

Modified Crow Search Algorithm for Protein Structure Prediction

Hesham Awadh Bahamish¹, Naziha Mohammed Al-Aidroos², Aziz Nasser Boraik³

¹Assistant professor, Computer Science Department, College of Computers and Information Technology, Hadhramout University, Yemen

²Associate professor, Computer Science Department, College of Computers and Information Technology, Hadhramout University, Yemen

³Assistant professor, Information Technology Department, Faculty of Computing and Information Technology, University of Science and Technology, Hadhramout Branch, Yemen

Abstract - This paper investigated the capability of the Crow Search Algorithm to solve the protein structure prediction problem (CSA-PSP) and proposed a modification to the algorithm (MCSA-PSP) to enhance its performance. Crow search algorithm is a nature-based optimization algorithm. It is inspired by the manner of the crow flocks in nature. Three modifications to the original algorithm were proposed which are: 1) Using random genetic crossover for generating new solutions, 2) using the Metropolis criteria in accepting the new solutions and 3) a new method for generating new solutions. The performance of the original and modified algorithm was tested on a well-known benchmark protein (Met-enkephalin). The results showed that the algorithm was able to find the reported lowest free energy conformation and that the introduced modifications enhanced the performance of the original algorithm.

Key Words: Crow Search Algorithm, Protein Structure Prediction, Optimization, Conformation Search

1. INTRODUCTION

Living organisms are made of small molecules called proteins. These proteins are built of different combinations of amino acids. The amino acids sequence or chain is known as the primary structure that is folded due to amino acid interactions into the secondary structure and then into the tertiary structure which is the active state or biological active state. Quaternary protein structure is formed of more than one protein chain.

Proteins perform the biological functions in the human body. The importance of the protein structure comes from the fact that the protein cannot perform its biological function unless it folds into the native state (i. e. tertiary structure) [1]. Therefore, in order to know the protein function, it is necessary to reveal its tertiary structure first.

The protein structure prediction (PSP) problem is one of the most complex problems in Structural Biology, Computational Biology, and Bioinformatics [2-5]. Protein structure can be determined experimentally and computationally. The current computational prediction methods are classified into three classes. The Homology Modelling, Threading, and Ab initio

In the Ab initio method, the structure is predicted based on amino acids sequence. This is based on the thermodynamic hypothesis stated by Anfinsen [1]. In this hypothesis, the protein structure or native state is the structure, which has the lowest free energy. So to predict the protein structure computationally, the problem is formulated into an optimization problem where the goal is to search the problem search space to locate that lowest free energy structure (conformation).

Exploring the protein search space is an NP hard problem due to the gigantic number of possible solutions. A huge number of optimization algorithms were applied to solve this problem such as Genetic Algorithms (GA) [6, 7], Simulated Annealing (SA) [8], Bat Algorithm (BA) [9], Marriage in Honey Bee algorithm (MBO) [10], Harmony Search (HS) [11], Bees Algorithm [12] Artificial Bee Colony (ABC) [13, 14] to name some.

Like other algorithms, which are inspired by nature, the Crow Search Algorithm (CSA) is a population-based optimization algorithm, which is inspired by the foraging behavior of crows. It is considered as is one of the most recently developed metaheuristic algorithms [15-17]. Askarzadeh proposed it in 2016 and applied it to solve continuous optimization problems, and since that time it was extensively applied to solve many optimization and real-world problems [18]. It showed an enhancement over other state-of-art algorithms in solving constrained engineering design problems [19].

Previous works reported the CSA advantages such as simplicity and ease of programming [16, 19], using only two variable parameters (flight length and awareness probability) [20, 21], the fast convergence rate [20], and good and acceptable results for solving optimization and real-world problems. Despite these advantages, other researchers reported some disadvantages of the algorithm. These disadvantages are the premature convergence [16, 19, 22], trapping in local minima [18], the poorly exploration capability of the algorithm for certain types of hard problems [16, 18] and in some circumstances it may fail to find the optimal solution [19]. Therefore, many modifications to the original algorithm had been proposed to enhance its performance.

Khalilpourazari and Pasandideh proposed a hybrid crow sine-cosine algorithm SCCSA [23] to enhance the exploration and exploitation of the CSA algorithm. The hybrid algorithm

was evaluated in unimodal, multimodal, fixed-dimensional multimodal, and composite benchmark functions.

Hassanien proposed a hybrid crow search algorithm (RCSA) by combining CSA and rough searching scheme (RSS) and dynamic flight length to improve the algorithm exploration process. By this hybridization, the convergence of the algorithm was improved and the algorithm was able to get out of the local minima trap. The algorithm was applied to solve engineering optimization problems. The advantage of the hybridized algorithm was the exploitation of the favorable areas in the search space that may contain the global optimal solutions [24].

CSA was hybrid with space transform search (STS) algorithm [18] to enhance the CSA diversification capability and was applied to solve IEEE CEC 2017 test functions [18].

Han et al. proposed ISCSA algorithm, in which the CSA was improved by adding three modifications. These modifications were: introducing the spiral search mechanism, a weight coefficient and Gaussian variation, and random perturbation approach [16]

To speed up the algorithm and enhance the value of the solutions, Allaoui et al. combined the CSA with a local search method to solve the DNA fragment assembly problem [25].

Díaz et al. modified the awareness probability and the random perturbation. The modified algorithm was applied to solve complex optimization energy problems [15].

Jain et al. added an experience factor, adaptive modification operator, and Levy flight to balance the CSA exploration and exploitation abilities and prevent premature convergence [19].

Fallah et al. proposed ICSA [26]. In this algorithm, the crow with the worst function value was replaced by a newly generated solution.

In this paper, the CSA algorithm was applied to solve the PSP and three modifications were proposed to enhance the performance of the CSA algorithm. These modifications are using random genetic crossover for generating new solutions, using the Metropolis criteria in accepting the new solutions, and a new method for generating new solutions. The remainder of the paper is organized as follows: Section 2 presents the crow in nature. Section 3 describes the standard crow search algorithm. Section 4 describes the crow search algorithm for protein structure prediction and the proposed modifications to CSA. Section 5 contains the experiments and algorithm implementation. Section 6, presents and discusses the results. Finally, the paper is concluded in Section 7.

2. Crow in Nature

The crow is an intelligent creature. It is considered one of the most intelligent birds [25-27]. It has a large brain size as compared to its body size [22]. It is the same relative size as

the chimpanzee brain [28]. Besides that, the crow has a brilliant memory [17] that enables it to:

- Remember its hidden food places, which can be from last season [29].
- Recognize the faces of humans.
- Modifies its migration route to get around dangerous areas where the crow had been attacked by humans or other animals.
- Retrieve other information that it uses in the future.

The intelligence of the crow is considered as a seven-year-old human child intelligence [30]. The intelligent behavior of the crow is clearly noticed in the daily activities that the crow performs in the nature environment. Scientists studied the crow behavior in its environment and revealed remarkable intelligent characters [27].

Crow plans to get its food [31]. It adapts to the changes in its environment. Hence it uses tricks, exploits other birds, and makes some kind of tools that are used later to ease the process of eating, getting food, or solving problems [27]. Crow communicates with other crows through some complicated ways [26, 27, 32, 33] which could be considered some kind of language [29].

Crow is a greedy bird so it tries to find food by observing other crows; it does that to know their hidden food storages to acquire better food [24, 32]. This task is not as simple as it seems since the crow is always alert of other crows watching it. In consequence, it changes its food storages or pretends that it is keeping its food in other fake places [18, 32]. It uses previous knowledge to predict the behavior of other crows [32, 34, 35].

The food foraging behavior of the crow can be described as following [32]:

1. Crow is a social bird, which lives in groups called a flock.
2. Crow has the ability to remember where it hides its food.
3. Crow tries to reveal the other crows hiding places to steal their food
4. In order to protect its food place, crow uses a probability.

3. Crow Search Algorithm

In this section, the CSA is described in detail. The algorithm is based on the natural intelligent behavior of the crow in getting and keeping its food away from pilferers. In this algorithm, N is the number of crows and the food sources are the possible solutions to the problem, which form the problem search space. Each solution consists of PV problem variables or problem dimensions. AP is the awareness probability, FL is the flight length, and MAX_ITER is the maximum number of iterations. In order to store the good solution in each iteration of the algorithm, each crow has its own memory C_tMEM_i , where t is the current iteration and i is the index of the crow.

The steps of the algorithm are outlined as the following phases:

Phase one (initial phase):

1. For 1 to N crow, an initial food source (solution) is generated (randomly or based on problem domain)
2. Each generated food source is evaluated using problem-specific evaluation function.
3. Each crow stores (memories) its initial food source in its memory ($C_{t_MEM_i}$).
4. Set the values of AP and FL parameters.

Phase two (optimization process):

In this, phase for MAX_ITER iterations, each crow i has to decide based on a randomly generated value ar to be in one state of the following states:

State 1: Follow a random crow j to steal its food source (solution) if ar is greater than or equal AP eq. 1.

$$C_{t+1,i} = C_{t,i} * ar_i * FL * (C_{t_MEM_j} - C_{t,i}), \quad \text{if } ar \geq AP \quad (1)$$

State 2: Generate a new food source randomly eq. 2.

$$C_{t+1,i} = \text{random food source} \quad (2)$$

After that, the generated food sources are checked for visibility. Only the visible food sources are evaluated using the evaluation function and the crow's memory is updated as the following:

$$C_{t_MEM_i} = \left\{ \begin{array}{ll} C_{t+1,i}, & \text{if } C_{t+1,i} \text{ is better than } C_{t_MEM_i} \\ C_{t_MEM_i} & \text{otherwise} \end{array} \right\} \quad (3)$$

This continues until the MAX_ITER is reached and the food source with the best value is selected as the optimal solution of the problem. The following pseudo code describe the standard CSA.

```

Phase one (initial phase):
Initialize the food sources (solutions).
Evaluate the generated food sources.
Memorize the initial food source in its memory (Ct\_MEMi).
Set the values of AP and FL parameters
Phase two (optimization process)
While t < MAX_ITER
for i = 1 : N (all N crows of the flock)
    Randomly choose one of the crows to follow e.g. j
    If ar >= AP
        Ct+1,i = Ct,i * ari * FL * (Ct\_MEMj - Ct,i)
    else
        Ct+1,i = a random food source
    end if
end for
Check the feasibility of new solutions
Evaluate the new position of the crows
Update the memory of crows using (3)
End while
    
```

4. CROW SEARCH ALGORITHM FOR PROTEIN STRUCTURE PREDICTION

In this section, the CSA is applied to predict the tertiary structure of the protein, this algorithm is called (CSA-PSP), and then we introduced three modifications to the algorithm to enhance its performance. The modified algorithm is called (MCSA-PSP).

In the CSA-PSP, each crow's food source is considered as a protein conformation (solution to the problem). This conformation is consists of a set of torsion angles TN which is the dimension of the search space. Phase one begins by setting the values of the parameters: FL, AP, and generating random protein conformations by setting random values in the range [-II, II] to each torsion angles. After that, these conformations are evaluated by a fitness function (energy function) to determine the energy of the generated conformations.

These initialized conformations are saved in the memory of each crow. After initializing the conformations and crows memories, phase two starts. For MAX_ITER iterations, each crow i has to decide based on a randomly generated value ar to be in one of the following states:

State 1: Follow a random crow j to steal its food source (solution) if ar is greater than or equal AP eq. 4.

$$Conf_{t+1,i} = Conf_{t,i} * ar_i * FL * (Conf_{t_MEM_j} - Conf_{t,i}), \quad \text{if } ar \geq AP \quad (4)$$

State 2: Generate a new food source randomly eq. 5.

$$Conf_{t+1,i} = \text{random food source} \quad (5)$$

After that, the generated conformations checked for visibility. Only the visible conformations are evaluated using the evaluation function and the crow's memory is updated as the following:

$$C_{t_MEM_i} = \left\{ \begin{array}{ll} Conf_{t+1,i}, & \text{if } (Conf_{t+1,i} \text{ is better than } Conf_{t_MEM_i}) \\ Conf_{t_MEM_i} & \text{otherwise} \end{array} \right\} \quad (6)$$

This continues until the MAX_ITER is reached and the conformation with the lowest free energy value is selected as the optimal solution of the problem.

4.1 Modified CSA-PSP

Three modifications to the CSA are introduced. The first is generate the new conformations based on a random genetic cross over instead of using eq. 1, the second, is accepting or rejecting the newly generated conformations based on Metropolis criteria, and the third is a new method for generating new random solutions. In the following subsections, these modifications are described.

4.1.1 Modification No. 1: Genetic Crossover:

In this modification instead of generating the new food source using eq. 1 for all problem variables, we modified it by performing a genetic crossover between the selected solution from the crow memory and the current solution of the crow

(fig. 1). This is performed by splitting the solution variables indexes randomly into two groups.

The values of the variables in group one are taken from the chosen solution from the crow memory (C_{t, MEM_j}) and the values of the variables in group two are taken from the current solution ($C_{t,i}$).

V1	V2	V3	V4	V5	V6	V7	V8
G1	G2	G2	G1	G2	G2	G1	G2

V1	V2	V3	V4	V5	V6	V7	V8
C_{ti}	C_{tMEM_j}	C_{tMEM_j}	C_{ti}	C_{tMEM_j}	C_{tMEM_j}	C_{ti}	C_{tMEM_j}

Fig -1: Generating Solution by Crossover

4.1.2 Modification No. 2: Metropolis criteria:

In CSA and all its variant modifications, the new food source is saved in the crow memory if it is better than the previously saved value. This is a greedy selection criterion. Using the Metropolis criterion [36] enable the algorithm to escape from the local minima traps by accepting low-quality solutions, which allows the overcoming of barriers and finding better solutions.

In this modification, a form of Metropolis criterion is used to accept or reject the replacement of the crow's memory by the new food source when the food source is worse than the crow's memory value. If the value of the food source is better it is accepted and replaced the old value in the crow's memory. Otherwise, it accepted if:

$$R \geq \exp(-(C_{t,i} - C_{t, MEM_i})) \tag{7}$$

Where R is a generated random number in [0,1].

4.1.2 Modification No. 3: Solution Generation:

In this modification, the new solution is generated as following:

$$\begin{aligned} &\alpha = r1 * 0.5 \\ &\text{if } r2 < 0.5 \\ &\quad \text{new} = \max_dim + \alpha; \\ &\text{else} \\ &\quad \text{new} = \max_dim - \alpha; \end{aligned}$$

While r1 and r2 are randomly generated numbers and max_dim is the maximum value of the problem dimension.

5. EXPERIMENTS AND ALGORITHM IMPLEMENTATION

Table I shows the specification of the machine, programming languages, and tools which were used in the implementation of the algorithm.

Table I Specifications of Algorithm Implementation

Processor	i7-3740QM CPU@ Intel® Core (TM)
Speed	2.70 GHz
Memory	16 GB
Programming language	Visual C++
Operating System	Windows 10

The algorithm was tested using Met-enkephalin, a benchmark protein. Its amino acids sequence is (Tyr-Gly-Gly-Phe-Met) which contains 24 main and side-chain torsion angles. For the evaluation of the generated solutions, the ECEPP/3 [37, 38] function was used

In all experiments, the values of parameters were set as following: $AP=0.2$, $FL=2$, $N=5$, and the number of iterations = 5000. Table II shows the settings of the eight experiments. In experiment 1 the standard CSA was used. In experiment 2, modification No. 1 was used. In Experiments 3, modification No. 2 was used. In experiment 4, modification No. 3 was used. In experiment 5, modification No 1 and 2 were used. In experiment 6, modification No 1 and 3 were used. In experiment 7, modification No 2 and 3 were used. In experiment 8, the three modifications were used.

Table II Experiments Settings

Experiment No.	Modification No. 1	Modification No. 2	Modification No. 3
1	No	No	No
2	Yes	No	No
3	No	Yes	No
4	No	No	Yes
5	Yes	Yes	No
6	Yes	No	Yes
7	No	Yes	Yes
8	Yes	Yes	Yes

6. RESULTS AND DISCUSSIONS

Table III shows the results of 30 independent runs of the eight experiments. The best, average, worst energies, standard deviation, and success rate are listed. In all experiments, the algorithm was able to find the reported lowest free energy solution (-12.43 kcal/mol).

Table III: Results of the Eight Experiments

No.	Best	worst	Average	STD	Success rate
1	-12.43	-8.89	-11.54	0.78	27%
2	-12.43	-9.34	-11.59	0.73	30%
3	-12.43	-9.81	-11.66	0.71	33%
4	-12.43	-10.15	-11.50	0.71	27%
5	-12.43	-9.78	-11.51	0.75	30%
6	-12.43	-10.51	-12.03	0.61	66%
7	-12.43	-10.60	-12.05	0.50	60%
8	-12.43	-11.27	-12.27	0.37	83%

The performance of the standard CSA was slightly improved in terms of success rate from 27% to 30% when modification no 1 was used and to 33% when modification no 2 was used, while there was no improvement when modification no 3 was used.

The combination of modification no 1 and 2 improve the success rate of the standard CSA to 30% while a significant improvement was gained when the combination of modification no 1 and 3 (%66), and the combination of modification no 2 and 3 (%60) were used.

The effect of the combination of the three modifications is clearly shown in the results of experiment 8. Which gained the best results in terms of success rate, average energy, and standard deviation.

The convergence of the algorithm in the eight experiments is shown in chart 1. It has clearly shown the effects of the modifications on the performance of the algorithm.

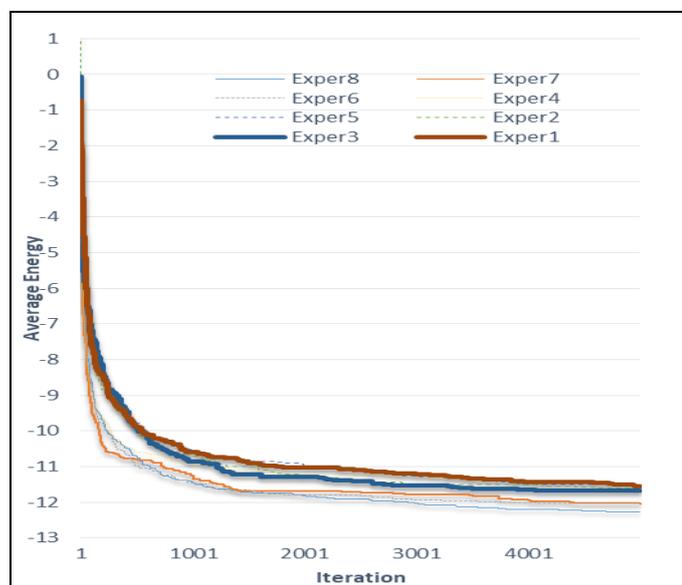


Chart -1: The Convergence of the Eight Experiments

Table V compares the CSA algorithm with other optimization algorithms applied to solve the PSP problem.

Table V Comparison with Other Algorithms

Algorithm	Best energy
CSA	-12.43
BA [9]	-12.43
MOLS [39]	-10.10
Basin paving [40]	-12.43
TS [41]	-12.39
Conformation Space Annealing [8]	-11.71
MBO [10]	-12.43
HHSA [11]	-12.43

Modified ABC [13]	-11.77
DCSaDE-LS [42]	-12.43
PTSPGA [43]	-9.55
SaDE [6]	-12.43
GA [7]	-12.43

7. CONCLUSIONS

In this paper, the capability of the CSA algorithm to solve the PSP problem was investigated. The algorithm was used as an optimization algorithm to solve the PSP problem. Three modifications to the original CSA algorithm were proposed. The original and the modified algorithms were able to find the reported lowest free energy conformation of the test protein. The proposed modifications enhanced the performance of the algorithm and the significant improvement was when the three modifications were used. Future work will be a study of the algorithm performance on longer protein size.

REFERENCES

- [1] C. B. Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, no. 96, pp. 223-230, 1973. [Online]. Available: <http://view.ncbi.nlm.nih.gov/pubmed/4124164>
- [2] M. Kanehisa, "Grand challenges in bioinformatics," *Bioinformatics*, vol. 14, no. 4, p. 309, May 1, 1998 1998, doi: 10.1093/bioinformatics/14.4.309.
- [3] J. Meidanis, "Current Challenges in Bioinformatics," in *String Processing and Information Retrieval*: Springer Berlin / Heidelberg, 2003, pp. 16-27.
- [4] G. Helles, "A comparative study of the reported performance of ab initio protein structure prediction algorithms," *JR Soc Interface*, vol. 5, no. 21, pp. 387-396, April 6, 2008 2008, doi: 10.1098/rsif.2007.1278.
- [5] E. Keedwell and A. Narayanan, *Intelligent Bioinformatics: The Application of Artificial Intelligence Techniques to Bioinformatics Problems*. John Wiley & Sons, 2005.
- [6] S. Sudha, S. Baskar, and S. Krishnaswamy, "Protein Tertiary Structure Prediction Using Evolutionary Algorithms," *International Journal of Emerging Technologies in Computational and Applied Sciences*, vol. 3, no. 3, p. 11, 2013.
- [7] G. Sindhu and S. Sudha, "Prediction of Protein Tertiary Structure Using Genetic Algorithm," in *Soft Computing Techniques in Vision Science*, S. Patnaik and Y.-M. Yang Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 15-22.

- [8] J. Lee and H. A. Scheraga, "Conformational space annealing by parallel computations: Extensive conformational search of Met-enkephalin and of the 20-residue membrane-bound portion of melittin," *International Journal of Quantum Chemistry*, vol. 75, no. 3, p. 11, 1999. [Online]. Available: [http://dx.doi.org/10.1002/\(SICI\)1097-461X\(1999\)75:3<255::AID-QUA15>3.0.CO;2-V](http://dx.doi.org/10.1002/(SICI)1097-461X(1999)75:3<255::AID-QUA15>3.0.CO;2-V)
- [9] H. A. Bahamish, N. M. Al-Aidroos, and A. N. Boraik, "Bat Algorithm for Protein Conformational Search," in 2019 First International Conference of Intelligent Computing and Engineering (ICOICE), 15-16 Dec. 2019 2019, pp. 1-7, doi: 10.1109/ICOICE48418.2019.9035182.
- [10] H. A. A. Bahamish, R. Abdullah, and M. A. Abu-Hashem, "A modified Marriage in Honey Bee Optimisation (MBO) algorithm for protein structure prediction," in The 2nd International Conference on Computer Technology and Development, 2-4 Nov. 2010 2010, pp. 65-69, doi: 10.1109/ICCTD.2010.5646067.
- [11] M. S. Abual-Rub, M. A. Al-Betar, R. Abdullah, and A. T. Khader, "A hybrid harmony search algorithm for ab initio protein tertiary structure prediction," *Network Modeling Analysis in Health Informatics and Bioinformatics*, journal article vol. 1, no. 3, pp. 69-85, September 01 2012, doi: 10.1007/s13721-012-0013-7.
- [12] H. A. A. Bahamish, R. Abdullah, and R. A. Salam, "Protein Conformational Search Using Bees Algorithm," presented at the Proceedings of the 2008 Second Asia International Conference on Modelling & Simulation (AMS), 2008.
- [13] Z. Alqattan and R. Abdullah, "Artificial Bee Colony Optimization Algorithm with Crossover Operator for Protein Structure Prediction," *Lecture Notes in Mathematics -Springer-verlag-*, vol. 378, pp. 147-157, 08/28 2013, doi: 10.1007/978-3-642-40567-9_13.
- [14] H. A. A. Bahamish, R. Abdullah, and R. A. Salam, "Protein Tertiary Structure Prediction Using Artificial Bee Colony Algorithm," in 2009 Third Asia International Conference on Modelling & Simulation, 25-29 May 2009 2009, pp. 258-263, doi: 10.1109/AMS.2009.47.
- [15] P. Díaz et al., "An Improved Crow Search Algorithm Applied to Energy Problems," *Energies*, vol. 11, no. 3, p. 571, 2018. [Online]. Available: <https://www.mdpi.com/1996-1073/11/3/571>.
- [16] X. Han, Q. Xu, L. Yue, Y. Dong, G. Xie, and X. Xu, "An Improved Crow Search Algorithm Based on Spiral Search Mechanism for Solving Numerical and Engineering Optimization Problems," *IEEE Access*, vol. 8, pp. 92363-92382, 2020, doi: 10.1109/ACCESS.2020.2980300.
- [17] P. K. K R and K. Kittusamy, "Amelioration of task scheduling in cloud computing using crow search algorithm," *Neural Computing and Applications*, 02/06 2019, doi: 10.1007/s00521-019-04067-2.
- [18] S. K. Majhi, M. Sahoo, and R. Pradhan, "A space transformational crow search algorithm for optimization problems," *Evolutionary Intelligence*, vol. 13, no. 3, pp. 345-364, 2020/09/01 2020, doi: 10.1007/s12065-019-00294-7.
- [19] M. Jain, A. Rani, and V. Singh, "An improved Crow Search Algorithm for high-dimensional problems," *Journal of Intelligent & Fuzzy Systems*, vol. 33, pp. 3597-3614, 2017, doi: 10.3233/JIFS-17275.
- [20] A. K. Bhullar, R. Kaur, and S. Sondhi, "Enhanced crow search algorithm for AVR optimization," *Soft Computing*, vol. 24, no. 16, pp. 11957-11987, 2020/08/01 2020, doi: 10.1007/s00500-019-04640-w.
- [21] K. M. Dhanya and S. Kanmani, "Performance Evaluation of Crow Search Algorithm on Capacitated Vehicle Routing Problem," in *Soft Computing Systems*, Singapore, I. Zelinka, R. Senkerik, G. Panda, and P. S. Lekshmi Kanthan, Eds., 2018// 2018: Springer Singapore, pp. 91-98.
- [22] G. I. Sayed, A. E. Hassanien, and A. T. Azar, "Feature selection via a novel chaotic crow search algorithm," *Neural Computing and Applications*, vol. 31, no. 1, pp. 171-188, 2019/01/01 2019, doi: 10.1007/s00521-017-2988-6.
- [23] S. Khalilpourazari and S. H. R. Pasandideh, "Sine-cosine crow search algorithm: theory and applications," *Neural Computing and Applications*, vol. 32, no. 12, pp. 7725-7742, 2020/06/01 2020, doi: 10.1007/s00521-019-04530-0.
- [24] A. E. Hassanien, R. M. Rizk-Allah, and M. Elhoseny, "A hybrid crow search algorithm based on rough searching scheme for solving engineering optimization problems," *Journal of Ambient Intelligence and Humanized Computing*, 2018/06/25 2018, doi: 10.1007/s12652-018-0924-y.
- [25] M. Allaoui, B. Ahiod, and M. El Yafrani, "A hybrid crow search algorithm for solving the DNA fragment assembly problem," *Expert Systems with Applications*, vol. 102, pp. 44-56, 2018/07/15/ 2018, doi: <https://doi.org/10.1016/j.eswa.2018.02.018>.
- [26] H. Fallah, O. Kisi, S. Kim, and M. Rezaie-Balf, "A New Optimization Approach for the Least-Cost Design of Water Distribution Networks: Improved Crow Search

- Algorithm," *Water Resources Management*, vol. 33, no. 10, pp. 3595-3613, 2019/08/01 2019, doi: 10.1007/s11269-019-02322-8.
- [27] N. J. Emery and N. S. Clayton, "The Mentality of Crows: Convergent Evolution of Intelligence in Corvids and Apes," *Science*, vol. 306, no. 5703, p. 1903, 2004, doi: 10.1126/science.1098410.
- [28] N. Clayton and N. Emery. (2005, FEBRUARY 08, 2005) Corvid cognition. *Current Biology*. 2.
- [29] D. Gupta, J. J. P. C. Rodrigues, S. Sundaram, A. Khanna, V. Korotaev, and V. H. C. de Albuquerque, "Usability feature extraction using modified crow search algorithm: a novel approach," *Neural Computing and Applications*, vol. 32, no. 15, pp. 10915-10925, 2020/08/01 2020, doi: 10.1007/s00521-018-3688-6.
- [30] L. Hennefield, H. G. Hwang, and D. J. Povinelli, "Going Meta: Retelling the Scientific Retelling of Aesop's the Crow and the Pitcher," *Journal of Folklore Research*, vol. 56, no. 2-3, pp. 45-70, 2019. [Online]. Available: <https://www.jstor.org/stable/10.2979/jfolkrese.56.2.3.04>.
- [31] J. H. Wimpenny, A. A. Weir, L. Clayton, C. Rutz, and A. Kacelnik, "Cognitive processes associated with sequential tool use in New Caledonian crows," (in eng), *PloS one*, vol. 4, no. 8, p. e6471, Aug 5 2009, doi: 10.1371/journal.pone.0006471.
- [32] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," *Computers & Structures*, vol. 169, pp. 1-12, 06/01 2016, doi: 10.1016/j.compstruc.2016.03.001.
- [33] E. Cuevas, E. Barocio, and A. Conde, "A Modified Crow Search Algorithm with Applications to Power System Problems," 2019, pp. 137-166.
- [34] B. Dey, B. Bhattacharyya, A. Srivastava, and K. Shivam, "Solving energy management of renewable integrated microgrid systems using crow search algorithm," *Soft Computing*, vol. 24, no. 14, pp. 10433-10454, 2020/07/01 2020, doi: 10.1007/s00500-019-04553-8.
- [35] G. I. Sayed, A. E. Hassanien, A. Gamal, and H. A. Ella, "An Automated Fish Species Identification System Based on Crow Search Algorithm," in *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018)*, Cham, A. E. Hassanien, M. F. Tolba, M. Elhoseny, and M. Mostafa, Eds., 2018// 2018: Springer International Publishing, pp. 112-123.
- [36] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of State Calculations by Fast Computing Machines," *The Journal of chemical physics*, vol. 21, no. 6, pp. 1087-1092, 1953. [Online]. Available: <http://link.aip.org/link/?JCP/21/1087/1>.
- [37] F. Eisenmenger, U. H. E. Hansmann, S. Hayryan, and C.-K. Hu, "An enhanced version of SMMP—open-source software package for simulation of proteins," *Computer Physics Communications*, vol. 174, no. 5, pp. 422-429, 03/2006 2006.
- [38] F. Eisenmenger, U. H. E. Hansmann, S. Hayryan, and C.-K. Hu, "[SMMP] A modern package for simulation of proteins," *Computer Physics Communications*, vol. 138, pp. 192-212, 2001. [Online]. Available: [http://dx.doi.org/10.1016/S0010-4655\(01\)00197-7](http://dx.doi.org/10.1016/S0010-4655(01)00197-7)
- [39] K. Vengadesan and N. Gautham, "A New Conformational Search Technique and Its Applications," *CURRENT SCIENCE*, vol. 88, no. 11, pp. 1760-1771, 10 June 2005 2005.
- [40] L. Zhan, J. Z. Y. Chen, and W.-K. Liu, "Conformational Study of Met-Enkephalin Based on the ECEPP Force Fields," *Biophysical Journal*, vol. 91, no. 7, pp. 2399-2404, October 1, 2006 2006, doi: 10.1529/biophysj.106.083899.
- [41] L. B. Morales, R. Garduno-Juarez, J. M. Aguilar-Alvarado, and F. J. Riveros-Castro, "A Parallel Tabu Search for Conformational Energy Optimization of Oligopeptides," *Journal of Computational Chemistry*, vol. 21, no. 2, pp. 147-156, 2000. [Online]. Available: [http://dx.doi.org/10.1002/\(SICI\)1096-987X\(20000130\)21:2<147::AID-JCC6>3.0.CO;2-6](http://dx.doi.org/10.1002/(SICI)1096-987X(20000130)21:2<147::AID-JCC6>3.0.CO;2-6).
- [42] S. Sudharsan, B. Subramanian, M. Amali, and S. Krishnaswamy, "Protein structure prediction using diversity controlled self-adaptive differential evolution with local search," *Soft Computing*, vol. 19, 06/01 2014, doi: 10.1007/s00500-014-1353-2.
- [43] B. Moghram, e. nabil, and A. Badr, "Ab-Initio Protein Tertiary Structure Prediction Using Genetic Algorithm," *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 18, pp. 26-32, 01/01 2016.