Addressing the NP-Hard Bin Packing Problem with Ant Colony Optimization and Nature-Inspired Algorithms

Candidate Number: 028387
University of Exeter

ABSTRACT

This research investigates the application of Ant Colony Optimization (ACO), an algorithm inspired by natural behaviour, to tackle the complex NP-hard Bin Packing Problem (BPP), where finding precise solutions proves computationally demanding. The study's main objective is to assess how well ACO can reduce the weight difference between the heaviest and lightest bins by adjusting critical parameters, particularly the pheromone evaporation rate and the number of ants involved. Experiments were carried out on two versions of the Bin Packing Problem, labelled BPP1 and BPP2, revealing that higher pheromone evaporation rates tended to result in a more balanced weight distribution across bins. These findings highlight how sensitive ACO is to its parameter choices, offering useful insights for refining the algorithm's performance on NP-hard combinatorial challenges.

1. INTRODUCTION TO BIN PACKING PROBLEM

1.1 Background and Importance of the Problem

The Bin Packing Problem (BPP) is a classic NP-hard combinatorial optimisation problem, meaning it becomes exponentially more challenging to solve as the number of items and bins increases. For large problem instances, finding an exact solution is computationally infeasible[2]. The Bin packing problem is a problem where the solution involves packing a finite set of items with weights into a finite number of bins without exceeding the specified maximum capacity of the bins, with the goal to minimise the weight difference between the largest and smallest bin (fitness) [1]. The exact methods for large-scale BPP instances are impractical; nature-inspired algorithms are used to find solutions. In recent years, evolutionary algorithms such as Genetic Algorithms (GA), Particle Swarm Optimisation (PSO), and Ant Colony Optimization (ACO) have shown promise in providing effective solutions to BPP [1].

1.2 Research Objectives and Scope of the Study

This study investigates how different parameter settings, specifically the pheromone evaporation rate and the number of ants can affect the performance of Ant Colony Optimization (ACO) in solving the Bin Packing Problem (BPP). It also examines how suboptimal parameter choices can lead the algorithm to converge prematurely to local minima, thus hindering it from finding the optimal solution[6]. Showing the importance of iterative experimentation, the study highlights the trial-and-error nature of parameter tuning in achieving optimal results.

2. Overview of Nature-Inspired Algorithms in Optimisation

Nature-inspired algorithms (NIAs) are problem-solving techniques inspired by natural phenomena such as evolution, swarm behaviour, and foraging. NIAs are particularly useful for complex, high-dimensional, or NP-hard problems, where traditional optimisation methods become computationally infeasible. Using stochastic techniques, like randomised processes to explore a wide range of solutions, NIAs can efficiently search for optimal or near-optimal solutions [5]. A key advantage of NIAs is their adaptability, as they do not rely on specific assumptions about problem structure, making them flexible and suitable for various complex problems [8].

3. Comparison of Nature-Inspired Algorithms for BPP.

Ant Colony Optimization (ACO) is one of many Nature Inspired Algorithms (NIA); such as Genetic Algorithms (GA), Particle Swarm Optimisation (PSO), and Simulated Annealing (SA). ACO is known for its unique approach to constructing and optimising solutions. Unlike GA, which will evolve new solutions over generations through selection, crossover, and mutation[1, 4], ACO uses a collaborative learning mechanism where artificial ants construct solutions based on pheromone trails[6]. These trails are iteratively updated to reflect solution quality, enabling ACO to balance exploration and exploitation effectively. PSO, inspired by social swarming behaviour, adjusts particles' positions based on both individual and group experiences[2]. Although PSO converges quickly, it can struggle with fine-tuning in constrained problems like BPP[2]. Similarly, Simulated Annealing, inspired by the metallurgical annealing process, explores solutions by accepting specific degradations in quality to escape local optima, though it may require extensive computational time. ACO's iterative pheromone updating mechanism is especially well-suited for combinatorial optimisation tasks like BPP, guiding the search process toward optimal or near-optimal solutions in complex search spaces.

In recent years, nature-inspired algorithms have shown significant potential in solving the Bin Packing Problem (BPP), where finding exact solutions is computationally prohibitive. Among these, Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), and more recent methods like Cuckoo Search (CS) with Lévy flights [2] and Lévy-based Whale Optimization (LWO) [2] offer unique strategies. GA, relying on crossover and mutation, can explore a broad solution space but often requires extensive computational resources to fine-tune high-dimensional solutions. PSO, inspired by swarming behaviour, achieves rapid convergence by updating particles based on individual and collective experiences, yet may prematurely settle on suboptimal solutions in constrained BPP scenarios [1]. Similarly, Lévy-based Whale Optimization, which mimics whale foraging behaviour with Lévy flights, adapts search patterns to balance exploration and exploitation effectively. Ant Colony Optimization (ACO) leverages a unique pheromone-feedback mechanism that reinforces promising solutions while balancing exploration.

4. DESCRIPTION OF RESULTS

This section examines the effects on ACO's performance under two categories of testing, BPP1 and BPP2. In BPP1, 500 items with sequential weights from 1 to 500 are packed into 10 bins. In BPP2, each of the 500 items has a weight defined by the formula $i^2/2$, where i represents the item's position from 1 to 500. Both BPPs will be tested under four different settings, with each having five trials: the first with p = 100 and e = 0.90, the second with p = 100 and e = 0.60, the third with p = 10, and e = 0.90, and finally p = 10, and e = 0.60. By varying these parameters, we observe key patterns in convergence speed, stability, and solution quality, which we compare with existing findings.

4.1 Effect of Evaporation Rate

Evaporation Rate 0.9: The higher evaporation rate promotes continued exploration, as shown by a steady but gradual convergence in BPP1 and BPP2 configurations with 100 ants. The fitness values gradually declined over the first 400 evaluations but had minor fluctuations, indicating active exploration as seen in **Figure1** and **Figure 2**. Due to the quadratic nature of BPP2, the final results were much less desirable, having a fitness in the tens of thousands on average[6].

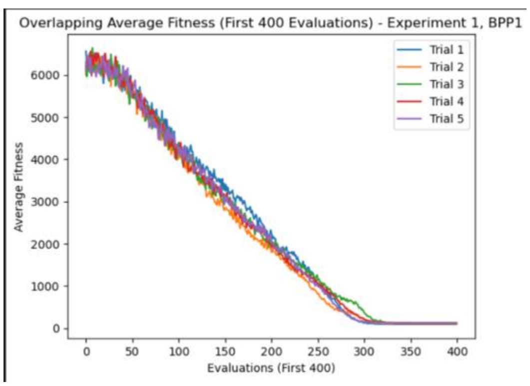

Figure 1. Ants = 100, Evaporation Rate = 0.9, BPP1

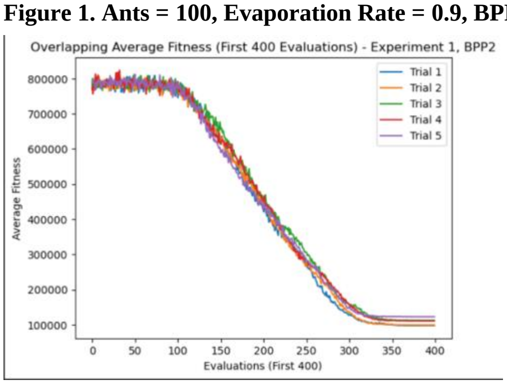

Figure 2. Ants = 100, Evaporation Rate = 0.9, BPP2

Evaporation Rate 0.6: Lower evaporation rates cause rapid convergence across both BPP1 and BPP2 configurations, especially in the early evaluations. The algorithm stabilises faster but risks settling in local minima, as shown by higher final fitness values in some trials, as seen in **Figure 3** and **Figure 4**. The fitness score for BPP1, where the parameters were p = 100 and evaporation was 0.90, was 91.00. In contrast, the fitness was 245.00 when the evaporation was 0.60. This shows that with lower evaporation, the algorithm converges too quickly and does not explore enough possible solutions before getting stuck in local minima.

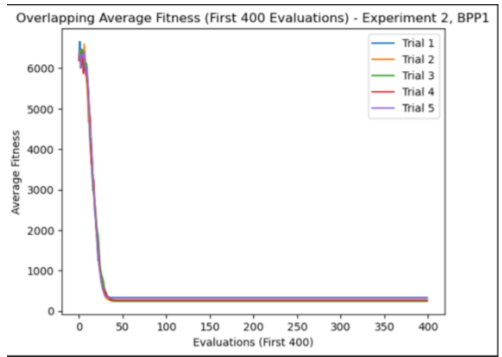

Figure 3. Ants = 100, Evaporation Rate = 0.6, BPP1

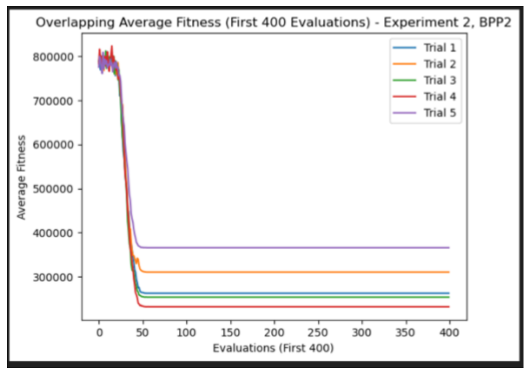

Figure 4. Ants = 10, Evaporation Rate = 0.6, BPP1

4.2 Effect of Ant Count

Higher Ant Count (100 ants): With 100 ants, the algorithm consistently achieves lower and more stable final fitness values across trials. Additionally, the convergence is much smoother compared to 10 ants, with minor variability in initial evaluations but strong consistency in later stages. Higher ant counts enhance the solution by reinforcing successful paths more robustly, a finding consistent throughout testing, achieving stable optimisation outcomes as shown in **Figure 1** compared to **Figure 8.**

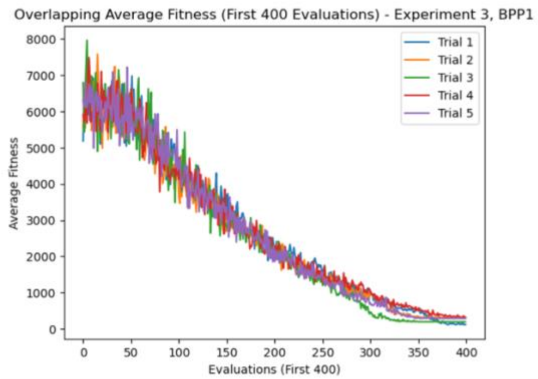

Figure 8. Ants = 10, Evaporation Rate = 0.9, BPP1

Lower Ant Count (10 ants): Lower counts lead to slower, more fluctuating convergence, with increased variability across trials. This setting promotes exploration but lacks stability, as the reduced reinforcement limits pheromone accumulation on optimal paths. The lack of stability with smaller populations observed here shows that smaller populations lead to greater exploration but may struggle to reinforce high-quality solutions consistently, as shown in **Figure 8.**

4.3 Complexity Impact (BPP1 vs. BPP2)

BPP2, which has a higher bin count and a more complex weight distribution, consistently shows higher initial fitness values and slower convergence than BPP1. Lower evaporation rates (0.6) stabilise BPP2 faster but may limit the exploration needed to find globally optimal solutions, as shown in **Figure 6** and **Figure 7.** When comparing the box plots of the two, it shows that BPP2 has fitness cores in the tens of thousands and a larger variation on fitness scores between the 5 trials. As the experiment continues, i.e., the number of ants and evaporation decreases, the fitness score increases, and the results become less predictable upon multiple trials, as in **Figure 6**. This is exacerbated when looking at **Figure 7**, which shows how BPP2 dwarfs the fitness scores of BPP1. However, the trend is still the same across BPP1 and BPP2 as when parameter settings are lowered, the fitness score increase.

BPP1

Experiment 1: Number of Ants = 100, Evaporation Rate = 0.9

Experiment 2: Number of Ants = 100, Evaporation Rate = 0.6,

Experiment 3: Number of Ants = 10, Evaporation Rate = 0.9,

Experiment 4: Number of Ants = 10, Evaporation Rate = 0.6,

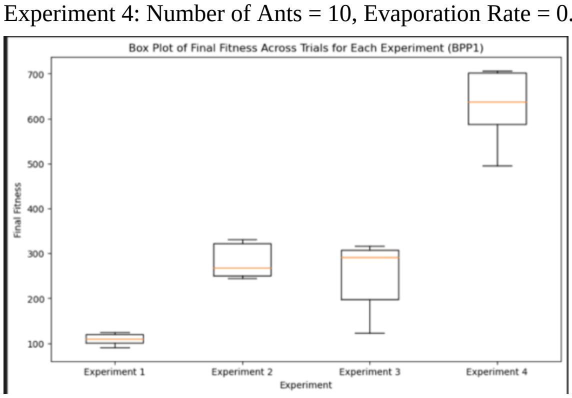

Figure 6. Box plot of best fitness BPP1 and BPP2

BPP2

Experiment 5: Number of Ants = 100, Evaporation Rate = 0.9

Experiment 6: Number of Ants = 100, Evaporation Rate = 0.6,

Experiment 7: Number of Ants = 10, Evaporation Rate = 0.9,

Experiment 8: Number of Ants = 10, Evaporation Rate = 0.6

Figure 7. Box plot of best fitness BPP1 and BPP2

5. DISCUSSION AND RESULTS

5.1 Question 1: Which combination of parameters produces the best results?

The best outcomes for reaching the lowest fitness values for BPP1 were obtained with 100 ants and an evaporation rate of 0.90. The method was able to search widely throughout the solution space before determining the best weight distribution among the bins. This parameter configuration effectively balances exploration and convergence. The efficiency of this combination in reducing the weight difference across bins is seen by the consistent fall in fitness scores over the first 400 evaluations.

5.2 Question 2: What do you think is the reason for your findings in Question 1?

The strong performance of the 100 ants and 0.90 evaporation rate combination is likely due to its ability to balance exploration and exploitation. A higher evaporation rate prevents excessive pheromone accumulation, which can cause the algorithm to converge too early on suboptimal paths. This configuration promotes a gradual convergence, giving the algorithm ample opportunity to explore alternative solutions. Additionally, the larger ant population of 100 consistently reinforces successful paths, enhancing solution stability and increasing the likelihood of finding near-optimal solutions.

5.3 Question 3: How do each of the parameter settings influence the performance of the algorithm?

Each parameter setting significantly impacts the algorithm's performance. A high evaporation rate of 0.90 leads to slower but more controlled convergence, facilitating better exploration and reducing the risk of local minima entrapment, which is especially beneficial for complex problem instances like BPP2. Conversely, a lower evaporation rate of 0.60 accelerates convergence but at the cost of reduced exploration, increasing the likelihood of premature convergence. Similarly, a high ant count of 100 contributes to solution stability by providing more pheromone reinforcement on successful paths, resulting in smoother convergence and lower variability across trials. In contrast, a low ant count of 10 increases exploration but limits pheromone accumulation, leading to less consistent reinforcement of optimal paths and higher solution variability.

5.4 Question 4: Do you think that one of the algorithms in your literature review might have provided better results? Explain your answer.

While alternative algorithms, such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), offer competitive approaches to combinatorial optimisation problems, they may not perform as effectively as ACO for the Bin Packing Problem. GA relies on selection and mutation, which can be computationally intensive and may require extensive fine-tuning to achieve stable solutions in high-dimensional spaces like BPP. PSO, while quick to converge, can struggle with fine-tuning in constrained problems, potentially leading to premature convergence. In contrast, ACO's pheromone feedback mechanism provides an adaptive balance of exploration and exploitation, reinforcing high-quality paths iteratively. This iterative feedback aligns well with the bin-packing objective of balancing bin weights, making ACO a more suitable choice for this problem. [7]

6. FURTHER WORKS

Additional testing was carried out by dramatically increasing the evaporation rate to 0.99 and increasing the number of ants to 200 to investigate potential enhancements. This modification aimed to determine whether a greater ant population and higher pheromone evaporation would produce better outcomes. On the other hand, BPP1's performance with these new parameters was one of the worst in the study, with a fitness score of 1163.55. The heightened parameter values might lead the algorithm to converge prematurely or become stuck in local minima; this finding emphasises how sensitive ACO is to parameter adjustment. In these conditions, the algorithm investigated multiple solution paths for each of the five trials, as shown in **Figures 9** and **Figure 10.** Still, no single path or solution consistently resulted in a superior outcome.

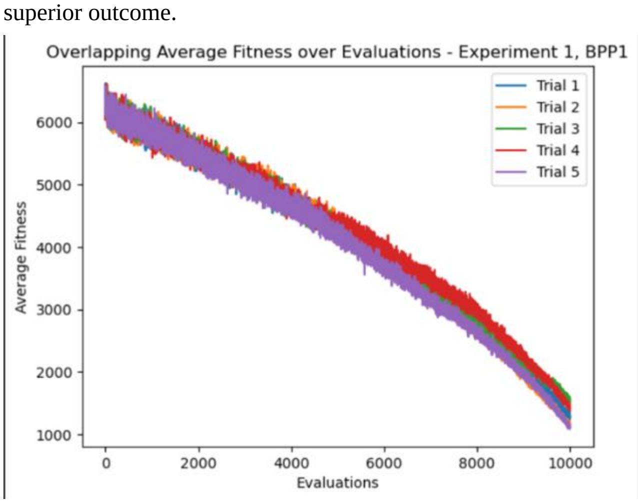

Figure 9. Ants = 200, Evaporation Rate = 0.99, BPP1 (1,000)

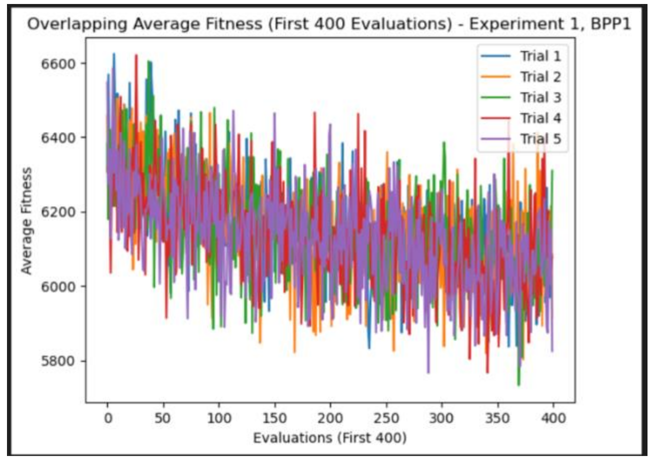

Figure 10. Ants = 200, Evaporation Rate = 0.99, BPP1 (400)

7. CONCLUSION

In conclusion, optimising the ACO's performance for the Bin Packing Problem relies on careful experiments to calibrate the parameters, especially for the evaporation rate and the ant count. The results show that ACO is a strong candidate for the BPP due to its ability to balance exploration and exploitation. The configuration that yielded the best results through countless experiments was the evaporation rate of 0.90 and 100 ants, achieving a stable convergence and low fitness score.

8. REFERENCES

[1] Abdel-Basset, M., Manogaran, G., Abdel-Fatah, L., and Mirjalili, S. 2018. An improved nature-inspired meta-heuristic algorithm for 1-D bin packing problems. *Personal and Ubiquitous Computing*, 22, 1117–1132. DOI= https://doi.org/10.1007/s00779-018-1132-7.

[2] Chanaleä, M., and Ezugwu, A. E. 2021. Metaheuristic algorithms for one-dimensional bin-packing problems: A survey of recent advances and applications. *Journal of Intelligent Systems*. https://doi.org/10.1515/jisys-2020-0117.

[3] González-San Martín, J., Cruz-Reyes, L., Dorronsoro, B., Quiroz-Castellanos, M., Fraire, H., Gómez-Santillán, C., and Rangel-Valdez, N. Optimization Models and Methods for Bin Packing Problems: A Case Study on Solving 1D BPP.

[4] Levine, J., and Ducatelle, F. 2004. Ant colony optimization and local search for bin packing and cutting stock problems.

[5] Misra, B., and Chakraborty, S. 2024. Ant Colony Optimization—Recent Variants, Application and Perspectives. In *Applications of Ant Colony Optimization and its Variants*, Nilanjan Dey (Ed.). Springer Tracts in Nature-Inspired Computing (STNIC), pp. 1–17.

[6] Ye, Y., Li, Y., and Wang, Y. 2014. An Improved ACO Algorithm for the Bin Packing Problem with Conflicts Based on Graph Coloring Model. In *Proceedings of the 2014 International Conference on Management Science & Engineering (21st)*, August 17-19, 2014, Helsinki, Finland

[7] Selvi, V., & Umarani, R. 2010. Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques. *International Journal of Computer Applications*, 5(4), 1-6.