ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 17, Issue, 12, pp.35708-35710, December, 2025
DOI: https://doi.org/10.24941/ijcr.49946.12.2025

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

GREY WOLF OPTIMIZER APPROACHES FOR RESOURCE ALLOCATION IN CLOUD COMPUTING

[1]Sonia Sharma and [2]Dr. Nipun Chhabra

[1]Research Scholar, School of Engineering Design and Automation, GNA University, Phagwara, Punjab, India
[2]Associate Professor, School of Engineering Design and Automation, GNA University, Phagwara, Punjab, India

ARTICLE INFO

Article History:
Received 19th June, 2025
Received in revised form
15th August, 2025
Accepted 31st October, 2025
Published online 30th December, 2025

Keywords:

Grey Wolf Optimizer, Cloud Computing, Resource Allocation, Metaheuristic Algorithms, Task Scheduling, QoS.

***Corresponding author:* Sonia Sharma**

ABSTRACT

Cloud computing's capacity to offer scalable, flexible, and on-demand infrastructure services has fundamentally altered the delivery of IT resources. To achieve quality of service (QoS), reduce expenses, and boost productivity, operations must allocate IT resources effectively. The Gray Wolf Optimizer (GWO) and other nature-inspired metaheuristic algorithms have shown great promise in handling difficult resource allocation problems. In this work, we analyze several GWO-based approaches to resource allocation in cloud computing by looking at algorithm architectures, variations, performance results, and benefits and drawbacks. The assessment's primary focus is on important factors like service quality, scalability, cost, power consumption, and longevity. The results show that the baseline GWO and a set of traditional metaheuristics are outperformed by the hybrid and multi-objective GWO versions.

Citation: Sonia Sharma and Dr. Nipun Chhabra. 2025. "Grey wolf optimizer approaches for resource allocation in cloud computing.". ***International Journal of Current Research, 17, (12), 35708-35710.***

INTRODUCTION

Cloud computing enables pay-as-you-go access to shared computing resources (CPU, memory, storage, and network) via the Internet. Real-time analytics, enterprise systems, Internet of Things (IoT) operations, and other scalable applications are supported by this architecture (Mell and Grance, 2011). Effective resource allocation techniques that strike a balance between cost, power consumption, and performance are becoming more crucial as cloud services expand. Large search spaces, numerous resources, and competing objectives make resource allocation in cloud computing intrinsically a challenging optimization problem (Buyya *et al*., 2010). In order to meet QoS requirements like throughput, response time, and SLA compliance while minimizing execution time (runtime), cost, and energy consumption, a scheduling system must dynamically assign virtual machines (VMs), containers, or computing instances to incoming tasks (Zhang *et al*., 2010). Conventional deterministic techniques, like first-come, first-served and circular analysis, are quick and easy, but they can produce subpar outcomes, particularly in large-scale, multi-objective situations (Calheiros *et al*., 2011). Because metaheuristic optimization techniques offer high-quality but approximative solutions to NP-hard problems, they are becoming more and more popular as a substitute. Particle swarm optimization (PSO), ant colony optimization (ACO), differential evolution (DE), and genetic algorithms (GA) are a few of these. Nevertheless, early convergence, scalability, and the requirement for substantial parameter tuning are issues that many metaheuristics must deal with (Talbi, 2009). Mirjalili *et al*., (2014) created the Gray Wolf Optimizer (GWO), an algorithm that mimics the cooperative hunting style and leadership structure of gray wolves (Canis lupus). It is appropriate for dynamic resources due to its adaptability, balanced trade-off between exploration and exploitation, and straightforward structure.

By continuously adjusting the exploration agents' locations based on the first three wolves, effective exploration of the problem space is carried out. GWO, including variants like hybrid GWO (e.g.) and multi-objective GWO (MOGWO). (g). GWO-PSO), is utilized for allocating cloud resources in a growing number of studies. However, the majority of studies concentrate on particular methods rather than conducting systematic comparisons. This study assesses several popular GWO-based resource allocation strategies' performance, design, benefits, and drawbacks. In addition to having useful ramifications for cloud resource management, this comparison shows when and why particular versions perform better.

Cloud Resource Allocation Context: One of the most important aspects of cloud resource allocation is the effective distribution of workloads among available computing resources. System throughput, response time, profitability, and energy consumption are all impacted by efficient distribution (Singh and Chana, 2016). The primary objective is to meet SLA requirements, minimize fines for infractions, and optimize resource utilization. Cloud size, real-time requirements, and various hardware configurations all have an impact on this complexity. Additionally, resource allocation may be multi-objective, requiring the simultaneous optimization of multiple metrics (like cost and performance). One of the biggest challenges is making allocation decisions fast and often in real-time to adjust to shifting workload patterns and user needs. Effective computational techniques that can manage uncertainty are needed for this. Metaheuristic approaches offer methods for making decisions and navigating vast and unexpected search environments.

Use of Grey Wolf Optimizer: GWO is distinct in that it develops a theoretical balance between exploitation (local cleanup) and exploration (global research) using mathematical concepts inspired by

Technique / Study	Type	Optimization Objective(s)	Key Strengths	Weaknesses / Limitations	Performance Metrics Improved
Basic GWO (Mishra & Mishra, 2018)	Standard	Makespan, SLA	Simple, low parameter tuning	Limited multi-objective support	Makespan, SLA adherence
GWO-PSO Hybrid (Wang *et al.*, 2019)	Hybrid	Makespan, cost	Better exploration-exploitation	Increased complexity	Makespan, cost
Fuzzy GWO (Shukla & Srivastava, 2020)	Hybrid	Multi-objective (QoS, energy)	Adaptive parameter control	Fuzzy rules require design	QoS, energy efficiency
MOGWO (Pare *et al.*, 2017)	Multi-objective	Cost, makespan, energy	Efficient Pareto front generation	Complex tradeoff management	Cost, energy, makespan
ML-Assisted GWO (Chen *et al.*, 2022)	Predictive + GWO	Workload prediction + allocation	Proactive adaptation	Needs training data	QoS, energy
Energy-Aware GWO (Lee & Zomaya, 2012)	Objective-weighted	Energy consumption	Reduced energy usage	May sacrifice makespan	Energy consumption
Cloud-Edge GWO (Kumar & Singh, 2021)	Distributed	Latency, cost	Handles edge scenarios	Network overhead	Latency, cost
GWO with GA Operators (Singh & Verma, 2018)	Hybrid	Diversity & convergence	Avoids local optima	More parameters	Success rate, makespan

wolves' hunting behavior. These are converted into equations that update locations, including hunting, the environment, and animal attacks, using adaptive control parameters. Because of its ability to prevent premature convergence and its ease of configuration (few parameters to tune), GWO is a popular option for cloud optimization problems. But there are some disadvantages to basic GWO, like the possibility of slower convergence in more complicated or dynamic situations. Researchers have expanded GWO to support multi-objective optimization or combined it with other methods (hybrid models) to get around this issue.

Grey Wolf Optimizer (GWO): Algorithmic Overview: The Gray Wolf Optimizer is a metaheuristic algorithm that simulates the hunting style of a pack of gray wolves.Potential solutions are ranked in the leaderboard in the following order: Alpha (best), Beta (second best), Delta (third best), and Omega (remaining solutions).

Position updates reminiscent of pack hunting are influenced by Alpha, Beta, and Delta wolves.

Key components:

- **Loot Environment:** Shows possible modifications near optimal locations.
- **Hunting:** Motivated by the most promising solutions.
- **Attack (Exploitation):** A more concentrated search around optima that have been found.
- **Search:** Random search is used to avoid local optimization.
- GWO automatically adjusts the position of each search agent based on the coefficient vectors A and C to balance exploration and exploitation.

The final position is determined using the first three answers (α, β, δ).

GWO-Based Resource Allocation Techniques in Cloud Computing

The literature presents various approaches to manage resource allocation based on GWO. These include:

- **Basic GWO:** Assign tasks directly to resources using Basic GWO.
- **Hybrid GWO:** Combine with other optimization techniques to improve efficiency.
- **Multi-objective GWO (MOGWO):** Extend GWOtooptimize multiple objectives simultaneously.
- **Predictive/ML-assisted GWO:** GWO uses machine learning to make judgments.
- **Energy-Aware GWO:** Fitness features include energy reduction.

Comparative Analysis

Major GWO-based resource allocation strategies are contrasted using typical performance measures.

Discussion of Comparative Outcomes

Makespan: Hybrid versions like GWO-PSO and GWO with GA operators frequently achieve lower makespan because of enhanced search space browsing.

Cost Optimization: Multi-objective techniques like MOGWO are very good at reducing cost without sacrificing performance since they explicitly include cost when determining fitness.

Energy Efficiency: Green cloud computing can benefit from fuzzy and energy-aware GWO approaches since they are more efficient at reducing power use.

Scalability: Although they have a greater computational cost, machine learning-integrated predictive GWO models exhibit flexibility in response to shifting workloads.

Challenges and Future Research

Even though GWO versions are superior to many traditional approaches, a number of problems remain:

- Real-time adaptation. Cloud systems need flexible, real-time GWO.
- Security boundaries: Optimization models rarely include privacy and security goals.
- Heterogeneous environments: Extreme heterogeneity and federation between clouds/resources requires further research.
- Standard benchmarks: The lack of regular benchmark data sets makes fair comparisons difficult.
- Future research should focus on GWO integrated with deep learning, security-aware scheduling, and federated cloud optimization.

CONCLUSION

The design, functionality,benefits, and drawbacks ofGWO-basedresource allocation techniques in cloud computing were examined in this study. The findings demonstrate that in terms of delivery time, cost, energy efficiency, and scalability, hybrid and multi-objective variations of GWO perform better than baseline GWO and many conventional metaheuristics. GWO is still a versatile and efficient optimization method, particularly when paired with hybrid algorithms and adaptive mechanisms. To implement diverse cloud systems and real-time decision making, more research is required.

REFERENCES

Buyya, R., Broberg, J., & Goscinski, A. (2010). *Cloud computing: Principles and paradigms.* Wiley.

Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C.A.F., Buyyaet, R. (2011). Cloud Sim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience, 41*(1), 23-50.

Chen, L., Huang, X., & Wang, G. (2022). *IEEE Transactions on Cloud Computing, 10*(1), 123-137.

Kumar, R., & Singh, D. (2021). GWO-based task offloading. *Journal of Cloud Networking, 8*(2), 45-60.

Lee, Y. C., & Zomaya, A. Y. (2012). Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing, 60*(2), 268-280.

Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. *NIST SP 800-145.*

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software, 69*, 46-61.

Mishra, S., & Mishra, A. (2018). Dynamic VM placement using GWO. *International Journal of Cloud Applications & Computing.4,23-30.*

Pare, P., Singh, A., & Verma, K. (2017). Task scheduling in cloud using GWO. *International Journal of Advanced Research in CS, 8*(5), 34-42.

Shukla, A., & Srivastava, S. (2020). Fuzzy GWO. *Journal of Intelligent & Fuzzy Systems, 39*(1), 321-337.

Singh, A., & Verma, P. (2018). Hybrid GWO with GA. *International Journal of Cloud Computing, 9*(1), 99-112.

Singh, S. K., & Chana, I. (2016). QoS-aware autonomic resource management. *ACM Computing Surveys, 48*(3), 1-46.

Talbi, E.G. (2009). *Metaheuristics: From design to implementation.* Wiley.

Wang, J., Li, W., & Ren, Z. (2019). Hybrid GWO-PSO. *Applied Soft Computing, 78*, 20–32.

Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud Computing: State-Of-The-Art and Research Challenges. *Journal of Internet Services and Applications, 1*(1), 7-18.

****