Neural Network Weight Optimization With Eagle Strategy

Gaurav C G
PES University, Outer Ring Rd, Banashankari 3rd Stage, Banashankari, Bengaluru, Karnataka 560085
Officialgauravcg429@gmail.com

Sanjay Chari
PES University, Outer Ring Rd, Banashankari 3rd Stage, Banashankari, Bengaluru, Karnataka 560085
Sanjaychari2xy2@gmail.com

Cyrus D'Lima
PES University, Outer Ring Rd, Banashankari 3rd Stage, Banashankari, Bengaluru, Karnataka 56008
cyrusdlima@gmail.com

***Abstract* - The traditional backprogation algorithm used in weight optimization in neural networks uses gradient descent, i.e., it calculates the gradient of the error function with respect to the neural network's weights, and adjusts the network weights accordingly in every epoch. The backpropagation method achieves good accuracy, but we observed that improvements can be made on the training time of the model. We propose an approach that utilizes eagle strategy for weight optimization to improve the training time of a neural network. Our approach utilises a method known as Lévy flight that performs a global search on weights, and then local search is performed on the neighbours of the weights that perform better than a given threshold accuracy. We utilise simulated annealing and random hill climbing to perform local search on the weights that pass the threshold. Our results show that training neural networks in this form achieves faster convergence.**

***Keywords* – Eagle Strategy, Neural Networks, Backpropagation, Levy Walk, Hill Climbing, Simulated Annealing**

I. INTRODUCTION

Eagle strategy is a two-stage method for optimization. It uses a combination of crude global search and intensive local search via a balance combination of different algorithms to suit different purposes.

In the problem of neural network weight optimization, we implement eagle strategy by implementing a crude global search via Levy flight, and then intensive local search on the weights that pass a given threshold, using simulated annealing and hill climbing.

A Lévy flight is a random walk in which the step-lengths have a probability distribution that is heavy-tailed. When defined as a walk in a space of dimension greater than one, the steps made are in isotropic random directions.

Simulated annealing is a probabilistic technique for approximating the global optimum of a given function. Specifically, it is a metaheuristic to approximate global optimization in a large search space for an optimization problem. For problems where finding an optimum in a short period of time is more important than the accuracy of the result,, simulated annealing works better than gradient descent.

Hill climbing is a mathematical optimization technique which belongs to the family of local search. It is an iterative algorithm that starts with an arbitrary solution to a problem, then attempts to find a better solution by making an incremental change to the solution. If the change produces a better solution, another incremental change is made to the new solution, and so on until no further improvements can be found. Like simulated annealing, hill c

climbing applied to the problem of local search in neural network weight optimization converges faster than gradient descent.

II. METHOD

Our method utilizes Lévy flight for global search on the neural network weights. The Levy flight algorithm returns a random list of weights that follow the Levy distribution , that can be observed below. The formula for the probability distribution is :

$$f(x;\mu,c)=\sqrt{\frac{c}{2\pi}}\;\frac{e^{-\frac{c}{2(x-\mu)}}}{(x-\mu)^{3/2}}$$

, where μ is the location parameter and c is the scale parameter.

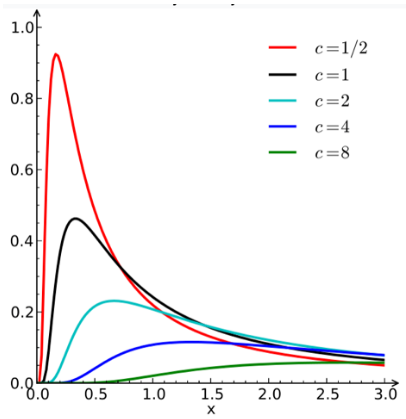

Figure 1 : Levy Probability Distribution

We test the accuracy of each of the randomly generated weights on the test set, and reject all those weights which achieve an accuracy of less than 0.5. The threshold value is set as 0.5 for global search as this is the bare minimum that a binary classifier is supposed to achieve in order to be accepted as a valid model.

On the weights that pass through our threshold , we apply intensive local search. We utilize simulated annealing and hill climbing to achieve this task.

Our simulated annealing approach starts with the weights that were generated from Levy flight. It explores neighbours of the given weight by adding and subtracting a given value from the weights. It then evaluates the neighbours by a fitness function and chooses the neighbour with the best fitness value for the next iteration. The number of weights

that produce a bad fitness function is reduced as the number of iterations, or as the temperature decreases.
We let simulated annealing run for around 300 iterations with a maximum of 50 attempts per iteration.
Our hill climbing approach takes the weights generated from Levy Flight, and looks at the neighbour of the given weights that produces the best fitness function, and moves to that neighbour; and repeats the same process until it reaches an optimum.

III. RESULTS

Our model was able to achieve considerable speedup when compared to gradient descent, with comparable accuracy as well.

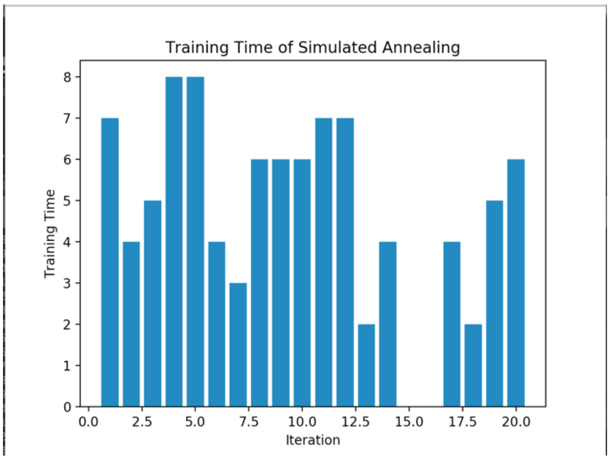

Figure 3 : Training Time of Eagle Strategy With SA

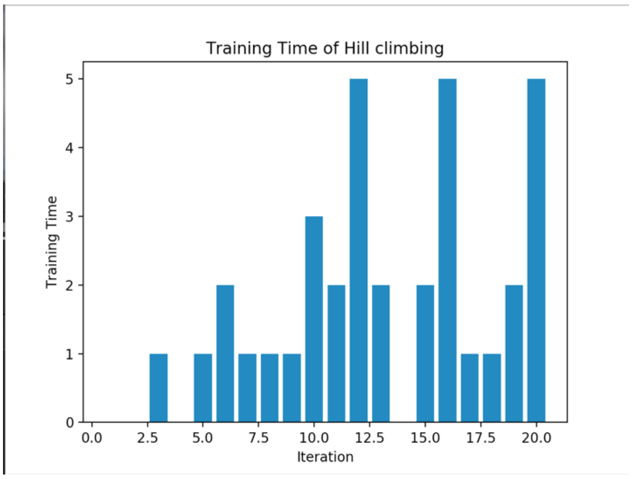

Figure 4 : Training Time of Eagle Strategy With Hill Climbing

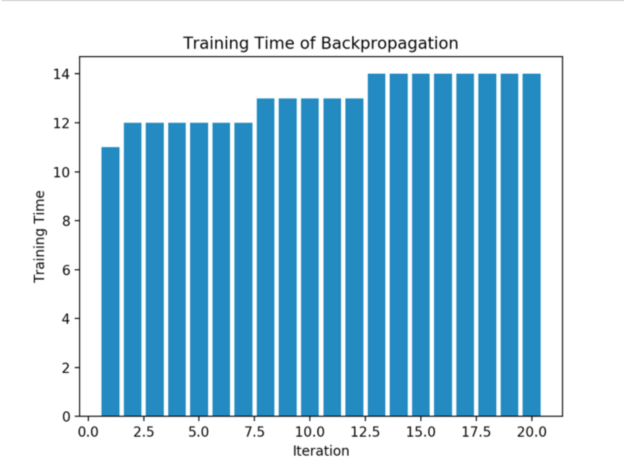

Figure 5 : Training Time of Eagle Strategy With Hill Climbing

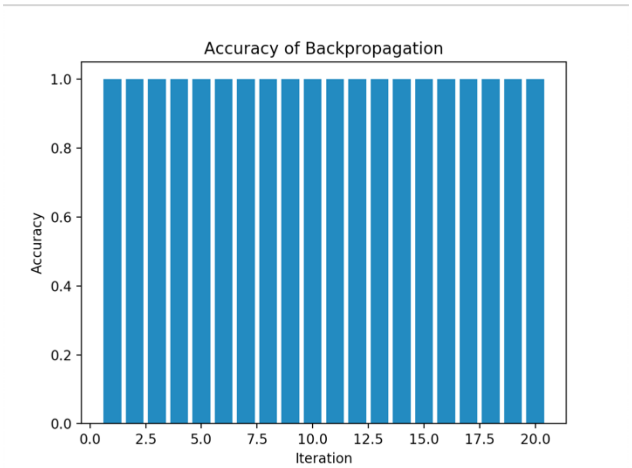

Figure 6 : Accuracy of Backpropagation

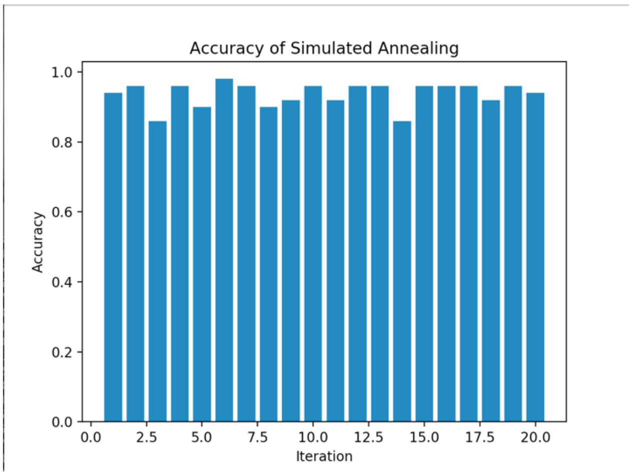

Figure 7 : Accuracy of Eagle Strategy With SA

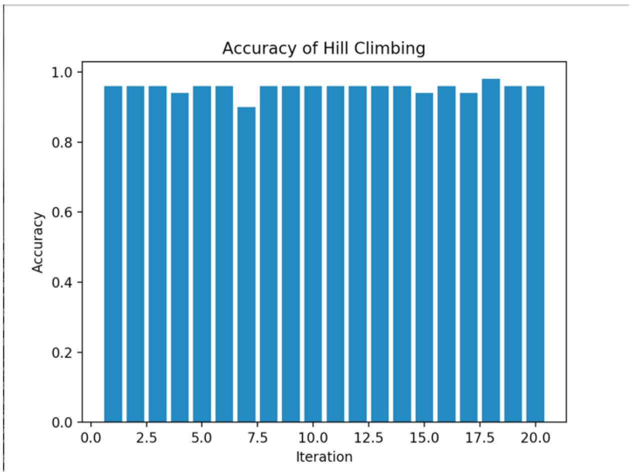

Figure 8 : Accuracy of Eagle Strategy With Hill Climbing

It can be observed from Figures 3 to 8, that eagle strategy provides a better training time when compared to backpropagation. In terms of accuracy, backpropagation outperforms eagle strategy. Thus, in applications where some amount of misclassification can be tolerated , but results are to be received quickly, eagle strategy would be an ideal fit. It should be noted that the term Iteration on the axis refers to the number of times global search and subsequent local search are being run, and each iteration does not receive any information from the previous iteration.

A summary of our results is described in the Figure 2.

Algorithm	Average Speed(in seconds)	Average Accuracy
Backpropagation	13	100%
Levy Walk + Simulated Annealing	6	95%
Levy Walk + Hill Climbing	4	96%

Figure 2 : Comparison of Results

It is evident from Figure 2 that our eagle strategy combined with hill climbing performs the classification in almost a quarter of the time that gradient descent takes to accomplish the task, while simulated annealing converges in approximately half the time as gradient descent.

IV. Conclusion

We apply Eagle Strategy to neural network optimization in an attempt to improve training time when compared with backpropagation. We achieve considerable speedup by using Levy flight for local search and simulated annealing and hill climbing for global search. This is useful in applications which require fast results.

The accuracy of our model is slightly worse than backpropagation. We leave any improvements that can be made on the accuracy to future work.

References

[1] Xin She Yang, Suash Deb, Xingshe He, Eagle Strategy With Flower Algorithm

[2] Benichou, O., Loverdo, C., Moreau, M., and Voituriez, R., Intermittent ´search strategies, Review of Modern Physics, 83, 81-129 (2011).

[3] Blum, C. and Roli, A. (2003) Metaheuristics in combinatorial optimisa-
tion: Overview and conceptural comparision, ACM Comput. Surv., Vol. 35, 268-308.

[4] Cagnina, L. C., Esquivel, S. C., and Coello, C. A., Solving engineering
optimization problems with the simple constrained particle swarm optimizer, Informatica, 32, 319-326 (2008).

[5] B. J. Copeland, The Essential Turing, Oxford University Press, 2004.

[6] Corne D. and Knowles, J., Some multiobjective optimizers are better than others, Evolutionary Computation, CEC'03, 4, 2506-2512 (2003).

[7] Dorigo, M. and Stutle, T., ¨ Ant Colony Optimization, MIT Press, (2004).