Progressive Gradient Walk for Neural Network Fitness Landscape Analysis

Anna S. Bosman
Department of Computer Science
University of Pretoria
Pretoria, South Africa
annar@cs.up.ac.za

Andries P. Engelbrecht
Department of Computer Science
University of Pretoria
Pretoria, South Africa
engel@cs.up.ac.za

Mardé Helbig
Department of Computer Science
University of Pretoria
Pretoria, South Africa
mhelbig@cs.up.ac.za

ABSTRACT

Understanding the properties of neural network error landscapes is an important problem faced by the neural network research community. A few attempts have been made in the past to gather insight about neural network error landscapes using fitness landscape analysis techniques. However, most fitness landscape metrics rely on the analysis of random samples, which may not represent the high-dimensional neural network search spaces well. If the random samples do not include areas of good fitness, then the presence of local optima and/or saddle points cannot be quantified. This paper proposes a progressive gradient walk as an alternative sampling algorithm for neural network error landscape analysis. Experiments show that the proposed walk captures areas of good fitness significantly better than the random walks.

CCS CONCEPTS

• **Computing methodologies** → **Continuous space search**; **Randomized search**; **Neural networks**; • **Mathematics of computing** → *Continuous functions*; • **Theory of computation** → Random walks and Markov chains;

KEYWORDS

neural networks, fitness landscape analysis, adaptive walk, random walk

ACM Reference Format:
Anna S. Bosman, Andries P. Engelbrecht, and Mardé Helbig. 2018. Progressive Gradient Walk for Neural Network Fitness Landscape Analysis. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3205651.3208247

1 INTRODUCTION

Neural networks (NNs) are mathematical models capable of representing an arbitrary non-linear mapping from inputs to outputs [1, 8]. Due to their non-linear information capacity, NNs have enjoyed unprecedented success in application areas such as image and speech recognition [6, 17], sequence modelling [30], and function approximation [12], amongst others. However, despite the practical success of NNs, certain theoretical properties of these models remain poorly understood. Specifically, the landscape properties of the objective functions associated with supervised NN training are hard to quantify and visualise due to the inherent high dimensionality of the search space [4, 25]. As a result, the influence of various NN parameters on the resulting error surface remains unknown.

Fitness landscape analysis (FLA) provides an excellent means of studying NN error landscapes under various parameter settings and algorithm contexts. FLA comprises of a large set of techniques designed to capture and quantify significant topographical features of fitness landscapes such as ruggedness, neutrality, modality, dispersion, and searchability [18, 24]. The FLA techniques can be used to better understand the problem at hand, and to aid the process of algorithm selection and dynamic parameter adaptation [13, 18].

The concept of FLA comes from the evolutionary context, and most metrics are defined for discrete search spaces [14, 21]. Fitness landscape properties are estimated by taking random samples of the search space, calculating the objective function value for every point in each sample, and analysing the relationship between the spatial and the qualitative characteristics of the sampled points. This concept can easily be translated to continuous search spaces, so long as an adequate sampling method is defined [20]. FLA of continuous fitness landscapes has attracted a significant amount of research recently [18, 19, 22, 29].

NN training is a continuous optimisation problem, which can be studied using FLA. The search space of a NN is made up of all possible real-valued weight combinations, where each weight combination corresponds to a certain measure of error. Training algorithms seek to minimise the error by searching for an optimal weight combination. Thus, the landscape of the error function defined in terms of the weights constitutes the fitness landscape, also referred to as the error landscape. Several studies have been conducted, showing FLA to be a useful tool for analysis and visualisation of the NN error surfaces [2, 3, 25, 31].

However, an earlier study by Smith *et al* [26] has shown that, given a difficult optimisation problem with a limited number of good fitness areas, random sampling may fail to capture the unique features of the landscape. Indeed, if the random sample fails to capture any points of good fitness, it will not be possible for the FLA metrics to correctly quantify properties such as modality and the presence of saddle points. In the context of NNs, quantifying local optima and saddle points is of very high interest, since there

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07…$15.00
https://doi.org/10.1145/3205651.3208247

is theoretical evidence for the prevalence of the latter over the former [5, 11], but no empirical evidence exists to date.

This study proposes a randomised gradient sampling technique based on the progressive random walk [20]. The proposed algorithm uses the error function gradient to choose the general direction of each step. The magnitude of the step is randomised within a closed interval per dimension, thus introducing stochasticity. Therefore, the proposed sampling is biased towards the search space areas that contain high fitness solutions. The added stochasticity makes the sampling general enough to not be algorithm-specific, and allows for the coverage of poor fitness as well as good fitness areas.

The rest of the paper is structured as follows: Section 2 discusses the problem of random and adaptive sampling in continuous search spaces, and proposes the progressive gradient walk for neural network fitness landscape analysis. Section 3 presents the empirical study of the proposed gradient walk compared to two random walks commonly used in the FLA literature. Section 4 concludes the paper and outlines possible directions for future work.

2 RANDOM WALKS

This section describes the concept of a random walk, and relates this notion to fitness landscape analysis. Section 2.1 discusses the random walk definition, Section 2.2 discusses the concept of adaptive walk in discrete spaces, Section 2.3 discusses how random and adaptive walks translate to continuous search spaces, and Section 2.4 proposes a progressive gradient walk as the most efficient way to ensure that areas of good fitness are present in the sample.

2.1 Definition of a Random Walk

A random walk of length n is a sequence of points in an m-dimensional search space, obtained by starting at a certain point in the search space, $\vec{x}_0$, and generating the next point, $\vec{x}_1$, by randomly selecting a neighbour of $\vec{x}_0$. In general, every $\vec{x}_{i+1}$ is obtained by randomly selecting a neighbour of $\vec{x}_i$. Thus, a random walk X_n is a sequence of points $(\vec{x}_0, \vec{x}_1, \ldots, \vec{x}_n)$, where every $\vec{x}_{i+1}$ is derived from $\vec{x}_i$ using a neighbourhood function, $\vec{x}_{i+1} \leftarrow N(\vec{x}_i)$.

Random walks are used in multiple scientific fields such as physics, biology, and economics [23, 27]. In the context of FLA, random walks provide an alternative to random sampling from a given probability distribution [20, 34]. As opposed to a random sample, where individual points in the sample are spatially uncorrelated, random walks generate spatially correlated samples. It is precisely the spatial correlation between the individual steps of the sample that can be exploited to obtain descriptive information such as the degree of ruggedness, neutrality, or gradients present in the search space [18].

2.2 Adaptive Walks

The unbiased nature of random walks ensures that each point in the search space has an approximately equal probability of being selected. However, if the purpose of the analysis is to quantify the presence and extent of local minima, such a randomised approach may not prove very useful. Indeed, if a random sample does not contain any points associated with good fitness, no conclusions about the landscape features of the areas associated with good fitness can be made. Thus, fitness landscape characteristics are often derived from an *adaptive* rather than a random walk [15, 24]. Adaptive walks were originally defined for binary problems. To perform an adaptive walk, a neighbour $\vec{x}_k$ of $\vec{x}_i$ is randomly chosen. The neighbour $\vec{x}_k$ is accepted as the next step of the walk, $\vec{x}_{i+1}$, if and only if the fitness of $\vec{x}_k$ is better than the fitness of $\vec{x}_i$ [15]. In the context of genetic algorithms, a neighbour of $\vec{x}_i$ can be generated by applying a random mutation to $\vec{x}_i$, i.e. randomly flipping one or more bits of $\vec{x}_i$. This approach is equivalent to stochastic hill climbing in a binary space. Kauffman and Levin [15] estimated the ruggedness of the landscapes based on the average length of the adaptive walk. A shorter average length would indicate a rugged landscape, whereas a longer average length would be indicative of larger areas of consistently decreasing fitness.

2.3 Random and Adaptive Walks in Continuous Search Spaces

Discrete space sampling can be performed exhaustively, as each point $\vec{x}_i$ will at all times have a finite number of neighbours. This is not the case in continuous spaces, where every point $\vec{x}_i$ has an infinite number of neighbours in every dimension. Therefore, both random walks and adaptive walks can only be used in continuous spaces if neighbour selection is defined as a finite process.

The neighbourhood of a point $\vec{x}_i$ in a continuous m-dimensional space can be defined as all points within a certain distance from $\vec{x}_i$. Malan and Engelbrecht [20] proposed the following hypercube definition of the continuous neighbourhood of $\vec{x}_i$:

$$\vec{x}_k \in N(\vec{x}_i) \iff |x_{kj} - x_{ij}| \leq s, \forall j \in \{1, \ldots, m\} \quad (1)$$

where $\vec{x}_k$ is a neighbour of $\vec{x}_i$ if and only if for every dimension j the absolute difference between x_{kj} and x_{ij} does not exceed some s.

Using Equation 1, a single step of a simple random walk can be defined as randomly generating an m-dimensional step vector $\vec{y}_k$, such that $y_{kj} \in [-s, s]\ \forall j \in \{1, \ldots, m\}$, and adding $\vec{y}_k$ to $\vec{x}_i$ to generate $\vec{x}_{i+1}$:

$$\vec{x}_{i+1} = \vec{x}_i + \vec{y}_k \quad (2)$$

A simple random walk is isotropic, i.e. not biased towards a particular direction, since the direction of each step is randomised. An anisotropic, or directionally biased variant of a random walk was proposed by Malan and Engelbrecht [20], called a progressive random walk. The progressive random walk assigns a randomly chosen direction bias to each walk in order to improve overall search space coverage. Direction bias is represented by an m-dimensional randomly generated bit mask $\vec{b}$. A single step of a progressive random walk can be defined as randomly generating an m-dimensional step vector $\vec{y}_k$, such that $y_{kj} \in [0, s]\ \forall j \in \{1, \ldots, m\}$, and setting the sign of each y_{kj} according to the corresponding b_j:

$$y_{kj} = \begin{cases} -y_{kj}, & \text{if } b_j = 0. \\ y_{kj}, & \text{otherwise.} \end{cases}$$

Equation 2 can then be used to generate $\vec{x}_{i+1}$. Thus, the magnitude of the step is randomised per dimension, but the overall direction of movement remains persistent. For a more detailed discussion of the algorithm, refer to [20].

Both the simple random walk and the progressive random walk do not take the fitness of the neighbours into account when generating the next step $\vec{x}_{i+1}$. Smith *et al* [26] have shown that when

the distribution of fitness values across the search space is highly skewed towards poor fitness, random sampling may produce an inadequate sample that does not capture enough points of high fitness. Thus, an adaptive walk for continuous search spaces is necessary.

Adaptive walks in discrete spaces rely on random mutations of the individual. The same approach can be employed in continuous spaces, thus emulating stochastic hill climbing. Mutations can be performed by adding random noise in one or more dimensions. If the mutated position has a higher fitness than the current position, the mutated position will be added to the walk. However, if the search space is high-dimensional and skewed towards poor fitness areas, such random mutations are likely to not produce neighbours of higher fitness. Thus, stochastic hill climbing in continuous search spaces will be computationally expensive, and may produce very short walks that neither adequately cover the search space, nor find areas of high fitness.

Particle swarm optimisation (PSO) has also been proposed in the past as a sampling method. Each particle in the swarm represents a candidate solution, and the next step of the walk can be defined in terms of the next step of the global best particle in the swarm [18]. There are two problems with this approach: Firstly, PSO sampling is algorithm-specific, and the trajectory will be highly sensitive to algorithm parameters. Secondly, PSO has been shown to exhibit divergent behaviour on NN training [32], which yields this algorithm a suboptimal choice for NN error landscape sampling.

A number of attempts have been made to study the error surface of NNs from the perspective of the gradient descent trajectory [9, 10, 16]. Gradient descent uses the numerical gradient of the error function, thus the fitness is likely to increase per step, provided there is an incline. Similarly to PSO, analysing the trajectory of gradient descent is algorithm-specific. Steep gradients combined with the learning rate parameter may induce large steps through the search space, while weak gradients may produce small steps. Thus, the step sizes are bound to be inconsistent, providing an unrealistic view of the search space. Additionally, the lack of stochasticity makes gradient descent unlikely to investigate the areas of poor and average fitness.

This study proposes to combine the gradient information available in case of NNs with the stochasticity of the progressive random walk. The proposed algorithm is described in the next section.

2.4 Progressive Gradient Walk

Gradient information, when available, is clearly the most direct way of performing hill-climbing in a continuous search space. In addition to being a reliable source of direction, the gradient is also more efficient to compute than choosing the best individual in a population. Population-based approaches such as PSO require each individual to be evaluated separately, whereas the gradient is computed once per step. Computational efficiency is an important concern for NNs, since NN search spaces are inherently high-dimensional.

To alleviate gradient descent specificity of the proposed adaptive walk, and to study the error landscape as a whole rather than an algorithm trajectory, the following approach is proposed:

(1) Gradient vector $\vec{g_i}$ is calculated for point $\vec{x_i}$.

(2) A binary direction mask $\vec{b_i}$ is extracted from $\vec{g_i}$:

$$b_{ij} = \begin{cases} 0, & \text{if } g_{ij} < 0. \\ 1, & \text{otherwise.} \end{cases}$$

(3) Progressive random walk algorithm is used to generate $\vec{x}_{i+1}$.

The progressive random walk algorithm requires two parameters to be set: maximum dimension-wise step size, s, and the boundaries of the search space. The progressive gradient walk requires the same two parameters. Behaviour of the progressive gradient walk with no search space boundaries is also investigated in this paper.

The next section provides an empirical analysis of the proposed gradient walk on a typical NN error surface.

3 EMPIRICAL ANALYSIS

The aim of this study is to illustrate that random sampling fails to capture high fitness solutions, and that the proposed progressive gradient walk generates more representative samples than the random walks. The rest of this section details the experiments conducted to illustrate these points. Section 3.1 outlines the dataset used. Section 3.2 describes the NN architecture employed. Section 3.3 provides a discussion of the obtained results.

3.1 Dataset

The XOR problem was chosen for the purpose of this study as the simplest classification problem requiring a non-linear solution. The entire dataset can be seen in Table 1. Despite being a seemingly trivial problem, the XOR is not linearly separable, and generates a complex error landscape that is still not fully understood [11, 28].

Table 1: The XOR Problem Dataset

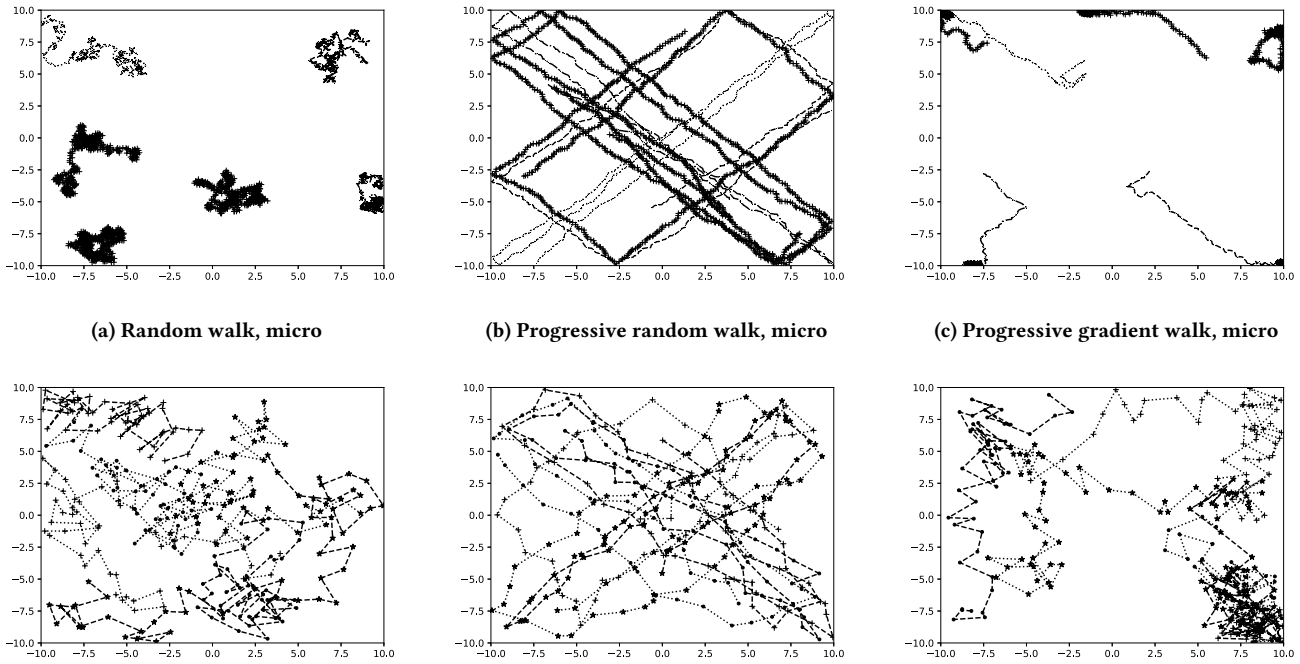
Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	0

3.2 Neural Network Architecture

Given the classic XOR problem, a corresponding fully-connected feed-forward NN architecture was chosen. The NN comprised of two input units, two hidden units, and one output unit [28]. Bias weights were associated with the hidden and the output units. The total number of weights was equal to 9. The sigmoid activation function was employed in the hidden and the output units. The mean squared error was used as an error metric to calculate the gradients and the fitness of any given point in the search space.

3.3 Experiments

Random sampling is typically performed within some predefined bounds. For the purpose of this study, the search space bounds were set to $[-10, 10]$. This range was chosen as the range likely to contain high fitness solutions [3]. Since the granularity of the walk, i.e. the average step size, has a bearing on the resulting FLA metrics [19], two granularity settings were used throughout the



(a) Random walk, micro (b) Progressive random walk, micro (c) Progressive gradient walk, micro

(d) Random walk, macro (e) Progressive random walk, macro (f) Progressive gradient walk, macro

Figure 1: Plots of the positions of paired dimensions of sample walks. Micro walks were performed over 500 steps. Macro walks were performed over 50 steps.

experiments: micro, where the maximum step size was set to 1% of the search space, and macro, where the maximum step size was set to 10% of the search space.

To illustrate the basic movement dynamics of the various walks, a sample of points obtained by a random walk, a progressive random walk, and a progressive gradient walk under micro and macro settings are shown in Figure 1. Each walk was performed in 9 dimensions corresponding to the NN weights. The first 6 dimensions of 2 independent walks are plotted in pairs along the axes. Thus, each axis corresponds to a weight of the NN, and the depicted subset of a walk illustrates how the walk progressed through a subset of 2 of the 9 dimensions. Micro walks performed 500 steps, and macro walks performed 50 steps. It is evident from Figure 1 that the progressive gradient walk is biased compared to the random walks, but does perform a reasonable amount of exploration. Smaller step size leads to more consistent trajectories (see Figure 1c), indicating that the surface is locally smooth. Previous theoretical studies have indicated that the NN error landscapes are comprised of plateaus and narrow ravines [10, 16]. The progressive gradient walk may be exploring these consistent structures.

To estimate the search space coverage of the three walks, 10,000 points were generated per macro walk, and 100,000 points were generated per micro walk. A total of 100 walks were performed under both the micro and macro setting. Micro walks performed 1000 steps each, and macro walks performed 100 steps each. The values across all dimensions were plotted in histograms of 100 equally sized bins each. The resulting histograms are shown in Figure 2. Mean and standard deviation values of the samples are also displayed above each histogram. While the random and the progressive random walks covered the search space near-uniformly (mean close to zero), the gradient walk leaned strongly towards the borders of the search space, especially in the micro case. Even though the walks were not allowed to leave the search space, it appears that the gradient direction often pointed outwards, thus causing the gradient walks to cluster around the boundaries. This observation is in line with the previous studies, proposing that the NN error landscapes have a "starfish" or "sombrero" structure, with ravines of lower error leading outwards [7, 16]. A question thus needs to be answered: should NN error landscapes be studied within predefined boundaries? Previous studies have argued that the boundaries are necessary, since random sampling cannot be performed in unbounded space [3, 20]. However, if progressive gradient sampling is used instead of random sampling, the gradient information should lead the walks to "interesting" areas of the landscape, rather than causing meaningless wandering. Unbounded progressive gradient sampling has been performed, and the resulting walk samples are shown in Figures 3a and 3d. In both micro and macro settings, the gradient walks tended to move away from the origin, once again aligning with the "starfish" structure. The search space coverage histograms for the unbounded gradient walks are shown in Figures 3b and 3e. Unbounded walks exhibited less clustering than bounded walks, and covered the search space

better. Spikes associated with particular ranges may be explained by the presence of local minima or saddle points that could have trapped the gradient walk.

Progressive gradient walks are suggested as a method of search space sampling that is more likely to find areas of good fitness than the random walks. To estimate the fitness coverage, the fitness frequency distribution of the sample points obtained by each of the walks under micro and macro setting were plotted in histograms of 100 equally-sized bins. The resulting histograms are shown in Figure 4. The means and standard deviations of each distribution are shown above the histograms. It is evident from Figure 4 that both the random and the progressive random walk failed to discover areas of good fitness. For both random sampling techniques, the average MSE was around 0.45, with a very sharp peak on the average value, and a heavy tail on the left, corresponding to areas of above average fitness. The lowest error sampled by the random walks hovered around 0.2. Thus, the random walks have sampled mostly average (random guess) fitness areas, and the areas of optimal fitness (near zero) were almost not sampled at all. Thus, whatever conclusions about the fitness landscapes are made based on these random walks, the conclusions would only be applicable to the areas of random guess fitness. These areas are the least interesting areas from an optimisation algorithm perspective, and an optimisation algorithm is expected to spend the least amount of time in those areas. Thus, the usefulness of studying low fitness areas is highly doubtful.

A progressive gradient walk, on the other hand, has successfully captured error values around zero (optimal fitness). Figures 4c and 4f indicate that the mean error of the gradient walks was below the lowest error of the random walks. Interestingly, both the micro and the macro gradient walks exhibited peaks around specific error values. This can be an indication of the presence of local minima or saddle points at those fitness values. Macro progressive gradient walk exhibited a good spread of fitness values between 0.0 and 0.5, indicating that the macro samples may have captured the information relevant to a potential training algorithm. Fitness frequency histograms have also been plotted for the unbounded gradient walks, shown in Figures 3c and 3f. Unbounded gradient walks have captured a similar distribution of fitness values as the bounded gradient walks, exhibiting similar peaks. The peaks may be indicative of the modality of the error landscape. Future research will investigate this correspondence.

Spread of fitness values was also calculated in terms of classification accuracy. Since the XOR problem has only four data points, the set of possible classification accuracy values is discrete, and is comprised of the following values: $\{0.0, 0.25, 0.5, 0.75, 1\}$, where 0.0 indicates incorrect output for each pattern, and 1.0 indicates correct output for all patterns. Frequency histogram for the various walks under micro and macro setting is shown in Figure 5. It is evident from Figure 5 that the random walks failed to capture areas of 100% accuracy, and sampled mostly average, or random guess accuracy points instead. Gradient walks, on the other hand, sampled mostly above average accuracy and 100% accuracy. Macro setting yielded a better coverage of average accuracy by the gradient walks. Once again, the gradient walks captured the areas of the landscape that are of a higher interest to a potential optimisation algorithm.

Fitness Landscape Metrics. Since progressive gradient walks capture a different distribution of fitness values compared to the random walks, the FLA metrics are expected to yield different results when calculated over the gradient walks. To test this hypothesis, three metrics were used, originally proposed as metrics calculated over the random walks. The metrics are:

(1) **First Entropic Measure of Ruggedness (*FEM*)**: Malan and Engelbrecht [19] proposed two ruggedness measures based on Vassilev's [33] first entropic measure (*FEM*). These measures quantify the change in fitness values based on the entropy of the random walk. The value of *FEM* is continuous and ranges between 0 and 1, where 0 indicates a perfectly smooth landscape, and 1 indicates maximal ruggedness.
(2) **Neutrality Measures** $M1$ **and** $M2$: Two random walk-based neutrality metrics were proposed in [31], $M1$ and $M2$. $M1$ measures the proportion of neutral 3-point structures in a walk, and $M2$ measures the relative length of the largest sequence of neutral steps in the walk. A step is classified as neutral if the fitness does not change by more than a certain specified threshold value. Both $M1$ and $M2$ range between 0 and 1, where 0 indicates a landscape with no neutral regions, and 1 indicates a completely flat landscape.

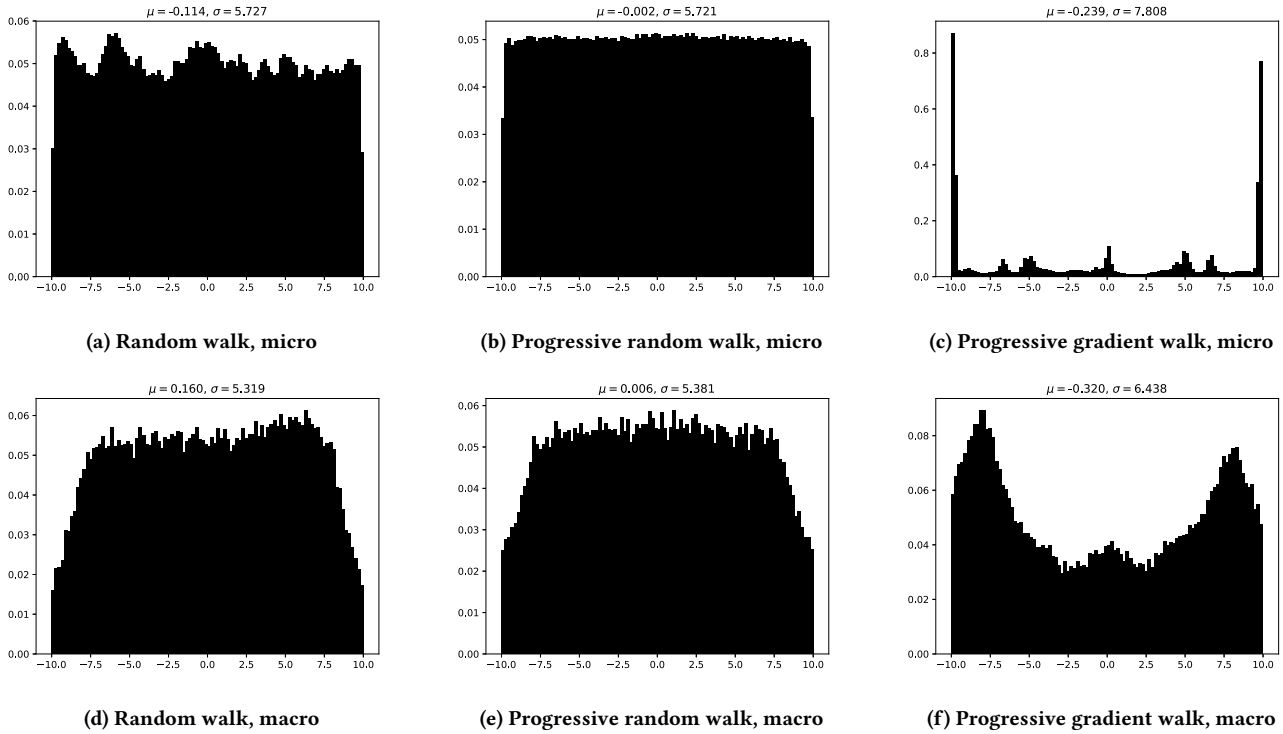
The resulting *FEM*, $M1$, and $M2$ values calculated over the various walks are shown in Table 2. Table 2 shows average values obtained over 30 independent runs, together with the corresponding standard deviations shown in parenthesis. Each run comprised of 100 walks. Each micro walk performed 1000 steps, and each macro walk performed 100 steps.

Table 2 shows that both bounded and unbounded progressive gradient walks exhibited a higher disparity between the micro and the macro *FEM* values than the random walks. Indeed, the random sampling algorithms have covered more or less the same areas of average fitness, while the gradient walks focused on areas of higher fitness. The maximal size of the step had an influence on the resulting *FEM*, since the error landscape is smooth when observed locally, and exhibits ruggedness when observed at a larger scale. Perhaps *FEM* values can be used to suggest an appropriate step size scaling for NN optimisation algorithms.

Table 2 shows that the values of $M1$ were more or less the same for all the bounded walks, and only the unbounded gradient walk captured areas of increased neutrality. Indeed, if the gradient walk is allowed to leave the bounded search space, it is likely to fall into one of the "ravines" that the NN landscapes are known to contain. Thus, the unbounded gradient walk captured a property of the NN landscape that the other walks did not.

Values of $M2$ shown in Table 2 indicate that none of the walks have experienced a long stretch of unchanging fitness values. Thus, the progressive gradient walk did not simply converge on a single point in the search space and sample it indefinitely.

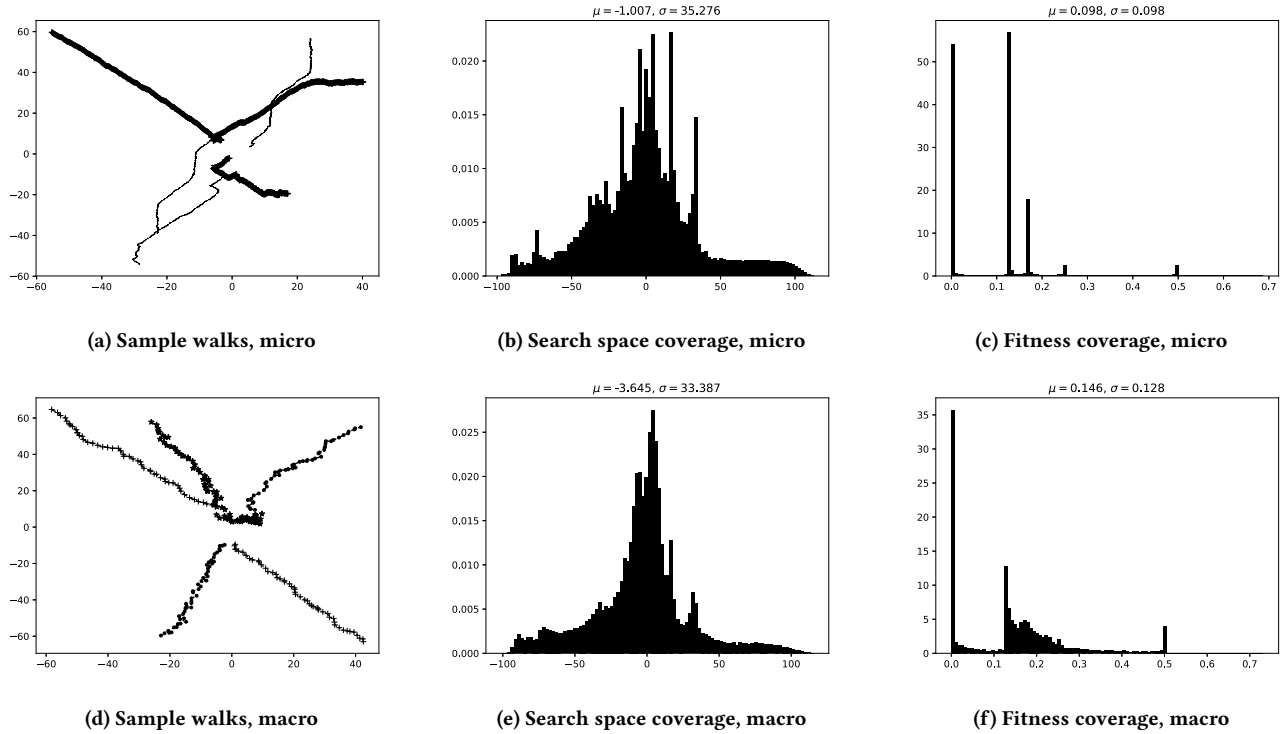
These results are presented here simply to indicate that the different walks do indeed capture different properties of the NN error landscapes, and analysing the gradient walks rather than the random walks may highlight more interesting and important features of the landscape than that of the unbiased random walks. Development of searchability and modality metrics based on the progressive gradient walk is left for future research.



(a) Random walk, micro **(b) Progressive random walk, micro** **(c) Progressive gradient walk, micro**

(d) Random walk, macro **(e) Progressive random walk, macro** **(f) Progressive gradient walk, macro**

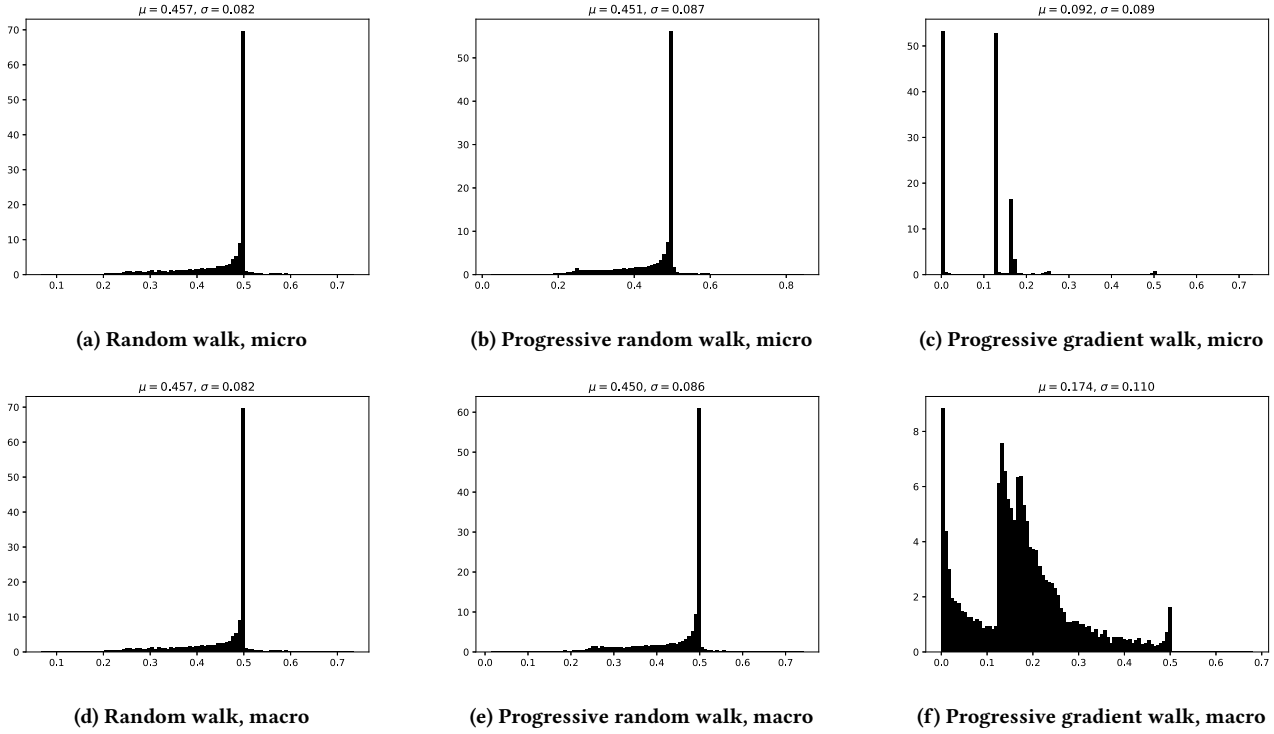
Figure 2: Frequency diagrams illustrating search space coverage by the various walks.



(a) Sample walks, micro **(b) Search space coverage, micro** **(c) Fitness coverage, micro**

(d) Sample walks, macro **(e) Search space coverage, macro** **(f) Fitness coverage, macro**

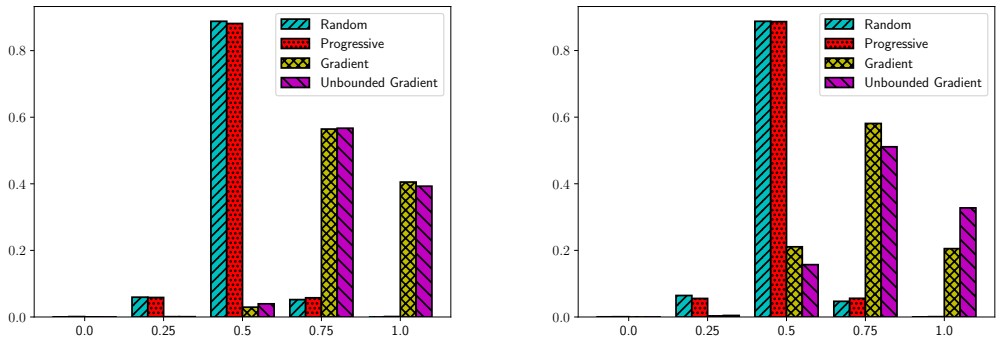
Figure 3: Unbounded progressive gradient walk



(a) Random walk, micro **(b) Progressive random walk, micro** **(c) Progressive gradient walk, micro**

(d) Random walk, macro **(e) Progressive random walk, macro** **(f) Progressive gradient walk, macro**

Figure 4: Frequency diagrams of the fitness (MSE) associated with the samples obtained by the various walks.



(a) Micro setting **(b) Macro setting**

Figure 5: Frequency diagrams of the classification accuracy associated with the samples obtained by the various walks.

4 CONCLUSIONS

This paper proposed a progressive gradient walk as an adaptive sampling mechanism for the analysis of neural network fitness landscapes. The gradient walk is more computationally efficient than a population-based adaptive walk, and has better guarantees of finding areas of high fitness. The gradient information is used to calculate the direction of the next step, but the magnitude of the step is randomised per dimension within the given bounds, thus adding stochasticity and preventing convergence.

Both bounded and unbounded progressive gradient walks were compared to the random and progressive random walks in terms of search space coverage and fitness coverage. It was shown that, even though random walks provide wider search space coverage, they fail to capture areas of high fitness. The gradient walk, on the other hand, is strongly biased towards the areas of high fitness, while also covering some of the poor fitness areas. Thus, the gradient walk is more representative of the search space in the context of applicability to function optimisation. In addition, the unbounded

Table 2: FLA metrics obtained over various walks

	Random	Progressive	Gradient	Unbounded
FEM	0.41178	0.30623	0.20239	0.20234
(micro)	(0.02238)	(0.00989)	(0.00263)	0.00263
FEM	0.47964	0.46006	0.65177	0.56348
(macro)	(0.01450)	(0.00611)	(0.00707)	(0.01880)
$M1$	0.01897	0.02956	0.01003	0.36281
(micro)	(0.01192)	(0.00502)	(0.01287)	(0.03622)
$M1$	0.00310	0.00238	0.00967	0.12001
(macro)	(0.00194)	(0.00084)	(0.00964)	(0.02479)
$M2$	0.00635	0.01095	0.00003	0.00142
(micro)	(0.00398)	(0.00162)	(0.00003)	(0.00026)
$M2$	0.00252	0.00188	0.00000	0.01999
(macro)	(0.00149)	(0.00064)	(0.00002)	(0.01042)

progressive gradient walk seems to provide a truer picture of the error landscape than the bounded gradient walk.

Finally, a selection of FLA metrics were calculated over the random and the gradient walks. While the obtained FLA metrics did not disagree with one another, the FLA metrics obtained from the gradient walks seemed to capture the specific known characteristics of NN error landscapes with better precision.

Future work will involve a scalability study of the proposed gradient walk. Extensive experiments will be conducted on more complex NN problems and architectures. The presence of high fitness points in the gradient samples also allows for development of searchability, modality, and dispersion metrics for NN error landscapes.

ACKNOWLEDGMENTS

This work is based on the research supported by the National Research Foundation (NRF) of South Africa (Grant Number 46712). The opinions, findings and conclusions or recommendations expressed in this article is that of the author(s) alone, and not that of the NRF. The NRF accepts no liability whatsoever in this regard.

REFERENCES

[1] C. M. Bishop. 1995. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, UK.

[2] Anna Bosman, Andries Engelbrecht, and Mardé Helbig. 2017. Fitness Landscape Analysis of Weight-Elimination Neural Networks. *Neural Processing Letters* (2017), 1–21.

[3] A. S. Bosman, A. P. Engelbrecht, and M. Helbig. 2016. Search Space Boundaries in Neural Network Error Landscape Analysis. In *Proceedings of the IEEE Symposium Series on Computational Intelligence*. IEEE, Athens, Greece, 1–8.

[4] Anna Choromanska, Yann LeCun, and Gérard Ben Arous. 2015. Open Problem: The landscape of the loss surfaces of multilayer networks. In *Proceedings of The 28th Conference on Learning Theory*, Vol. 40. JMLR, Paris, France, 1756–1760.

[5] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. 2014. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems*. Curran Associates, Montréal, Canada, 2933–2941.

[6] Li Deng, Geoffrey Hinton, and Brian Kingsbury. 2013. New types of deep neural network learning for speech recognition and related applications: An overview. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, Vancourver, Canada, 8599–8603.

[7] John Denker, Daniel Schwartz, Ben Wittner, Sara Solla, Richard Howard, Lawrence Jackel, and John Hopfield. 1987. Large automatic learning, rule extraction, and generalization. *Complex systems* 1, 5 (1987), 877–922.

[8] G. Dreyfus. 2005. *Neural networks: methodology and applications.* Springer, Berlin, Germany.

[9] Marcus Gallagher and Tom Downs. 2003. Visualization of learning in multilayer perceptron networks using principal component analysis. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 33, 1 (2003), 28–34.

[10] M. R. Gallagher. 2000. *Multi-layer Perceptron Error Surfaces: Visualization, Structure and Modelling.* Ph.D. Dissertation. University of Queensland, St Lucia 4072, Australia.

[11] Leonard GC Hamey. 1998. XOR has no local minima: A case study in neural network error surface analysis. *Neural Networks* 11, 4 (1998), 669–681.

[12] Guang-Bin Huang, Lei Chen, and Chee Kheong Siew. 2006. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Transactions on Neural Networks* 17, 4 (2006), 879–892.

[13] Jérémie Humeau, Arnaud Liefooghe, E-G Talbi, and Sébastien Verel. 2013. ParadisEO-MO: From fitness landscape analysis to efficient local search algorithms. *Journal of Heuristics* 19, 6 (2013), 881–915.

[14] Terry Jones. 1995. *Evolutionary algorithms, fitness landscapes and search.* Ph.D. Dissertation. The University of New Mexico.

[15] Stuart Kauffman and Simon Levin. 1987. Towards a general theory of adaptive walks on rugged landscapes. *Journal of theoretical Biology* 128, 1 (1987), 11–45.

[16] Mirosław Kordos and Wlodzisław Duch. 2004. A survey of factors influencing MLP error surface. *Control and Cybernetics* 33, 4 (2004), 611–631.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. Curran Associates, Lake Tahoe, USA, 1097–1105.

[18] Katherine Mary Malan. 2014. *Characterising continuous optimisation problems for particle swarm optimisation performance prediction.* Ph.D. Dissertation. University of Pretoria.

[19] Katherine M Malan and Andries P Engelbrecht. 2009. Quantifying ruggedness of continuous landscapes using entropy. In *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, Trondheim, Norway, 1440–1447.

[20] Katherine M Malan and Andries P Engelbrecht. 2014. A progressive random walk algorithm for sampling continuous fitness landscapes. In *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, Beijing, China, 2507–2514.

[21] Peter Merz and Bernd Freisleben. 2000. Fitness landscape analysis and memetic algorithms for the quadratic assignment problem. *IEEE Transactions on Evolutionary Computation* 4, 4 (2000), 337–352.

[22] Mario A Muñoz, Yuan Sun, Michael Kirley, and Saman K Halgamuge. 2015. Algorithm selection for black-box continuous optimization problems: A survey on methods and challenges. *Information Sciences* 317 (2015), 224–245.

[23] Karl Pearson. 1905. The problem of the random walk. *Nature* 72, 1867 (1905), 342.

[24] Erik Pitzer and Michael Affenzeller. 2012. A comprehensive survey on fitness landscape analysis. In *Recent Advances in Intelligent Engineering Systems*. Springer, 161–191.

[25] Anna Rakitianskaia, Eduan Bekker, Katherine Malan, and Andries Engelbrecht. 2016. Analysis of Error Landscapes in Multi-layered Neural Networks for Classification. In *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, Vancouver, Canada, 5270–5277.

[26] Tom Smith, Phil Husbands, and Michael O'Shea. 2001. Not measuring evolvability: Initial investigation of an evolutionary robotics search space. In *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 1. IEEE, Seoul, South Korea, 9–16.

[27] Frank Spitzer. 2013. *Principles of random walk.* Vol. 34. Springer Science & Business Media.

[28] Ida G Sprinkhuizen-Kuyper and Egbert JW Boers. 1999. The local minima of the error surface of the 2-2-1 XOR network. *Annals of Mathematics and Artificial Intelligence* 25, 1-2 (1999), 107.

[29] Yuan Sun, Saman K Halgamuge, Michael Kirley, and Mario A Muñoz. 2014. On the selection of fitness landscape analysis metrics for continuous optimization problems. In *International Conference on Information and Automation for Sustainability*. IEEE, Colombo, Sri Lanka, 1–6.

[30] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. Montréal, Canada, 3104–3112.

[31] Willem Abraham van Aardt, Anna Sergeevna Bosman, and Katherine Mary Malan. 2017. Characterising neutrality in neural network error landscapes. In *IEEE Congress on Evolutionary Computation*. IEEE, San Sebastian, Spain, 1374–1381.

[32] A. B. Van Wyk and A. P. Engelbrecht. 2010. Overfitting by PSO trained feedforward neural networks. In *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, Barcelona, Spain, 1–8.

[33] Vesselin K Vassilev, Terence C Fogarty, and Julian F Miller. 2003. Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application. In *Advances in evolutionary computing*. Springer, 3–44.

[34] Edward Weinberger. 1990. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological cybernetics* 63, 5 (1990), 325–336.