

FOUR UNIQUE ARTIFICIAL INTELLIGENCE ALGORITHMS

¹Satish Gajawada

¹IIT Roorkee Alumnus

Satish Gajawada: satish.gajawada.iit@gmail.com

Corresponding Author: SATISH GAJAWADA

ABSTRACT

This article contributes four unique Artificial Intelligence algorithms to Swarm Intelligence which is an active area of research. Cricket Match Runs Algorithm (CMRA), Rice Bags Sales Algorithm (RBSA), English Language Sentence Algorithm (ELSA) and Object Swarm Optimization Algorithm (OSOA) are four novel Swarm Intelligence algorithms designed in this article. CMRA, RBSA and ELSA belongs to Human Swarm Optimization (HSO) field. The Object Swarm Optimization Algorithm (OSOA) does not belong to any particular category of Swarm Intelligence like Particle Swarm Optimization or Human Swarm Optimization but it belongs to Object Swarm Optimization where Objects move in search space. Hence OSOA belongs to Object Swarm Intelligence.

KEYWORDS: Cricket match, CMRA, Rice Bags, RBSA, English sentence, ELSA, Objects, OSOA, Swarm Intelligence, Object Swarm Intelligence, Human Swarm Optimization, HSO, Artificial Intelligence

1. INTRODUCTION

Articles [1] to [5] show literature related to Swarm Intelligence. Four Artificial Intelligence algorithms are designed in this work. Section 2 shows Cricket Match Runs Algorithm (CMRA). Rice Bags Sales Algorithm (RBSA) is shown in Section 3. Section 4 shows English Language Sentence Algorithm (ELSA). Object Swarm Optimization Algorithm (OSOA) is shown in Section 5. Conclusions are made in Section 6 followed by references at the end.

2. CRICKET MATCH RUNS ALGORITHM

The Runs scored in a cricket match are stored in Cricket_Runs_Scored Array. In line no. 2 the ball number is set to 1. Population of Humans is initialized in line no. 3. In 4th line, Generation counter is set to 1. Line no. 5 identifies Best_Human with best fitness value. In line no. 6 for each Human loop is started. Direction is calculated in lines 7 and

8. Line no. 9 shows position update equation. The Human moves along the Best_Human direction and the magnitude of this movement is Cricket_Runs_Scored[ball] which is nothing but the runs scored in the current ball. The ball is incremented by 1. Line no. 11 ends for each Human loop. The Generation is incremented by 1. This process is continued until termination condition is reached in line no. 13.

Procedure: Cricket Match Runs Algorithm (CMRA)

1. Initialize Cricket_Runs_Scored Array
2. Set ball = 1
3. Initialize Population of Humans
4. Set Generation = 1
5. Identify Best_Human with best fitness value
6. for each Human:
7. Direction = Best_Human – Human
8. Convert Direction into unit vector
9. position = position + Direction*Cricket_Runs_Scored[ball]
10. ball = ball + 1
11. end for each Human loop
12. Generation = Generation + 1
13. Loop until termination condition is reached

3. RICE BAGS SALES ALGORITHM

Probabilities of number of Rice Bags sold to current customer are initialized in first four lines. Population of Humans is initialized and Generation count is set to 1. Best_Human with best fitness value is identified. In line no. 8, for each Human

loop is started. Direction of movement is calculated in lines 9 and 10. The position update equation is based on random number generated and probabilities. Human moves along Direction and magnitude of this movement is equal to the number of Rice Bags lifted or sold to customer. This is shown in lines 12 to 19. In line no. 20, for each Human loop is ended. Generation count is incremented by 1. This process is continued until termination condition is reached in line no. 22.

Procedure: Rice Bags Sales Algorithm (RBSA)

1. One_Rice_Bag_Lifted_Probability = 0.5
2. Two_Rice_Bags_Lifted_Probability = 0.25
3. Three_Rice_Bags_Lifted_Probability = 0.125
4. Four_Or_More_Rice_Bags_Lifted_Probability = 0.125
5. Initialize Population of Humans
6. Set Generation = 1
7. Identify Best_Human with best fitness value
8. for each Human:
9. Direction = Best_Human – Human
10. Convert Direction into unit vector
11. Generate Random Number “R”
12. if $0 < R < 0.5$ then
13. position = position + Direction*1
14. if $0.5 < R < 0.75$ then
15. position = position + Direction*2
16. if $0.75 < R < 0.875$ then
17. position = position + Direction*3
18. if $0.875 < R < 1$ then
19. position = position + Direction*4
20. end for each Human loop
21. Generation = Generation + 1
22. Loop until termination condition is reached

4. ENGLISH LANGUAGE SENTENCE ALGORITHM

Ten English Sentences are selected in line no. 1. Sentence_Length_Array contains the length of 10 sentences. Population of Humans is initialized and Generation count is set to 1. In line no. 5, Best_Human with best fitness value is identified. For each Human loop is started in line no. 6. Direction of movement is calculated in lines 7 and 8. Each sentence is selected with a probability 0.1. The position update equation depends on random number R and probabilities. The Human moves along the Direction and magnitude of this movement is Sentence_Length_Array[Sentence_Number_Selected]. This is shown in lines 10 to 29. For each Human loop is ended in line no. 30. Generation count is incremented by 1. This process is continued until termination condition is reached in line no. 32.

Procedure: English Language Sentence Algorithm (ELSA)

1. Select 10 English Sentences
2. Calculate Sentence_Length_Array
3. Initialize Population of Humans
4. Set Generation = 1
5. Identify Best_Human with best fitness value
6. for each Human:
7. Direction = Best_Human – Human

8. Convert Direction into unit vector
9. Generate Random Number “R”
10. if $0 < R < 0.1$ then
11. position = position + Direction* Sentence_Length_Array[1]
12. if $0.1 < R < 0.2$ then
13. position = position + Direction* Sentence_Length_Array[2]
14. if $0.2 < R < 0.3$ then
15. position = position + Direction* Sentence_Length_Array[3]
16. if $0.3 < R < 0.4$ then
17. position = position + Direction* Sentence_Length_Array[4]
18. if $0.4 < R < 0.5$ then
19. position = position + Direction* Sentence_Length_Array[5]
20. if $0.5 < R < 0.6$ then
21. position = position + Direction* Sentence_Length_Array[6]
22. if $0.6 < R < 0.7$ then
23. position = position + Direction* Sentence_Length_Array[7]
24. if $0.7 < R < 0.8$ then
25. position = position + Direction* Sentence_Length_Array[8]
26. if $0.8 < R < 0.9$ then
27. position = position + Direction* Sentence_Length_Array[9]
28. if $0.9 < R < 1$ then
29. position = position + Direction* Sentence_Length_Array[10]
30. end for each Human loop
31. Generation = Generation + 1
32. Loop until termination condition is reached

5. OBJECT SWARM OPTIMIZATION ALGORITHM

The position of objects is initialized in line no. 1. Generation count is set to 1. In line no. 3, Best_Object with best fitness value is identified. For each Object loop is started in line no. 4. The Direction of movement is calculated in lines 5 and 6. Rotate_Direction Probabilities are initialized.

Based on Random number generated and probabilities, the Direction vector is rotated. Position update equation is shown in line number. 17. The Object moves along Direction and magnitude of this movement is Step value. In line no. 18, for each Object loop is ended. Generation counter is incremented by 1. This process is continued until termination condition is reached in line no. 20.

Procedure: Object Swarm Optimization Algorithm (OSOA)

1. Initialize position of Objects in Search Space
2. Set Generation = 1
3. Identify Best_Object with best fitness value
4. for each Object:
5. Direction = Best_Object – Object
6. Convert Direction into unit vector
7. Rotate_Direction_By_Zero_Degrees_Probability = 0.8
8. Rotate_Direction_By_Five_Degrees_Probability = 0.1
9. Rotate_Direction_By_Minus_Five_Degrees_Probability = 0.1

10. Generate Random Number “R”
11. if $0 < R < 0.8$ then
12. do nothing
13. if $0.8 < R < 0.9$ then
14. Rotate Direction by 5 Degrees
15. if $0.9 < R < 1$ then
16. Rotate Direction by -5 Degrees
17. position = position + Direction*Step
18. end for each Object loop
19. Generation = Generation + 1
20. Loop until termination condition is reached

6. CONCLUSIONS

This article designed four unique Swarm Intelligence algorithms CMRA, RBSA, ELSA, and OSOA. CMRA is inspired by Cricket match. RBSA is inspired by Rice Bags Sales. ELSA is inspired by Sentences in English. The final algorithm OSOA belongs to Object Swarm Intelligence. Objects move in search space in OSOA algorithm. The first three algorithms CMRA, RBSA and ELSA belong to Human Swarm Optimization field. Artificial Humans move in search space in these three algorithms.

REFERENCES

- [1] Mehrdad Rostami, Kamal Berahmand, Elahe Nasiri, Saman Forouzandeh, Review of swarm intelligence-based feature selection methods, *Engineering Applications of Artificial Intelligence*, Volume 100, 2021,104210, ISSN 0952-1976, <https://doi.org/10.1016/j.engappai.2021.104210>. (<https://www.sciencedirect.com/science/article/pii/S0952197621000579>)
- [2] J. Tang, G. Liu and Q. Pan, "A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends," in *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 10, pp. 1627-1643, October 2021, doi: 10.1109/JAS.2021.1004129.
- [3] Chao WANG, Shuyuan ZHANG, Tianhang MA, Yuetong XIAO, Michael Zhiqiang CHEN, Lei WANG, Swarm intelligence: A survey of model classification and applications, *Chinese Journal of Aeronautics*, Volume 38, Issue 3, 2025, 102982, ISSN 1000-9361, <https://doi.org/10.1016/j.cja.2024.03.019>. (<https://www.sciencedirect.com/science/article/pii/S1000936124000931>)
- [4] Nguyen, L. V. (2024). Swarm Intelligence-Based Multi-Robotics: A Comprehensive Review. *AppliedMath*, 4(4), 1192-1210. <https://doi.org/10.3390/appliedmath4040064>
- [5] Chinglemba, T., Biswas, S., Malakar, D., Meena, V., Sarkar, D., Biswas, A. (2023). Introductory Review of Swarm Intelligence Techniques. In: Biswas, A., Kalayci, C.B., Mirjalili, S. (eds) *Advances in Swarm Intelligence*. Studies in Computational Intelligence, vol 1054. Springer, Cham. https://doi.org/10.1007/978-3-031-09835-2_2