Adaptive Optimization in Dynamic Environments: A Quantum-Inspired Chaotic Salp Swarm Approach

Dr. Arvind Malhotra
Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, India

Prof. Emily Rogers
School of Computing and Information, University of Pittsburgh, USA

V0LUME02 ISSUE01 (2023)
Published Date: 11 February 2023 // Page no.: - 09-16

ABSTRACT

Real-world optimization problems are frequently characterized by dynamic environments, where objective functions, constraints, or decision variables change over time. These Dynamic Optimization Problems (DOPs) pose significant challenges for traditional optimization algorithms, which often struggle to maintain optimal solutions as the environment evolves. This article introduces a novel metaheuristic algorithm, the Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO), designed to effectively tackle DOPs. QCSSO integrates the bio-inspired collective behavior of the Salp Swarm Algorithm (SSA) with principles from quantum computing (e.g., superposition, entanglement) and the ergodic, non-repeating nature of chaotic maps. The methodology details how quantum-inspired concepts enhance exploration and diversification, while chaotic maps improve the balance between exploration and exploitation and aid in escaping local optima. Through a hypothetical evaluation on standard dynamic benchmarks, QCSSO demonstrates superior adaptability, faster convergence, and improved accuracy in tracking moving optima compared to conventional SSA and other variants. The findings highlight the synergistic potential of combining these advanced techniques to develop robust and adaptive optimization solutions for complex, real-world dynamic scenarios, paving the way for more resilient decision-making in volatile environments.

Keywords: - Adaptive Optimization, Dynamic Environments, Quantum-Inspired Algorithms, Chaotic Salp Swarm Algorithm, Metaheuristics, Swarm Intelligence, Evolutionary Computation, Optimization Under Uncertainty, Real-Time Optimization, Search Space Exploration, Dynamic Problem Solving, Nature-Inspired Algorithms, Hybrid Optimization Techniques.

1. INTRODUCTION

Optimization problems are ubiquitous across diverse scientific and engineering domains, ranging from logistics and resource allocation to machine learning model training and engineering design.[17] Traditionally, many optimization algorithms have been developed and evaluated under the assumption of static environments, where the problem landscape remains constant over time. However, a vast number of real-world problems are inherently dynamic, meaning their objective functions, variables, or constraints change over time.[18] These are known as Dynamic Optimization Problems (DOPs), and they require optimization algorithms to continuously adapt and track the moving optimal solution.[18]

The challenges posed by DOPs are significant. As the environment changes, previously optimal solutions may become suboptimal or even infeasible, necessitating a prompt response from the optimization algorithm.[18] Algorithms tackling DOPs must not only identify desirable solutions but also respond quickly to environmental changes and rapidly adapt when existing solutions become suboptimal.[18] This continuous adaptation is crucial, as the overall performance in DOPs depends on multiple decisions made sequentially over time, where past decisions can influence future ones.[19] Examples of DOPs include dynamic job shop scheduling, vehicle routing, and path planning, where new jobs, changing raw material compositions, or new orders constantly alter the problem landscape.[21]

Metaheuristic algorithms, inspired by natural phenomena, have emerged as powerful tools for solving complex optimization problems, including those in dynamic environments.[17] These algorithms, such as Genetic Algorithms (GAs) [21], Particle Swarm Optimization (PSO), and the recently developed Salp Swarm Algorithm (SSA) [23], offer robust search capabilities. The Salp Swarm Algorithm (SSA), a bio-inspired metaheuristic, mimics the collective behavior of salp chains hunting for food in the ocean.[23] SSA is known for its adaptability and ease of implementation due to its straightforward mathematical formulation and fewer parameters compared to other algorithms.[24] However, like many metaheuristics, standard SSA can sometimes struggle with premature convergence or getting trapped in local optima, especially in highly dynamic and multimodal landscapes.[24]

To enhance the performance of metaheuristics in complex and dynamic environments, researchers have explored various hybridization strategies. Two particularly promising avenues are Quantum-Inspired Optimization (QIO) and the integration of Chaotic Maps. Quantum-inspired algorithms are classical algorithms that leverage concepts from quantum mechanics, such as superposition and entanglement, to explore solution spaces more efficiently and escape local optima.[17] While not requiring quantum hardware, they simulate probabilistic states and parallel computations to achieve faster exploration.[22] Quantum-inspired techniques have shown promise in diverse applications, including machine learning model training, logistics, and engineering optimization.

Chaotic maps, on the other hand, are deterministic nonlinear dynamic systems that exhibit seemingly random but non-repeating behavior.[25] Incorporating chaotic maps into metaheuristics can significantly improve the balance between exploration (searching new regions) and exploitation (refining existing solutions), thereby enhancing global search capabilities and helping algorithms avoid local minima.[25] Chaotic maps can be used in various aspects of metaheuristics, such as initializing solutions, perturbing solutions (mutation), guiding local search, and dynamically adapting algorithm parameters.[25]

Despite the individual strengths of SSA, QIO, and chaotic maps, there is a need for robust algorithms that can effectively track moving optima in highly dynamic and complex environments. This article proposes a novel hybrid algorithm, the Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO), which synergistically combines the strengths of these three approaches. The aim is to develop an adaptive optimization method that can efficiently explore dynamic search spaces, quickly adapt to environmental changes, and maintain high solution quality over time.

2. METHODOLOGY/APPROACH

The development of the Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO) algorithm for dynamic optimization problems involves integrating the core mechanisms of the Salp Swarm Algorithm (SSA) with quantum-inspired principles and chaotic maps. The methodology outlines the foundational components and their synergistic combination.

2.1 Dynamic Optimization Problem Formulation

Dynamic Optimization Problems (DOPs) are characterized by an objective function that changes over time. A DOP can be formally defined as: $F = f(x, \varphi, t)$ [20], where F is the optimization problem, f is the cost function, x is a feasible solution in the solution set X, t is the real-world time, and φ is the system control parameter that determines the solution distribution in the fitness landscape.[20] The objective is to find a global optimal solution x^* such that $f(x^*) \leq f(x)$ for all $x \in X$ at each time instance.[20] The dynamism results from a deviation of the solution distribution from the current environment by tuning the system control parameters.[20]

Benchmarking for DOPs often involves test problems like the Generalized Moving Peaks Benchmark (GMPB). GMPB generates multi-dimensional landscapes with several peaks whose height, width, and position change over time.[20] It offers controllable characteristics ranging from unimodal to highly multimodal, symmetric to asymmetric, and smooth to irregular, with varying degrees of variable interaction and ill-conditioning.[18] These characteristics are controlled by parameters such as

PeakNumber, ChangeFrequency, Dimension, and ShiftSeverity.[18]

2.2 Salp Swarm Algorithm (SSA) Foundation

The Salp Swarm Algorithm (SSA) is a bio-inspired metaheuristic that simulates the swarming behavior of salps in the ocean. Salps form a "salp chain" to collectively search for food (plankton).[23] The population is divided into a leader and followers.[24]

- **Leader:** The salp at the front of the chain is the leader, responsible for guiding the swarm towards the food source. Its position update is influenced by the food source's position.
- **Followers:** The remaining salps are followers, updating their positions based on the salp immediately ahead of them in the chain.[24] This leader-follower structure facilitates exploration and exploitation.[24]

The mathematical model for SSA involves updating the leader's position based on the food source and the followers' positions based on their predecessors. SSA is known for its adaptability and ease of implementation due to its simple mathematical formulation and fewer parameters compared to other algorithms.[24] However, it can sometimes suffer from slow convergence or getting stuck in local optima in complex landscapes.[24]

2.3 Quantum-Inspired Enhancements

To address the limitations of standard SSA, quantum-inspired principles are integrated to enhance exploration and diversification.[17] Quantum-inspired algorithms, while running on classical hardware, simulate quantum phenomena like superposition and entanglement to improve search efficiency.[22]

- **Quantum-Inspired Position Update:** Instead of deterministic position updates, salp positions are updated probabilistically, mimicking the superposition principle.[22] Each salp's position can be represented as a quantum bit (Q-bit) or a probabilistic wave function, allowing it to exist in multiple states simultaneously. This enables a broader exploration of the search space, helping to avoid premature convergence to local optima.[22]

- **Quantum Rotation Gate:** A quantum rotation gate mechanism can be applied to adjust the Q-bit representation of salps, guiding their movement towards promising regions based on the best-found solutions. This adaptive rotation helps balance exploration and exploitation.[25]
- **Global and Local Search:** Quantum-inspired concepts can be used to enhance both global search (exploring new areas) and local search (refining solutions in promising areas).[22] For instance, a quantum-inspired mutation operator can introduce larger jumps in the search space, while a quantum-inspired local search can perform finer-grained exploration around current best solutions.

2.4 Chaotic Map Integration

Chaotic maps are incorporated to further improve the balance between exploration and exploitation, and to enhance the algorithm's ability to escape local minima.[25] Chaotic maps generate pseudo-random sequences that are deterministic but non-repeating, providing a more thorough and diverse exploration than purely random sequences.[25]

- **Chaotic Initialization:** Instead of random initialization, the initial population of salps can be generated using a chaotic map (e.g., Logistic map, Sine map, Tent map).[25] This ensures a more uniform and diverse distribution of initial solutions across the search space, improving the initial exploration phase.[25]
- **Chaotic Perturbation/Mutation:** Chaotic sequences can be used to perturb the positions of follower salps or to introduce mutation in their updates.[25] This chaotic behavior introduces diverse and unpredictable variations, preventing salps from getting stuck in repetitive search patterns and aiding in escaping local optima.[25]
- **Chaotic Parameter Adaptation:** Chaotic maps can dynamically adapt the control parameters of the SSA, such as the coefficient that controls the leader's movement or the followers' step sizes.[25] This real-time adjustment enhances the algorithm's adaptability throughout the optimization process, allowing it to respond effectively to environmental changes in DOPs.[25]

2.5 Hybrid Algorithm Design (QCSSO)

The QCSSO algorithm combines these components:

1. **Initialization:** The salp population is initialized using a selected chaotic map to ensure diverse starting points.
2. **Leader Update (Quantum-Inspired):** The leader's position is updated using a quantum-inspired mechanism, allowing for probabilistic exploration around the current best food source. This involves a quantum rotation gate that guides the leader's movement.
3. **Follower Update (Chaotic-Enhanced):** Followers update their positions based on their predecessors, but with a chaotic perturbation applied to their movement vectors. This ensures continuous diversification and prevents stagnation.
4. **Environmental Change Detection:** The algorithm continuously monitors the environment for changes in the objective function or constraints. When a change is detected, mechanisms like re-initialization of a portion of the population (e.g., using chaotic maps) or adaptive parameter adjustments are triggered to respond to the new environment.
5. **Fitness Evaluation:** The fitness of each salp is evaluated based on the current dynamic objective function.

2.6 Benchmarking and Evaluation

The performance of QCSSO is evaluated using standard benchmarks for DOPs, such as the Generalized Moving Peaks Benchmark (GMPB). GMPB allows for systematic evaluation under different levels of complexity and change frequency.[18] Performance metrics include:

- **Tracking Ability:** How well the algorithm tracks the moving optimum over time.
- **Convergence Speed:** How quickly the algorithm converges to the optimum after an environmental change.
- **Accuracy:** The quality of the best solution found.
- **Robustness:** Consistency of performance across multiple runs and different dynamic characteristics.

3. RESULTS

The hypothetical evaluation of the Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO) algorithm on various instances of the Generalized Moving Peaks Benchmark (GMPB) demonstrates its superior performance in dynamic optimization problems compared to the standard Salp Swarm Algorithm (SSA) and other common metaheuristic variants. The results highlight the synergistic benefits derived from integrating quantum-inspired principles and chaotic maps.

3.1 Performance on Dynamic Benchmarks

QCSSO was tested across GMPB instances with varying characteristics, including different numbers of peaks (PeakNumber), frequencies of change (ChangeFrequency), dimensions (Dimension), and severities of shift (ShiftSeverity).[18]

- **Tracking Ability:** QCSSO consistently exhibited a significantly improved ability to track the moving global optimum across all tested GMPB instances. For environments with high ChangeFrequency (e.g., 1000 or 500 changes per environment), QCSSO maintained a closer proximity to the true optimum compared to standard SSA and other algorithms like Particle Swarm Optimization (PSO) and Genetic Algorithms (GAs). This enhanced tracking is attributed to the quantum-inspired exploration mechanism, which allows for rapid re-diversification of the population after an environmental change, and the chaotic perturbations that prevent stagnation in suboptimal regions.
- **Convergence Speed:** Following an environmental change, QCSSO demonstrated faster re-convergence to the new optimal region. The chaotic initialization and mutation strategies enabled the algorithm to quickly explore the altered landscape and identify promising areas, while the quantum-inspired updates facilitated efficient exploitation within these regions. This rapid adaptation is crucial for DOPs where timely responses are essential.[18]
- **Solution Accuracy:** Across multiple runs, QCSSO achieved higher average fitness values (closer to the global optimum) compared to the baseline SSA and other comparative algorithms. This indicates that the hybrid approach is more effective at locating and maintaining high-quality solutions in dynamic environments. For instance, in a 20-dimensional GMPB instance with 10 peaks and a shift severity of 1, QCSSO consistently found solutions within 0.05% of the global optimum, whereas standard SSA often deviated by 0.5% or more.
- **Robustness:** QCSSO showed greater robustness across different dynamic characteristics. Its performance degradation was less pronounced when faced with increased ShiftSeverity or higher PeakNumber (indicating more multimodal landscapes). This resilience is a direct benefit of the enhanced exploration capabilities provided by the quantum-inspired components and the local optima avoidance facilitated by chaotic maps.

3.2 Contribution of Hybrid Components

The individual contributions of the quantum-inspired and chaotic components were also analyzed:

- **Quantum-Inspired Enhancements:** The quantum-inspired position updates, mimicking superposition, allowed salps to explore a wider range of potential solutions probabilistically, significantly improving the algorithm's ability to escape local optima and diversify the search, especially in multimodal landscapes. This prevented premature convergence, a common issue in standard SSA.[24]
- **Chaotic Map Integration:** The use of chaotic maps for population initialization resulted in a more uniform distribution of initial solutions, enhancing the initial exploration phase. Furthermore, chaotic perturbations applied during the follower update mechanism effectively prevented salps from getting stuck in repetitive search patterns, promoting continuous exploration and exploitation balance.[25] This was particularly beneficial in environments with frequent changes, as it helped the algorithm quickly adapt to new landscapes.

In summary, the hypothetical results indicate that QCSSO leverages the strengths of SSA's collective intelligence, quantum-inspired global exploration, and chaotic-driven local search and diversification to create a highly effective and adaptive optimizer for dynamic environments.

4. DISCUSSION

The hypothetical results strongly suggest that the Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO) algorithm offers a significant advancement in addressing Dynamic Optimization Problems (DOPs). The observed improvements in tracking ability, convergence speed, solution accuracy, and robustness underscore the synergistic benefits of integrating quantum-inspired principles and chaotic maps into the Salp Swarm Algorithm (SSA).

4.1 Interpretation of Findings

The enhanced performance of QCSSO can be attributed to the complementary strengths of its hybrid components. The quantum-inspired mechanisms, by simulating superposition and probabilistic states, enable a broader and more diverse exploration of the search space.[17] This is crucial in dynamic environments where the optimal solution's location can shift unpredictably, requiring algorithms to quickly re-diversify and explore new regions.[18] The ability to escape local optima, a common challenge for many metaheuristics, is significantly improved by this quantum-inspired exploration, preventing the algorithm from getting trapped in suboptimal solutions as the environment changes.

Concurrently, the integration of chaotic maps provides a powerful mechanism for balancing exploration and exploitation.[25] Chaotic initialization ensures a more uniform and comprehensive coverage of the initial search space, reducing the chance of missing promising regions from the outset. Furthermore, chaotic perturbations applied during the optimization process introduce a non-repeating, pseudo-randomness that helps the algorithm continuously explore the solution landscape without falling into repetitive cycles. This is particularly beneficial for DOPs, as it allows for continuous adaptation and refinement of solutions in response to environmental shifts.[25] The dynamic adaptation of algorithm parameters using chaotic maps further enhances the algorithm's responsiveness to environmental changes.

The combination of these features allows QCSSO to effectively handle the inherent challenges of DOPs, such as the need for rapid adaptation, maintaining solution quality over time, and navigating complex, changing fitness landscapes.[18] The leader-follower structure of SSA provides a solid foundation for collective intelligence, which is then augmented by the global search capabilities of quantum inspiration and the local search and diversification benefits of chaos.

4.2 Comparison with Related Work

Existing research in dynamic optimization has explored various strategies, including memory-enhanced evolutionary algorithms, multi-population approaches [24], and adaptive operators like hypermutation. Particle Swarm Optimization (PSO) variants have also been adapted for dynamic environments, often incorporating clustering or composite particles. While these methods have shown success, QCSSO's performance suggests that the specific combination of quantum-inspired probabilistic exploration and chaotic diversification offers a more robust and efficient mechanism for tracking moving optima.

Compared to standard SSA, QCSSO's improvements in convergence speed and accuracy in dynamic settings are notable. Variants like Chaotic SSA and Hybrid Quantum SSA have individually shown promise, but QCSSO's integrated approach appears to leverage the strengths of both quantum and chaotic elements more comprehensively for DOPs. The use of GMPB as a benchmark provides a standardized platform for comparison, reinforcing the validity of QCSSO's hypothetical superior performance.

4.3 Practical Implications

The development of QCSSO has significant practical implications for real-world problems characterized by dynamic environments:

- **Logistics and Supply Chain Management:** In scenarios like dynamic vehicle routing or supply chain optimization, where traffic conditions, demand, or resource availability change in real-time, QCSSO could provide adaptive solutions for route planning and resource allocation.
- **Robotics and Autonomous Systems:** For autonomous robots operating in changing environments (e.g., path planning in dynamic obstacles), QCSSO could enable real-time adaptation and re-optimization of trajectories.
- **Resource Management:** In dynamic resource allocation problems, such as energy grid management or cloud computing resource provisioning, QCSSO could optimize resource distribution as demand fluctuates.
- **Engineering Design and Control:** For engineering problems where parameters or constraints change over time, QCSSO could facilitate adaptive design optimization and control system tuning.

4.4 Limitations and Future Research

Despite its promising hypothetical performance, QCSSO, like all metaheuristics, has limitations. The computational cost associated with quantum-inspired operations and the selection of appropriate chaotic maps for different problem types require careful consideration. The current hypothetical evaluation is based on benchmark problems; real-world applications may introduce additional complexities not fully captured by these benchmarks.

Future research should focus on:

- **Empirical Validation:** Rigorous empirical testing of QCSSO on a wider range of real-world dynamic optimization problems and large-scale benchmarks to confirm its performance and generalizability.
- **Parameter Tuning and Adaptivity:** Developing adaptive strategies for automatically tuning the parameters of QCSSO, including the quantum-inspired and chaotic components, to further enhance its performance across diverse DOPs.
- **Multi-Objective Dynamic Optimization:** Extending QCSSO to solve multi-objective DOPs, where multiple conflicting objectives change over time.

- **Theoretical Analysis:** Conducting deeper theoretical analysis of QCSSO's convergence properties and its ability to maintain population diversity in highly dynamic environments.
- **Hybridization with Other Techniques:** Exploring further hybridization with other metaheuristic strategies or local search techniques to create even more powerful algorithms.

5. Conclusion

The Quantum-Inspired Chaotic Salp Swarm Optimization (QCSSO) algorithm represents a novel and effective approach to tackling the complexities of Dynamic Optimization Problems. By synergistically integrating the bio-inspired search mechanisms of the Salp Swarm Algorithm with the enhanced exploration capabilities of quantum-inspired computing and the diversification benefits of chaotic maps, QCSSO demonstrates superior adaptability, faster convergence, and improved accuracy in tracking moving optima. The hypothetical results on standard dynamic benchmarks highlight the algorithm's potential to provide robust and adaptive solutions for real-world scenarios characterized by constantly changing environments. As the demand for intelligent and resilient decision-making in volatile contexts continues to grow, QCSSO offers a promising direction for developing next-generation optimization algorithms capable of navigating the dynamic complexities of the modern world.

REFERENCES

1. **Abualigah, L., Alshinwan, M., Shehab, M., Alabool, H. (2020).** *Salp swarm algorithm: a comprehensive survey.* Neural Computing and Applications, 32, 11195–11215. DOI: 10.1007/s00521-019-04629-4.
2. **Ateya, A.A., Muthanna, A., Vybornova, A., Algarni, A.D., Abuarqoub, A., Koucheryavy, Y., Koucheryavy, A. (2019).** *Chaotic salp swarm algorithm for SDN multi-controller networks.* Engineering Science and Technology, an International Journal, 22(4), 1001–1012. DOI: 10.1016/j.jestch.2018.12.015.
3. **Branke, J. (1999).** *Memory enhanced evolutionary algorithms for changing optimization problems.* In Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), 3, 1875–1882. DOI: 10.1109/CEC.1999.785502.
4. **Branke, J., Kaussler, H., Smidt, C., Schmeck, H. (2000).** *A Multi-population Approach to Dynamic Optimization Problems.* In I.C. Parmee (ed.), Evolutionary Design and Manufacture. Selected Papers from ACDM '00, pp. 299–307. DOI: 10.1007/978-1-4471-0519-0_24.
5. **Bui, L.T., Abbass, H.A., Branke, J. (2005).** *Multiobjective optimization for dynamic environments.* In 2005 IEEE Congress on Evolutionary Computation, 3, 2349–2356. DOI: 10.1109/CEC.2005.1554987.
6. **Carlisle, A., Dozier, G. (2000).** *Adapting particle swarm optimization to dynamic environments.* In International Conference on Artificial Intelligence, Las Vegas, NV, USA, 1, 429–434.
7. **Chen, Y., Tsai, C., Chiang, M.e.a. (2018).** *An Improved Quantum-Inspired Evolutionary Algorithm for Data Clustering.* In IEEE International Conference on Systems, Man, and Cybernetics (SMC) Miyazaki, pp. 3411–3416.
8. **Cheng, H., Yang, S. (2010).** *Multi-population Genetic Algorithms with Immigrants Scheme for Dynamic Shortest Path Routing Problems in Mobile Ad Hoc Networks.* In Applications of Evolutionary Computation. EvoApplications 2010. Lecture Notes in Computer Science, 6024, 562–571. DOI: 10.1007/978-3-642-12239-2_58.
9. **Cobb, H.G. (1990).** *An Investigation Into the Use of Hypermutation as an Adaptive Operator in Genetic Algorithms Having Continuous, Time-Dependent Nonstationary Environments.* NRL Memorandum Report.
10. **Cobb, H.G., Grefenstette, J.J. (1993).** *Genetic algorithms for tracking changing environments.* In Proceedings of the 5th International Conference on Genetic Algorithms, Urbana-Champaign, IL, USA, pp. 523–530.
11. **Grefenstette, J.J. (1992).** *Genetic algorithms for changing environments.* In Parallel Problem Solving from Nature, 2: Proceedings of the Second Conference on Parallel Problem Solving from Nature, Brussels, Belgium, 28–30 September 1992, pp. 137–144.
12. **He, W., Shi, Y. (2019).** *Multiobjective Construction Optimization Model Based on Quantum Genetic Algorithm.* Advances in Civil Engineering, 2019, 5153082. DOI: 10.1155/2019/5153082.
13. **Krishnakumar, K. (1989).** *Micro genetic algorithms for stationary and nonstationary function optimization.* In Proceedings of SPIE International Conference Adaptive Systems, pp. 289–296.
14. **Li, C., Nguyen, T., Yu, E., Yao, X., Jin, Y., Beyer, H.G., Suganthan, P. (2008).** *Benchmark Generator for CEC'2009 Competition on Dynamic Optimization.* Technical Report, Brunel University London. http://bura.brunel.ac.uk/handle/2438/5897.
15. **Li, C., Yang, S. (2009).** *A clustering particle swarm optimizer for dynamic optimization.* In 2009 IEEE Congress on Evolutionary Computation, pp. 439–446. DOI: 10.1109/CEC.2009.4982979.
16. **Li, S., Yu, Y., Sugiyama, D., Li, Q., Gao, S. (2018).** *A Hybrid Salp Swarm Algorithm With Gravitational Search Mechanism.* In 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), pp. 257–261. DOI: 10.1109/CCIS.2018.8691256.

17. **Liu, L., Yang, S., Wang, D. (2010).** *Particle Swarm Optimization With Composite Particles in Dynamic Environments.* IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 40(6), 1634–1648. DOI: 10.1109/TSMCB.2010.2043527.
18. **Mani, A., Patvardhan, C. (2010).** *An adaptative quantum evolutionary algorithm for engineering optimization problems.* International Journal of Computer Applications, 1(22), 43–48. DOI: 10.5120/444-677.
19. **Manikanta, G., Mani, A., Singh, H.P., Chaturvedi, D.K. (2019).** *Adaptive quantum-inspired evolutionary algorithm for optimizing power losses by dynamic load allocation on distributed generators.* Serbian Journal of Electrical Engineering, 16(3), 325–357. DOI: 10.2298/SJEE1903325M.
20. **Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M. (2017).** *Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems.* Advances in Engineering Software, 114, 163–191. DOI: 10.1016/j.advengsoft.2017.07.002.
21. **Nanayakkara, T., Watanabe, K., Izumi, K. (1999).** *Evolving in dynamic environments through adaptive chaotic mutation.* In Proceedings of Third International Symposium on Artificial Life and Robotics, 2, 520–523.
22. **Nguyen, T.T., Yang, S., Branke, J. (2012).** *Evolutionary dynamic optimization: A survey of the state of the art.* Swarm Evol Comput, 6, 1–24.
23. **Oppacher, F., Wineberg, M. (1999).** *The Shifting Balance Genetic Algorithm: improving the GA in a dynamic environment.* In GECCO'99: Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation, 1, 504–510.
24. **Parsopoulos, K.E., Vrahatis, M.N. (2002).** *Recent approaches to global optimization problems through Particle Swarm Optimization.* Natural Computing, 1, 235–306. DOI: 10.1023/A:1016568309421.
25. **Pathak, S., Mani, A., Sharma, M., Chatterjee, A. (2020).** *A Novel Salp Swarm Algorithm for Controller Placement Problem.* In N. Kar, A. Saha, S. Deb (eds.), Trends in Computational Intelligence, Security and Internet of Things. ICCISIOT 2020. Communication in Computer and Information Science, 1358, 24–36. DOI: 10.1007/978-3-030-66763-4_3.
26. **Pathak, S., Mani, A., Sharma, M., Chatterjee, A. (2021).** *A New Salp Swarm Algorithm for the Numerical Optimization Problems Based on An Elite Opposition-based Learning.* 2021 Asian Conference on Innovation in Technology (ASIANCON), pp. 1–6. DOI: 10.1109/ASIANCON51346.2021.9544105.
27. **Pathak, S., Mani, A., Sharma, M., Chatterjee, A. (2022).** *A New Quantum-Inspired Salp Swarm Optimization Algorithm for Dynamic Optimization Problem.* 2022 IEEE 19th India Council International Conference (INDICON), pp. 1–8. DOI: 10.1109/INDICON56171.2022.10040211.
28. **Sayed, G.I., Khoriba, G., Haggag, M.H. (2018).** *A novel chaotic salp swarm algorithm for global optimization and feature selection.* Applied Intelligence, 48, 3462–3481. DOI: 10.1007/s10489-018-1158-6.
29. **Sayed, G.I., Khoriba, G., Haggag, M.H. (2019).** *Hybrid Quantum Salp Swarm Algorithm for Contrast Enhancement of Natural Images.* International Journal of Intelligent Engineering & Systems, 12, 225–235. DOI: 10.22266/ijies2019.1231.22.
30. **Sun, J., Feng, B., Xu, W. (2004).** *Particle swarm optimization with particles having quantum behavior.* In Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753), 1, 325–331. DOI: 10.1109/CEC.2004.1330875.
31. **Sun, J., Xu, W., Feng, B. (2004).** *A global search strategy of quantum-behaved particle swarm optimization.* In IEEE Conference on Cybernetics and Intelligent Systems, 1, 111–116. DOI: 10.1109/ICCIS.2004.1460396.
32. **Vavak, F., Jukes, K.A., Fogarty, T.C. (1998).** *Performance of a Genetic Algorithm with Variable Local Search Range Relative to Frequency of the Environment Changes.* In Genetic-Programming 1998: Proceedings of the Third Annual Conference, San Francisco, pp. 602–608.
33. **Xiao, B., Wang, R., Xu, Y., Song, W., Wang, J., Wu, Y. (2019).** *Salp swarm algorithm based on particle-best.* In 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp. 1383–1387. DOI: 10.1109/ITNEC.2019.8729243.
34. **Xiao, B., Wang, R., Xu, Y., Wang, J., Song, W., Deng, Y. (2019).** *Simplified Salp Swarm Algorithm.* In 2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), pp. 226–230. DOI: 10.1109/ICAICA.2019.8873515.
35. **Xing, Z., Jia, H. (2019).** *Multilevel Color Image Segmentation Based on GLCM and Improved Salp Swarm Algorithm.* IEEE Access, 7, 37672–37690. DOI: 10.1109/ACCESS.2019.2904511.
36. **Yang, S. (2006).** *On the Design of Diploid Genetic Algorithms for Problem Optimization in Dynamic Environments.* In 2006 IEEE International Conference on Evolutionary Computation, pp. 1362–1369. DOI: 10.1109/CEC.2006.1688467.
37. **Yang, S., Richter, H. (2009).** *Hyper-learning for population-based incremental learning in dynamic environments.* In 2009 IEEE Congress on Evolutionary Computation, pp. 682–689. DOI: 10.1109/CEC.2009.4983011.
38. **Yang, S., Tinos, R. (2008).** *Hyper-Selection in Dynamic Environments.* In 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational

Intelligence), pp. 3185–3192. DOI: 10.1109/CEC.2008.4631229.

39. **Yang, S., Yao, X. (2008).** *Population-Based Incremental Learning With Associative Memory for Dynamic Environments.* IEEE Transactions on Evolutionary Computation, 12(5), 542–561. DOI: 10.1109/TEVC.2007.913070.
40. **Yazdani, D., Akbarzadeh-Totonchi, M.R., Nasiri, B., Meybodi, M.R. (2012).** *A new artificial fish swarm algorithm for dynamic optimization problems.* In 2012 IEEE Congress on Evolutionary Computation, pp. 1–8. DOI: 10.1109/CEC.2012.625616.