore its robust control capabilities, crucial for real-world applications where external factors can influence system dynamics unpredictably. Additionally, the superior performance of the SDA in benchmark function tests suggests its efficacy not only in this specific application but also potentially in broader optimization contexts. Overall, these findings not only validate the superiority of the SDA algorithm but also shed light on the intricate interplay between optimization techniques, controller design, and system behavior, offering valuable insights for future research and application development in control systems engineering.

The results of benchmark function tests revealed a significant improvement in the accuracy of the proposed algorithm, particularly in the case of the SDA. When addressing the PID control problem, it was observed that both PID controllers optimized by the proposed and original algorithms effectively controlled the flexible manipulator. However, the opposing SDA demonstrated a superior control response compared to the ABC.

#### 4. CONCLUSIONS

This paper focuses on optimizing the PID controller parameters for a flexible-link manipulator using the ABC and SDA algorithms. The SDA method demonstrates superior performance with the lowest overshoot of 5.83%, compared to 16.64% for the heuristic method and 8.82% for ABC. SDA also has the fastest rise time at 0.03267s, while the heuristic method and ABC show 0.09578s and 0.05589s, respectively. The settling time for SDA is 0.18445s, significantly better than 0.51834s for the heuristic method and 0.23174s for ABC. Both SDA and ABC have a steady-state error of 0.49%, compared to 0.81% for the heuristic method. These results indicate that the SDA algorithm outperforms the other methods in minimizing overshoot and steady-state error while achieving faster response times.

The demonstrated superiority of the SDA algorithm suggests significant potential applications in robotics, including surgical robots, automated assembly lines, and space exploration, where precision and efficiency are critical. Additionally, the algorithm's effectiveness can benefit autonomous vehicles, drones, and industrial process control systems.

Future research could explore hybridizing SDA with other optimization techniques or machine learning to enhance performance further. Applying SDA to different manipulators or control systems and validating its real-world performance could broaden its applicability. Investigating the algorithm's scalability in multi-objective optimization problems is another promising direction. These avenues can fully harness the potential of the SDA algorithm in control system optimization.

#### REFERENCES

- [1] Akyuz, I.H., Yolacan, E., Ertunc, H.M., Bingul, Z. (2011). PID and state feedback control of a single-link flexible joint robot manipulator. In 2011 IEEE International Conference on Mechatronics, Istanbul, Turkey, pp. 409-414. <https://doi.org/10.1109/ICMECH.2011.5971320>
- [2] Zouari, F., Saad, K.B., Benrejeb, M. (2013). Adaptive backstepping control for a single-link flexible robot manipulator driven DC motor. In 2013 International Conference on Control, Decision and Information Technologies (CoDIT), Hammamet, Tunisia, pp. 864-871. <https://doi.org/10.1109/CoDIT.2013.6689656>
- [3] Shukor, A.Z.H., Miskon, M.F., Ali, F., Nor, M.K.M. (2015). Position control of single link underwater robot manipulator. *Jurnal Teknologi*, 74(9): 169-173. <https://doi.org/10.11113/jt.v74.4822>
- [4] Dharavath, P.K., Ohri, J. (2020). Position and vibration control of single link flexible joint manipulator. In 2020 First IEEE International Conference on Measurement, Instrumentation, Control and Automation (ICMICA), Kurukshetra, India, pp. 1-6. <https://doi.org/10.1109/ICMICA48462.2020.9242807>
- [5] Ullah, H., Malik, F.M., Raza, A., Saeed, A., Mazhar, N., Khan, R. (2020). Robust nonlinear output feedback tracking control of single-link flexible joint robotic manipulator system. In 2020 14th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, pp. 1-6. <https://doi.org/10.1109/ICOSST51357.2020.9333022>
- [6] Samiei, S.K., Mirzaei, M., Yarinia, N., Rafatnia, S. (2022). Design and practical implementation of an input-constrained nonlinear controller for a single-link flexible joint robotic manipulator. In 2022 10th RSI International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, pp. 551-556. <https://doi.org/10.1109/ICRoM57054.2022.10025095>
- [7] Zuo, K., Wang, D. (1992). Closed loop shaped-input control of a class of manipulators with a single flexible link. In Proceedings 1992 IEEE International Conference on Robotics and Automation, Nice, France, pp. 782-787. <https://doi.org/10.1109/ROBOT.1992.220274>
- [8] Abachizadeh, M., Yazdi, M.R.H., Yousefi-Koma, A. (2010). Optimal tuning of PID controllers using artificial bee colony algorithm. In 2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Montreal, QC, Canada, pp. 379-384. <https://doi.org/10.1109/AIM.2010.5695861>
- [9] Wang, H., Du, H., Cui, Q., Song, H. (2022). Artificial bee colony algorithm based PID controller for steel stripe deviation control system. *Science Progress*, 105(1): 00368504221075188. <https://doi.org/10.1177/00368504221075188>
- [10] Du, H., Liu, P., Cui, Q., Ma, X., Wang, H. (2022). PID controller parameter optimized by reformative artificial bee colony algorithm. *Journal of Mathematics*, 2022(1): 3826702. <https://doi.org/10.1155/2022/3826702>
- [11] Hashim, M.R., Tokhi, M.O. (2016). Enhanced chaotic spiral dynamic algorithm with application to controller design. In 2016 IEEE International Conference on Power and Energy (PECon), Melaka, Malaysia, pp. 752-756. <https://doi.org/10.1109/PECON.2016.7951659>
- [12] Rizal, N.A.M., Mohammad, S., Abd Razak, A.A., Jusof, M.F.M., Nasir, A.N.K., Ahmad, M.A., Hashim, A.I. (2019). Hybrid sine-spiral dynamic algorithm for dynamic modelling of a flexible manipulator. In 2019 19th International Conference on Control, Automation and Systems (ICCAS), Jeju, Korea (South), pp. 120-125. <https://doi.org/10.23919/ICCAS47443.2019.8971498>
- [13] Nasir, A.N.K., Ahmad, M.A., Tokhi, M.O. (2021). Hybrid spiral-bacterial foraging algorithm for a fuzzy control design of a flexible manipulator. *Journal of Low Frequency Noise, Vibration and Active Control*, 41(1): 340-358. <https://doi.org/10.1177/14613484211035646>
- [14] Lara-Molina, F.A., Gonçalves, R.S. (2023). Reliability-based optimization of flexible manipulators. *Journal of Vibration Engineering & Technologies*, 11(7): 3147-3162. <https://doi.org/10.1007/s42417-022-00737-z>
- [15] Al-khazarji, H.A.H., Abdulsada, M.A., Abduljabbar, R.B. (2020). Robust approach of optimal control for DC motor in robotic arm system using matlab environment. *International Journal on Advanced Science, Engineering and Information Technology*, 10(6): 2231.
- [16] Abut, T. (2016). Modeling and optimal control of a DC motor. *International Journal of Engineering Trends and Technology*, 32(3): 146-150.
- [17] Suman, S.K., Giri, V.K. (2016). Speed control of DC motor using optimization techniques based PID Controller. In 2016 IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, India, pp. 581-587. <https://doi.org/10.1109/ICETECH.2016.7569318>
- [18] Pareek, S., Kishnani, M., Gupta, R. (2014). Application of artificial bee colony optimization for optimal PID

- tuning. In 2014 International Conference on Advances in Engineering & Technology Research (ICAETR - 2014), Unnao, India, pp. 1-5. <https://doi.org/10.1109/ICAETR.2014.7012817>
- [19] Shah, H., Tairan, N., Mashwani, W.K., Al-Sewari, A.A., Jan, M.A., Badshah, G. (2017). Hybrid global crossover bees algorithm for solving boolean function classification task. In Intelligent Computing Methodologies: 13th International Conference, ICIC 2017, Liverpool, UK, pp. 467-478. [https://doi.org/10.1007/978-3-319-63315-2\\_41](https://doi.org/10.1007/978-3-319-63315-2_41)
- [20] Ghani, N.A., Nasir, A.K., Tokhi, M.O. (2014). Optimization of fuzzy logic scaling parameters with spiral dynamic algorithm in controlling a stair climbing wheelchair: Ascending task. In 2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR), Miedzydroje, Poland, pp. 776-781. <https://doi.org/10.1109/MMAR.2014.6957454>
- [21] Razak, A.A.A., Nasir, A.N.K., Jusof, M.F.M., Mohammad, S., Rizal, N.A.M. (2019). Opposition based spiral dynamic algorithm with an application to a PID control of a flexible manipulator. In 2019 9th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, pp. 126-130. <https://doi.org/10.1109/ICCSCE47578.2019.9068556>