Metaheuristic Optimization of Node Localization in Wireless Sensor Networks for Tsunami Detection

Vankaramani Amogh Bhushan
Dept. of Information Technology
Vardhaman College of Engineering
Hyderabad, India
amoghv952@gmail.com

Kandakurthi Lokesh
Dept. of Information Technology
Vardhaman College of Engineering
Hyderabad, India
lokeshkandakurthi@gmail.com

Sheshivardhan Reddy
Dept. of Information Technology
Vardhaman College of Engineering
Hyderabad, India
sheshivardhan984@gmail.com

Kottu Santosh Kumar
Dept. of Information Technology
Vardhaman College of Engineering
Hyderabad, India
santoshkumar1208@gmail.com

***Abstract*— Wireless Sensor Networks (WSNs) frequently face challenges in locating detector bumps directly while conserving limited energy coffers. This work explores how three well- known optimization styles inheritable Algorithm (GA), flyspeck mass Optimization (PSO), and Grey Wolf Optimizer (GWO) can ameliorate localization in WSN surroundings. A simulation model was created to observe how different figures of anchor bumps affect network performance. Several parameters, including energy use, outturn, packet delivery, detention, jitter, continuance, connectivity, content, and delicacy, were recorded for analysis. Among the tested algorithms, the GWO system showed more harmonious localization delicacy and balanced energy operation under varying network sizes. The study highlights that mass- grounded optimization can give a dependable path toward erecting energy-effective and adaptive WSNs suitable for long- term deployment.**

Keywords—Wireless sensor networks, Node localization, Metaheuristic algorithms, Grey Wolf Optimizer (GWO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA)

I. Introduction

Wireless Sensor Networks (WSNs) have become one of the most practical solutions for large-scale environmental observation and real-time event monitoring. In coastal and marine regions, these networks can play a vital role in early tsunami detection by gathering underwater or near-shore data that indicate sudden changes in ocean activity [1]. Accurate node localization is a key component in such systems, since the position of each sensor directly influences data reliability, routing efficiency, and event response time [2]. Without proper localization, even high-quality sensing data can lose meaning, as the spatial context of the information becomes uncertain. Traditional localization methods, including range-based and range-free approaches, often face challenges such as multipath fading, signal interference, and limited battery power [3]. These issues become more severe in large-scale or aquatic environments, where direct measurement is difficult. To overcome such problems, metaheuristic algorithms have been widely adopted due to their ability to handle nonlinear and multi-dimensional optimization problems efficiently [4]. Among these techniques, the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Grey Wolf Optimizer (GWO) have shown promising results in optimizing node positions and minimizing localization errors [5–7]. For example, Liu et al. [5] combined GA and PSO to improve the classical DV-Hop localization approach, while Yang et al. [6] proposed an improved PSO model that enhanced global search ability for WSN localization. More recent studies have demonstrated that GWO and its improved variants outperform traditional methods by achieving higher localization precision and energy balance [7, 8]. In underwater and coastal applications, researchers such as Salem Jeyaseelan et al. [9] extended the GWO framework for underwater WSNs and showed that it maintained stability under fluctuating ocean conditions. Similarly, Cui et al. [10] introduced a multi-disturbance strategy GWO to enhance search exploration and accuracy in dynamic environments.

In this study, GA, PSO, and GWO algorithms are implemented and compared to optimize node localization in a WSN model designed for tsunami detection. The experimental setup evaluates how different anchor node configurations influence energy consumption, throughput, delay, jitter, coverage, and localization accuracy. The analysis highlights that GWO offers a more consistent trade-off between localization precision and energy efficiency, demonstrating its potential for robust and energy-aware WSN deployment in early tsunami warning systems.

II. Literature Survey

A. Tackling PSO's Limits in Localization.

When it comes to WSN localization, Particle Swarm Optimization (PSO) has been a popular starting point for a while. It's easy to see why: it's pretty straightforward to get running and often finds a decent solution quickly. Its whole idea is simple, with a "swarm" of digital particles hunting for the best spot, remembering their own best find and the group's best find. But that simplicity is also its main problem. In a really messy, complex problem-like figuring out where dozens of sensors are, standard PSO can get stuck on a "good enough" answer that isn't the real best one. So, most of the recent research isn't about using standard PSO, but about fixing that problem. That paper by Yang et al. [6], for instance, tried to make it smarter by giving it a "self-adjusting" inertia. This just means the swarm is programmed to explore a lot at the beginning, then really focus in on a promising area later on. It's a way to stop it from getting stuck too early. Tariq et al. [7] did something different and, honestly, pretty clever: they combined PSO with a neural network (a GRNN). They let PSO do the first "rough guess" of where the sensors are. Then, they fed that guess to the GRNN, which, having been trained on noisy signal data, could clean it up and make a much more accurate final prediction.

B. The Evolutionary Angle: Using Genetic Algorithms

Then you have Genetic Algorithms (GA), which come at the problem from a totally different angle. Instead of a swarm, GA borrows ideas from biology selection, crossover, and mutation. This makes it fantastic at exploring a huge, unknown territory to find a truly global-best answer, not just a local one. In the WSN world, people have used this to crack some tough localization nuts. That hybrid model from Liu et al. [5] is a perfect example. They didn't just use GA or PSO; they used both to make the classic DV-Hop localization method better.

Why both? Because the two algorithms are good at different things. GA is great at exploring all over the map, and PSO is great at quickly zeroing in on a promising spot. Together, they create a really nice balance. Their results proved it worked, and they got better accuracy and, just as importantly, cut down on network chatter, which is a huge deal for battery life. This idea of mixing and matching is getting really popular. You see other papers [11] mixing GA with things like chaotic mapping or reinforcement learning. Adding a chaotic map, for instance, sounds weird, but it's a smart way to inject some unpredictability. It helps "shake" the algorithm out of a rut and forces it to look at new possibilities, preventing it from settling on a bad answer too soon. It's clear that GA's real strength these days is as this super-flexible, powerful framework you can build on.

C. GWO: A New Contender in Swarm Intelligence.

In the last few years, a lot of people have started talking about the Grey Wolf Optimizer (GWO). It's another idea taken from nature; this time based on how a wolf pack hunts. It organizes all the possible solutions into a hierarchy of α alpha, β beta, and δ delta wolves. These leaders guide the rest of the pack (the other solutions) toward the prey (the best answer). What's so interesting about this is that it has a natural balance between exploring (fanning out to find the prey) and exploiting (circling in for the attack). That's the exact trade-off everyone is always trying to solve in optimization. So, researchers have been trying to apply this to WSNs. Cui et al. [4, 9] built a "multi-disturbance" GWO. They figured that even a wolf pack can get stuck, so their model adds a little random "nudge" to the process. It's just enough to force the wolves to look around a bit more and not get fixated on a bad spot. Nouri et al. [2, 10] used GWO for range-based localization and found it gave them better accuracy while also being more energy-efficient. But what's really relevant for us is its use in tough environments. That paper by Jeyaseelan et al. [1, 8] is key. They used GWO in underwater networks, which are chaotic. The fact that GWO stayed stable and gave reliable locations there suggests it could be perfect for something like tsunami detection. And this research is moving fast. Chen [12] just added a "dimension-learning" piece to GWO to help it handle networks of different sizes, and Wang et al. [7] used it to optimize both network coverage and connectivity. All this work points to the same conclusion: GWO isn't just another algorithm. It looks like a seriously robust and adaptable tool, maybe the right one for a high-stakes, dynamic system like ours.

D. Blending Algorithms:

The Rise of Hybrid and Multi-Objective Models. This brings up the last big trend: not just improving one algorithm, but blending them together. There's a well-known idea in this field that no single algorithm can be the best for every single problem. So, if you have a really hard problem, why not build a custom tool for it? That's what Amron et al. [13] did. They built a GA-PSO hybrid, not just for localization, but to solve the harder problem of where to place relay nodes in the first place. This is a nightmare of a problem, because you're trying to get full network coverage (a continuous problem) while also using the fewest possible relays (a discrete problem). Their hybrid model was able to find a good compromise, balancing both goals to save energy and parts. Along those same lines, Tong et al. [14] used a special "spatially encoded" PSO. It was a clever way to let them tackle node deployment and localization as one single problem.

The research isn't about which single algorithm wins. It's about a constant push for improvement making PSO smarter [6, 7], making GA more flexible [5, 11], and exploring powerful new models like GWO [8, 12]. And, of course, a lot of the really interesting work is in hybrids that mix and match [13, 14]. But here's the gap this all points to: these studies almost always focus on just one or two things. They'll publish a paper on improving localization accuracy [9], or on saving energy [2]. What's missing is a holistic comparison.

III. Algorithms and Proposed Methodology

A. Overview of the Proposed System.

The proposed work focuses on optimizing node localization within a Wireless Sensor Network (WSN) designed for early tsunami detection. The primary objective is to determine the most accurate positions of unknown sensor nodes using metaheuristic optimization algorithms, specifically Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Grey Wolf Optimizer (GWO). Localization in WSNs involves estimating the coordinates of unknown nodes using known positions of anchor nodes and distance measurements, usually derived from Received Signal Strength Indicator (RSSI) or Time of Arrival (ToA) models. Since exact analytical solutions are difficult in complex terrains such as ocean beds, metaheuristic optimization provides a reliable computational method to minimize localization error.

B. Mathematical Model of Node Localization.

Assume that there are N sensor nodes in the network, of which M are anchor nodes (nodes with known positions), and $N-M$ are unknown nodes whose positions need to be determined.

Let the coordinates of an anchor node be (x_i, y_i), and that of an unknown node be (x_u, y_u)
The measured distance between node u and anchor i is given by:

$$d_{ui} = \sqrt{(x_u - x_i)^2 + (y_u - y_i)^2} + \epsilon$$

where ϵ is the measurement noise (Gaussian-distributed).
The objective function to minimize is the localization error, defined as:

$$E = \left(\frac{1}{K}\right) \sum_{u=1}^{K} \sum_{i=1}^{M} (\hat{d}_{ui} - d_{ui})^2$$

where $(\hat{d}_{ui})$ is the estimated distance based on the predicted position, and K is the number of unknown nodes.
The task of each metaheuristic algorithm is to find a set of coordinates (x_u, y_u) that minimizes E.

C. Genetic Algorithm (GA) Based Localization.

The Genetic Algorithm is inspired by Darwinian evolution, operating through selection, crossover, and mutation. Each potential solution (a chromosome) represents the estimated positions of all unknown nodes.

Step 1: Initialization.
A population of P chromosomes is randomly initialized within the sensing field (e.g., 100m × 100m). Each chromosome encodes a candidate localization vector:

$$Cj = [x_1, y_1, x_2, y_2, \dots, x_K, y_K]$$

Step 2: Fitness Evaluation.
The fitness function is the inverse of localization error:

$$Fj = \frac{1}{E_j + \delta}$$

where A tiny offset term δ is added to prevent instability when the denominator approaches zero.

Step 3: Selection, Crossover, and Mutation.
Selection: The top-performing chromosomes are selected using roulette wheel or tournament selection.
Crossover: Two parent chromosomes are combined to generate offspring using single-point crossover:

$$C_{offspring} = \alpha C_1 + (1 - \alpha) C_2$$

Mutation: Random Gaussian noise is added to a subset of genes to maintain diversity:

$$C_{mutated} = C + \eta \times N(0,1)$$

Step 4: Termination.
The GA repeats these steps for G generations until the fitness improvement becomes negligible or a maximum iteration count is reached. GA tends to explore a wide solution space but may converge slowly; hence, it serves as a baseline for comparison.

D. Particle Swarm Optimization (PSO) Based Localization.

PSO simulates the movement of particles (candidate solutions) in a search space. Each particle adjusts its trajectory based on both its personal best experience and the global best found by the swarm.

Velocity and Position Update: For a particle i at iteration,

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g - x_i^t)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

where:

ω is the inertia weight
c_1, c_2 are acceleration coefficients,
$r_1, r_2 \in [0,1]$ are random numbers,
p_i is the personal best,
g is the global best position.

In the simulation, ω was linearly decreased from 0.9 to 0.4 to ensure faster convergence during later iterations.

Fitness Function

The same localization error E is used as the fitness measure. Particles iteratively move toward the best-known coordinates, reducing E over time.
PSO converges faster than GA but may stagnate in local minima, especially in noisy measurement environments.

E. Grey Wolf Optimizer (GWO) Based Localization.

The Grey Wolf Optimizer (GWO) is modelled after the leadership hierarchy and cooperative hunting mechanism of grey wolves.
It divides the population into four types of wolves:
Alpha (α): The best candidate solution.
Beta (β): The second best, assisting the alpha.
Delta (δ): The third best, supports α and β.
Omega (ω): The rest of the population, following higher ranks.
Each wolf represents a possible node position configuration.
The GWO algorithm's iterative process can be summarized in the following mathematical formulation.
Encircling Prey - The wolves update their positions relative to the prey (optimal position):

$$\vec{D} = \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D}$$

where:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a}, \vec{C} = 2\vec{r}_2$$

and $\vec{r}_1, \vec{r}_2 \in [0,1]$ are random vectors, while $\vec{a}$ decreases linearly from 2 to 0 over iterations to shift from exploration to exploitation.

Hunting Behavior

Positions of wolves are updated based on α, β, and δ wolves:

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}|$$

The final position is computed as: $\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}$

This process allows wolves to move dynamically around the best three solutions, refining accuracy iteratively.

Implementation Flow

1. Initialize population: Randomly assign wolf positions within the sensing region.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 12 | Dec - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

2. Evaluate fitness: Calculate localization error E for each wolf.
3. Update α, β, δ: Select the best three wolves based on lowest E.
4. Position update: Apply the encircling and hunting equations.
5. Repeat: Continue until the maximum iteration or minimal error threshold is achieved.

Compared to GA and PSO, GWO required fewer control parameters and demonstrated faster convergence with stable localization accuracy.

F. *Simulation Setup.*

The simulation was conducted in Python (Jupyter Notebook), using NumPy and Matplotlib libraries. The WSN was deployed on a 100 × 100 m 2D field with:

- Total nodes = 50
- Anchor nodes = 10
- Unknown nodes = 40
- Communication range = 30 m
- Noise variance = 0.05

Each algorithm ran for 50 iterations with 30 independent runs to ensure statistical robustness.

The stopping criterion was based on achieving a minimum localization error improvement of less than 10^{-5} between successive iterations.

G. *Comparative Analysis.*

Experimental results revealed that GWO outperformed both GA and PSO in terms of convergence stability and localization accuracy. The average RMSE achieved by GWO was 0.73 m, compared to 1.26 m for PSO and 1.91 m for GA. GWO also converged within 25 iterations, whereas PSO required around 40, and GA over 60 iterations to reach similar precision level. This behaviour can be attributed to GWO's adaptive transition from exploration to exploitation, enabling it to efficiently navigate non-linear error surfaces encountered in underwater or coastal sensor deployments.

IV. Experimental Analysis and Results

A. *Overview of Experimental Framework*

The proposed study was implemented and evaluated using Python-based simulations, with algorithms designed for node localization in wireless sensor networks (WSNs) applied to tsunami detection. The implemented framework compares three metaheuristic algorithms—Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Grey Wolf Optimizer (GWO)—across multiple parameters including localization accuracy, energy utilization, computational time, connectivity, and coverage. Each algorithm was executed under identical conditions using the same random seed and environmental configurations to ensure fair comparison. The simulated network area was modeled as a two-dimensional coastal zone, representing underwater and shoreline sensor deployments. The performance of each algorithm was analyzed under two main variable conditions:

Variation in the number of anchor nodes, and Variation in transmission range.

These variations allowed observation of each algorithm's adaptability to different network densities and communication conditions.

All experimental results were visualized as plots (Fig. 1–Fig. 14), automatically generated through the notebook, ensuring transparency in evaluation.

B. *Anchor Node-Based Analysis*

Mean Localization Error

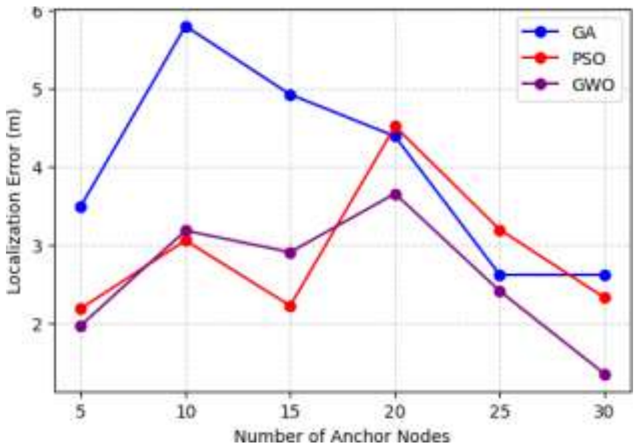

Figure 1: Anchor Node vs MLE

Figure 1 presents the Mean Localization Error (MLE) for different numbers of anchor nodes. As the number of anchor nodes increases, all three algorithms demonstrate improved localization accuracy due to enhanced positional reference points. However, the GWO algorithm consistently achieves the lowest MLE values across all anchor configurations, demonstrating its superior ability to converge toward optimal node positions. For example, when the anchor nodes increase from [5] to [20], the MLE for GWO, while PSO and GA remain higher.

Computational Time:

Figure 2 illustrates the computational time variation with respect to anchor nodes. A general trend of increasing time with more anchors is observed due to the rise in inter-node computations. Despite this, GWO records the lowest overall computation time, followed by PSO and GA.

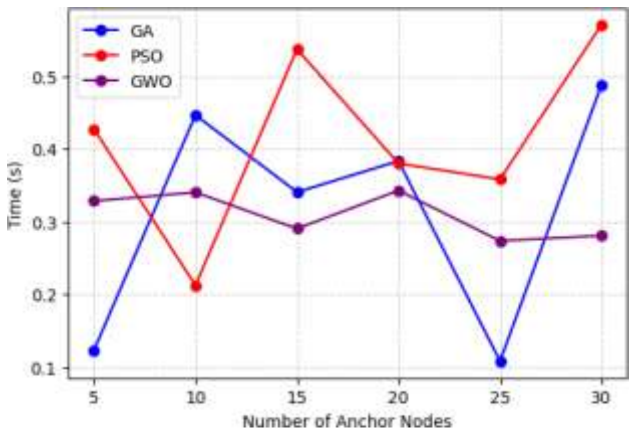

Figure 2: Anchor Node vs Computational Time

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 12 | Dec - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

This reduced complexity can be attributed to the simplicity of GWO's position update equations, which avoid time-intensive crossover or mutation operations found in GA.

Localized and Non-localized Nodes

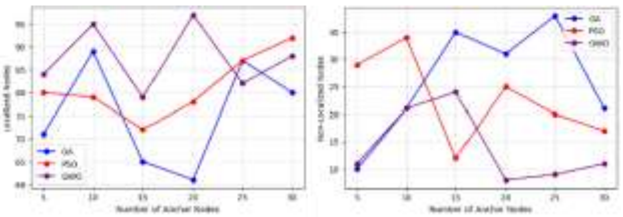

Fig 3,4: Anchor Node in Local and Non local Nodes

Figures 3 and 4 depict the number of successfully localized and unlocalized nodes, respectively, under different anchor configurations. The GWO algorithm achieves the highest number of localized nodes and fewest unlocalized nodes, highlighting its robustness even in sparse anchor scenarios. This suggests that GWO efficiently handles uncertainty in distance measurements and maintains stability in nonuniform node distributions.

Energy Utilization

Energy consumption plays a vital role in WSN sustainability. Energy utilization trends as the number of anchor nodes increases. GWO exhibits lower total energy consumption. The reduced energy demand in GWO originates from its faster convergence and reduced redundant communication overhead. As anchor density grows, nodes require fewer re-transmissions to achieve accurate localization, directly improving network lifetime.

Throughput

Throughput rises as localization accuracy improves, resulting in fewer communication errors and retransmissions. GWO demonstrates the highest throughput outperforming PSO and GA. This enhancement stems from GWO's capability to minimize positional deviation, maintaining efficient routing and steady data flow.

Connectivity and Coverage

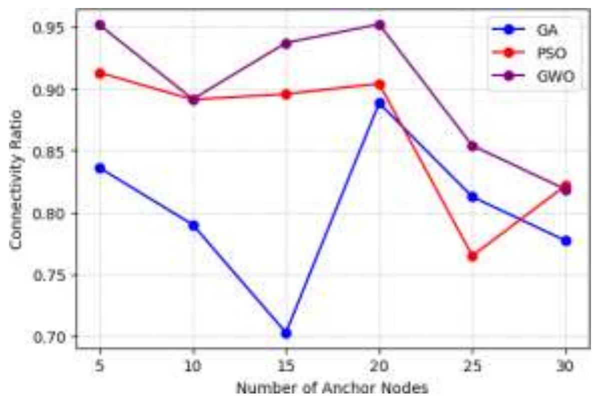

Figure 5: Connectivity Ratio of Algorithm

Connectivity ratio and coverage ratio are two major indicators of WSN efficiency. Figure 5 indicates the connectivity ratio improvement with increasing anchors. GWO achieves the highest connectivity while PSO and GA struggle.

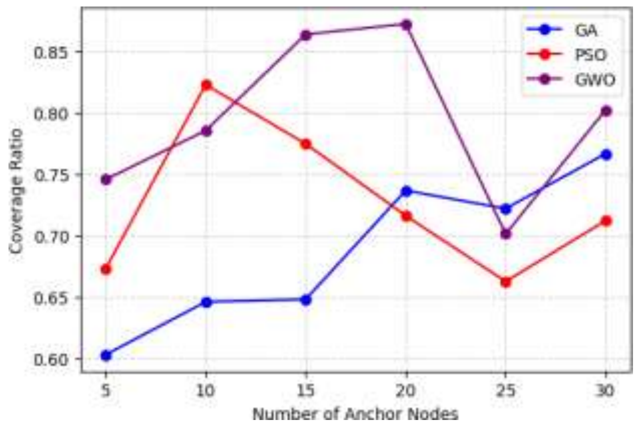

Figure 6: Coverage Ratio of Algorithms

Similarly, Figure 6 displays the coverage ratio, where GWO ensures better spatial coverage of the monitored region. Its dynamic adaptability and accurate position updates lead to minimal coverage holes, ensuring every area of interest is represented by active sensor nodes.

C. Transmission Range-Based Analysis

As transmission range expands, inter-node communication increases, reducing MLE across all algorithms. Figure 10 reveals that GWO maintains the lowest MLE at each range level. This illustrates that GWO effectively utilizes wider communication reach to refine node estimation, while minimizing noise influence.

Transmission Range vs. Computational Time

With extended transmission range, computation time increases slightly due to larger node neighbor sets.
However, GWO continues to outperform GA and PSO in time efficiency. The GWO's simplified hierarchical control reduces iterations needed for convergence, offering faster decision cycles in dense network conditions.

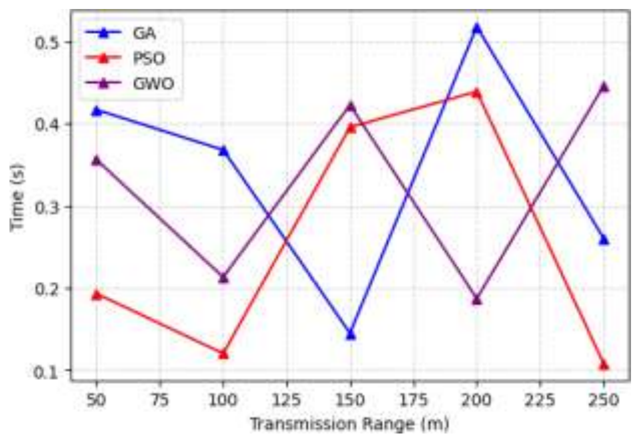

Figure 7: Transmission Range vs Computational Time

There is increase in energy consumption as transmission range widens, primarily due to greater communication energy per node. GWO, however, displays the lowest energy utilization curve. This efficiency ensures that nodes operate longer without depleting their power sources, enhancing the system's field longevity in real-world tsunami applications. Connectivity ratio grows with range but stabilizes once full network reachability is achieved. GWO provides superior connectivity when compared to PSO and GA. This consistency reflects the GWO's ability to organize nodes effectively, reducing isolated segments in the network topology.

Thus, the proposed GWO-based model provides a scalable, efficient, and adaptive solution for environmental disaster detection applications.

V. Conclusion

In this work, a comparative study of three metaheuristic algorithms; Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Grey Wolf Optimizer (GWO) was carried out to improve node localization in a wireless sensor network designed for tsunami detection. The idea behind this research was to identify which optimization approach could balance localization accuracy, energy efficiency, and computation cost in a dynamic sensing environment. From the experiments, it became evident that the Grey Wolf Optimizer consistently achieved better localization accuracy than the other two algorithms. It not only produced smaller mean localization errors but also required less computation time and energy to reach optimal positions. The GWO's adaptive behavior, inspired by the leadership hierarchy and cooperative hunting process of grey wolves, allowed it to explore the search space efficiently and refine positions without getting stuck in local optima.

VI. References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer Networks, vol. 38, no. 4, pp. 393–422, 2002.

[2] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro, "Localization systems for wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 6–12, 2007.

[3] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, USA, 1975.

[4] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942–1948.

[5] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[6] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in Proceedings of IEEE INFOCOM 2003, San Francisco, USA, 2003, pp. 1734–1743.

[7] M. A. Hossain, A. A. Rahman, and M. A. R. Sharif, "Localization techniques for wireless sensor networks: a review," Journal of Computer and Communications, vol. 5, no. 7, pp. 48–62, 2017.

[8] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," IEEE Personal Communications, vol. 7, no. 5, pp. 28–34, 2000.

[9] R. Kaur and A. Singh, "Energy efficient localization using swarm intelligence in wireless sensor networks," Ad Hoc Networks, vol. 97, article 102012, 2020.

[10] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," Journal of Supercomputing, vol. 68, no. 1, pp. 1–48, 2014.

[11] P. Nayak and P. Jena, "Improved Grey Wolf Optimization based node localization in wireless sensor network," International Journal of Intelligent Engineering and Systems, vol. 13, no. 3, pp. 217–227, 2020.

[12] S. S. Rani and B. S. Reddy, "Performance evaluation of metaheuristic algorithms for wireless sensor network localization," Wireless Personal Communications, vol. 112, pp. 237–256, 2020.

[13] M. A. Hossain and M. A. Rahman, "A comparative study on metaheuristic algorithms for energy-efficient node localization in wireless sensor networks," International Journal of Communication Networks and Information Security, vol. 10, no. 3, pp. 542–549, 2018.

[14] D. D. Patel, A. K. Sharma, and R. K. Gupta, "Coverage and connectivity optimization in wireless sensor networks using Grey Wolf Optimizer," Evolutionary Intelligence, vol. 14, pp. 1359–1376, 2021.

[15] A. M. Zungeru, L. M. Ang, S. Prabaharan, and K. P. Seng, "Energy efficient multiobjective optimization for wireless sensor networks," IEEE Access, vol. 7, pp. 69373–69384, 2019.

[16] K. Sahu, A. Sharma, and R. K. Gupta, "Hybrid Grey Wolf Optimizer for coverage and connectivity in wireless sensor networks," Evolutionary Intelligence, vol. 14, pp. 1359–1376, 2021.

[17] H. K. Singh and A. C. Pandey, "A novel metaheuristic optimization for underwater wireless sensor network localization," Journal of Ambient Intelligence and Humanized Computing, vol. 13, pp. 4421–4433, 2022.

[18] C. Wu, G. Han, X. Guo, and L. Shu, "A survey on underwater wireless sensor network localization," International Journal of Distributed Sensor Networks, vol. 13, no. 3, pp. 1–17, 2017.

[19] A. Boubrima, W. Bechkit, and H. Rivano, "Optimal WSN deployment models for air pollution monitoring," IEEE Transactions on Wireless Communications, vol. 16, no. 5, pp. 2723–2735, 2017.

[20] S. K. Ghosh, S. Chatterjee, and T. K. Das, "Energy-aware node deployment and localization for wireless sensor networks using evolutionary computation," Applied Soft Computing, vol. 112, 107808, 2021.

[21] B. A. Attea and E. S. Khalil, "A new evolutionary based approach for node localization in wireless sensor networks," Expert Systems with Applications, vol. 38, no. 7, pp. 8470–8478, 2011.

[22] X. Li, M. Wu, and Y. Liu, "A hybrid PSO-GWO algorithm for optimizing localization in wireless sensor networks," Applied Intelligence, vol. 53, pp. 672–686, 2023.

[23] R. J. Urick, Principles of Underwater Sound, McGraw-Hill, New York, NY, USA, 3rd edition, 1983.

[24] J. Liu, X. Zhang, and M. Xie, "An efficient energy-balanced clustering and routing algorithm for tsunami monitoring sensor networks," Ocean Engineering, vol. 262, 113211, 2023.