Exploiting Fractalness of Error Surfaces: New Methods for Neural Network Learning

Andrew B. Kahng

UCLA Computer Science Dept., Los Angeles, CA 90024-1596, (213)206-7073

Abstract

Learning in neural networks can be formulated as global optimization of a multimodal error function over a high-dimensional space of connection weights; this problem is intractable. We develop a general scaling model which describes the error surface as high-dimensional fractional Brownian motion *(fBm), i.e., as a class of random fractals. Using techniques first proposed by Sorkin [15], the parameter of fBm can be extracted by spectral analysis of the error profile over a* random walk *in weight space. Scaling structure within the error surface has important implications for stochastic optimizations such as Boltzmann learning. We review experimental data which confirm fractalness of error surfaces for a wide range of problems and connection topologies, as well as implications of these results.*

1 Introduction

Learning in neural networks can be formulated as global optimization of a multimodal error function that is defined over the high-dimensional space of connection weights. This global optimization is both theoretically intractable [11] [16] and difficult in practice. Traditional learning heuristics, e.g., back-propagation [13] or Boltzmann learning [4], are largely based on gradient methods or stochastic hill-climbing, reflecting traditional global optimization approaches (see, e.g., [3] for a survey). While these methods have shown promise on smaller problem instances, optimizing larger connectionist architectures raises several difficulties which motivate the present work:

- **No performance bounds.** Local optima can be arbitrarily far from global optima. Indeed, for certain "hard" combinatorial optimization instances, the expected result of either the steepest-descent or simulated annealing approaches will be no better than a random solution [2].
- **No estimates of minimal network topology.** We cannot tell *a priori* whether an n-node topology is capable of a prescribed discrimination task, nor do we know whether our training algorithm will in fact allow a network to achieve its capability. Thus, choice of network architecture is largely an *ad hoc* decision. For large problems, we can only implicitly justify trained solutions as the result of huge amounts of computation, or by arguing that the training methods had previously been effective on smaller problems.
- **Instability.** Training heuristics are unpredictable, and sensitive to initial random seeds [8]. Thus, we have little idea of the incremental benefit of spending additional CPU time on network training. Ideally, methods will be stable and exhibit a smooth tradeoff between performance and CPU cost.
- **Poor scaling.** Theoretical results on learning methods are essentially statements of convergence to local minima in the error surface. This is of little practical impact since local search heuristics exhibit an "error catastrophe" [6] as problems grow large. We require methods where solution quality scales with problem size.
- **Finite time requirements.** Finally, several learning algorithms (particularly Boltzmann-style algorithms) are "provably good", but only in the limit of infinite time. In practice, we require a theory which explicitly addresses the tradeoff between

The remainder of this paper describes a model of learning as "search" in the error surface, then qualitatively reviews a new model for error surfaces as a class of high-dimensional fractional Brownian motions (fBm). We conclude with experimental confirmations of this theory, and their implications.

2 Optimization as Search

In neural network learning, the usual strategy for searching for an optimum weight assignment is to generate a slight perturbation of the current solution, then decide whether this change should be accepted. Generation of the perturbation yields structure: some points in the solution space are directly *reachable* from others, and the induced reachability graph is called a *neighborhood structure* [9]. The quality of solutions defines a cost surface over the neighborhood structure, and optimization is search for a global optimum in this cost surface.

Existing classes of heuristics correspond to various rules for generating a new candidate solution and deciding whether to adopt the solution. For this discussion, we adopt the general stochastic hill-climbing template in Figure 1 known as *simulated annealing* with temperature parameter T [7]. In Figure 1, $f(x)$ denotes the value of the cost function for the solution x.

0-7803-0593-0/92 $3.00 1992 IEEE

Generate candidate solution x'
If $f(x') < f(x)$ then $x = x'$
/*always accept improvement */
Else $x = x'$ with probability $e^{-(f(x')-f(x))/T}$
/* Boltzmann acceptance */
otherwise $x = x$
/* leave x unchanged */

Figure 1: The Boltzmann Learning (BL) Inner Loop.

Basic optimization strategies, e.g., greedy and random search, are limiting cases of annealing. Random search can be approximated by taking a *random walk* in the neighborhood structure, i.e., iteratively moving to a randomly chosen neighbor of the current solution, then returning the best cost value found on the walk. This is equivalent to BL with $T = \infty$. At the other extreme, BL with $T = 0$ is equivalent to greedy search. Practical cost surfaces seem to be best searched using strategies "intermediate" between randomness and greed.

To devise new and effective hill-climbing heuristics, we require a model for the smoothness or correlatedness of the cost surface. Certainly, for pathological surfaces with a single "gopher hole" as global minimum will require exhaustive search, and BL will actually be less efficient than enumeration or random search. On the other hand, BL will be no more useful than greed for "simple" surfaces. As noted in, e.g., [1] and [17], it seems that hill-climbing would be a win, as it has been in practical experience, when the error surface is "smooth, but not too smooth". Slices through small neural network error surfaces have been exhaustively plotted to gain insight into the structure of actual problems, but no general structural models have resulted.

3 Scaling of the Error Surface

We propose a general scaling model for neural network error surfaces; we note that Sorkin [15] has investigated similar ideas for combinatorial optimization problems in VLSI layout. A useful definition (see [5] [15] [18] for necessary background in fractal sets and random processes) is:

Definition: A *fractional Brownian motion* (fBm) with parameter $H \geq 0$ is a sequence of random variables $X(t)$ such that the random variable $X(t_2) - X(t_1)$ is Gaussian with mean zero and variance

$$E[|X(t_2) - X(t_1)|^2] \propto |t_2 - t_1|^{2H} \qquad (1)$$

Because any sample of a fBm is statistically similar to the original fBm [18], fractional Brownian motions are called *statistical fractals*. Equation (1) implies that the scaling of fBm is power-law, e.g., at time t a "drunkard's walk" on the one-dimensional lattice has expected divergence of $t^{1/2}$ from the origin.

The *cost profile* over a random walk to be the sequence of cost values encountered as we iteratively move to random adjacent solutions. A fundamental observation [15] is the following:

Fact 1: The cost profile over a random walk in a high-dimension fBm error surface will itself be a fBm in one dimension. □

We have found strong evidence for the model of neural network error surfaces as high-dimensional fBm. Specifically, the parameter of fBm can be extracted by spectral analysis of the error profile as we take a random walk in weight space. A natural algorithmic approach to quantifying the *pseudo-fractalness* (i.e., smoothness) of the error surface is: (i) take a small (random-walk) sample of the weight space, (ii) treat the error values over this sample as a time series $X_0, \ldots, X_T$, (iii) take the discrete Fourier transform (DFT) $X(f, T)$

$$X(f,T) \;=\; 1/T \int_0^T X(t) e^{2\pi i f t} dt \;,$$

and (iv) plot the resulting spectral density, or periodogram $S_X(f)$

$$S_X(f) = \lim_{T \to \infty} T \cdot |X(f,T)|^2$$

on a log-log scale to confirm the power-law scaling via a straight-line fit. For example, the 1-D drunkard's walk would correspond to fBm with parameter $H = 1/2$, and the fitted slope of the log-log plot of the power spectrum is $-1/2$. Since the DFT algorithm is $O(n \log n)$, where n is the length of the random walk, and since the sampling of error values along the random walk may be performed in parallel, we have a very efficient computational methodology. The central idea is that the random walks will capture information that is useful for predicting the ease of finding the global optimum solution; we may therefore rapidly test a variety of candidate network architectures and training heuristics.

4 Experiments and Implications

Experiments have dealt with a wide range of benchmark classification tasks from the literature, as well as practical tasks involving various types of real-world sensor data (e.g., ground-penetrating radar, FLIR images). We have implemented standard DFT code [12] along with standard moving-window smoothing techniques using C in a Sun-4 environment. Our results confirm that error surfaces over a wide range of classification tasks and connection topologies are indeed describable as high-dimensional fBm. We emphasize that the power-law scaling relationship is very strong in *all* of the examples that we have considered. We now list some questions that are addressed by our research.

The basic underlying question concerns bounded-time search: What is the best search strategy for network learning when we are given only k time steps? Mitra et al. posed this problem in [10], but centered on imperfect convergence to stationarity in the general simulated annealing process. Under our model,

we may address the special problem of searching a fractal surface for a global minimum; our analysis begins with the case where only a single value of T in the annealing inner loop is allowed (such a search is called a *diffusion* at temperature T). One can show:

Fact 2: In searching a high-dimensional fBm surface for a global optimum, the optimal diffusion temperature T (i.e., the value of T which gives the best expected result at time k) is a function of the k and the fBm parameter H. □

Sorkin [15] has previously shown that the Boltzmann learning algorithm is provably good, i.e., will reach the optimum solution *in polynomial time* on a class of deterministically constructed fractals; this result can in fact be strengthened slightly. This lends hope that Boltzmann learning, which is no more efficient than exhaustive search on pathological error surfaces, is actually very effective for "real" error surfaces.

Our experiments have plotted average best-so-far energy (the minimum cost seen during the first k steps of the random walk) versus T, and obtained unimodal curves which confirm both the existence of an optimal T each given k, as well as the expected shift in T with varying time bound k. Studying the performance of bounded-time hill-climbing search in this way suggests the concept of a natural time scale, or relaxation time, for each optimization instance. As observed by Sibani and coworkers [14], the scaling of best-so-far energy can be used to predict ground state energy in the annealing, i.e., the cost of the global optimum solution. An example is shown by the sequence of Figures 2 - 4, which depict random-walk analysis of error surfaces in feedforward neural networks for a standard benchmark, the exclusive-OR problem. Each of these plots corresponds to a different network topology and shows the average best-so-far error obtained for the given architecture at different stages in the annealing process, as a function of fixed annealing temperature T (note that the vertical error axis is reparametrized in each figure).[1] Figures 2 - 4 show the result for networks with different numbers of hidden units applied to the XOR problem. The networks were connected in a full feedforward manner, i.e., each hidden node received activation from the two inputs as well as any other hidden units "behind" it. These networks vary from zero hidden units (a network which cannot carry out the task) to a network with five hidden units (many more than required). The figures show an increasingly pronounced "valley", i.e., a well-defined temperature at which the optimization is most likely to result in the optimal solution obtainable within the time bound k. The presence and depth of the valley, as well as the shape of the curve for each T, is an indication of whether the network can solve the problem within a given time bound. Due to the large number of trials at each of many data points, we confined our experiments to relatively small time scales; resolution capability should continue to increase as walks are lengthened.

[1] The initial values of a network's weights and thresholds were independent random variables uniform distributed between +1 and −1. At each step in the walk, a random perturbation were generated and accepted or rejected based on the Boltzmann criterion shown in Figure 1.

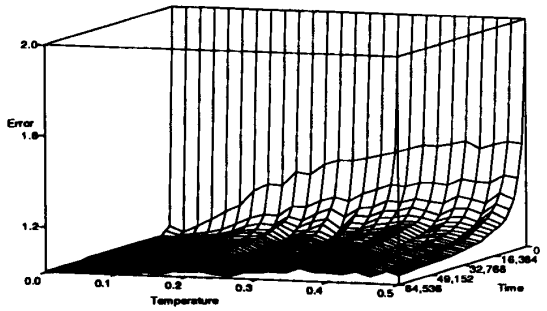

Figure 2: Random walks for XOR neural network with No Hidden Units.

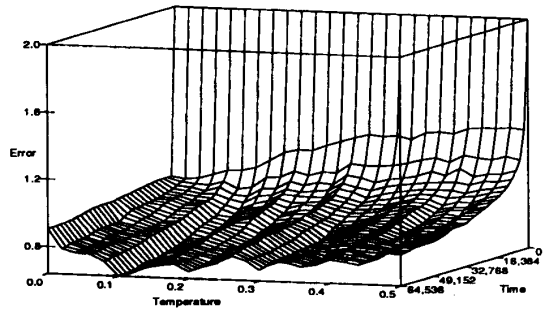

Figure 3: Random walks for XOR neural network with Two Hidden Units.

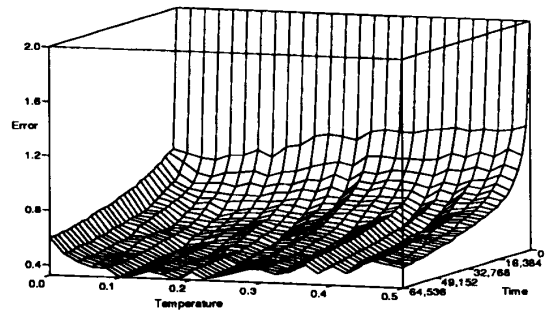

Figure 4: Random walks for XOR neural network with Five Hidden Units.

Other results address the issues of scale invariance and robustness of the random walk-based estimates. We can show that the same estimates for the utility of the architecture and the expected error of the trained network will be obtained regardless of the initial state in the random walk; this is a consequence of the statistical fractalness of the cost surface. A corollary is that the tightness of the estimate becomes

stronger both as the random walk length increases, and as the size of the solution space of weight vectors grows large compared to the available computational resources. This is reflected in the figures by the diminishing noise of the optimal-temperature valleys as the lengths of random walks increases. We are currently applying the "best-so-far error" scaling in Boltzmann learning to determine connectionist architectures that are *likely* to be trainable to a prescribed capability within a given time bound. In other words, we can examine the early progress of the Boltzmann learning algorithm to see if success is likely; if not, then a new connectionist topology can be tried.

In summary, the model and methodology we propose have resulted in new Boltzmann learning variants which we tune to individual problem instances by quickly estimating a parameter of smoothness in the cost surface. As described above, the parameter we use can be quickly calculated from Fourier spectra of random-walk cost profiles, and temperature schedules for annealing can be derived from this parameter. Under the scaling assumption for the cost surface, the quality of random walk-derived learning strategies improves with the amount of CPU time invested, and does not degrade as problem sizes grow large.

A number of open issues are raised by this work. We would like to extend the fractal model to attack difficult practical classification tasks such as object recognition. As suggested by the above results, we are using scaling properties of error surfaces induced by each connection topology to find smallest-possible "useful" networks for which simulated annealing optimization is *expected* to yield good recognition performance. Moreover, the scaling theory may allow rapid testing of whether the problem *representation* itself induces an error surface that is amenable to Boltzmann search. Finally, our theoretical models may extend to derive complete cooling schedules for simulated annealing, and also lead to new hill-climbing variants which specifically exploit the fractalness of practical neural network error surfaces.

References

[1] P. Baldi, "Linear Learning: Landscapes and Algorithms", *Proc. Neural Information Processing Systems*, 1988, pp. 65-72.

[2] T. N. Bui, S. Chaudhuri, M. Sipser and F. T. Leighton, "Graph Bisection Algorithms With Good Average-Case Behavior", *Combinatorica* 7(2) (1987), pp. 171-191.

[3] R. Hecht-Neilsen, *Neurocomputing*, Addison-Wesley, 1990.

[4] G. E. Hinton, "Deterministic Boltzmann Learning Performs Steepest Descent in Weight Space", *technical report* CRG-TR-89-1, Univ. of Toronto, 1989.

[5] A. B. Kahng and G. Robins, "On Structure and Randomness in Practical Optimization", *UCLA Computer Science Department Annual* (1990), pp. 23-38.

[6] S. Kauffman and S. Levin, "Toward a General Theory of Adaptive Walks on Rugged Landscapes", *J. Theoretical Biology* 128 (1987), pp. 11-45.

[7] S. Kirkpatrick, C. D. Gelatt, Jr. and M. P. Vecchi, "Optimization by Simulated Annealing", *Science* 220(4598) (1983), pp. 671-680.

[8] J. F. Kolen and J. B. Pollack, "Back Propagation is Sensitive to Initial Conditions", *technical report* TR 90-JK-BPSIC, Ohio State Univ., 1990.

[9] D. C. Llewellyn, C. Tovey and M. Trick, "Local Optimization on Graphs", *Disc. Applied Math.* 23 (1989), pp. 157-178.

[10] D. Mitra, F. Romeo and A. Sangiovanni-Vincentelli, "Convergence and Finite-Time Behavior of Simulated Annealing", *Adv. in Applied Probability* 18(1986), pp. 747-771.

[11] P. M. Pardalos and G. Schnitger, "Checking Local Optimality in Constrained Quadratic Programming is NP-Hard", *Operations Research Letters* 7(1) (1988), pp. 33-35.

[12] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing* , Cambridge, Cambridge University Press, 1988.

[13] D. E. Rumelhart, J. L. McClelland et al., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* , Cambridge, MIT Press, 1986.

[14] P. Sibani, J. M. Pedersen, K. H. Horrmann and P. Salamon, "Monte Carlo Dynamics of Optimization Problems: A Scaling Description", *draft* , May 1990.

[15] G. B. Sorkin, "Simulated Annealing on Fractals: Theoretical Analysis and Relevance for Combinatorial Optimization", *Proc. Sixth MIT Conf. on Adv. Research in VLSI* , March 1990, pp. 331-351.

[16] A. Torn and A. Zilinskas, *Global Optimization* , Lecture Notes in Computer Science 350, G. Goos and J. Hartmanis, eds., Springer-Verlag, 1987.

[17] D. Touretzky, "Analyzing the Energy Landscapes of Distributed Winner-Take-All Networks", *Advances in Neural Information Processing Systems I*, D. Touretzky, ed., Morgan Kaufmann, 1989, pp. 626-633.

[18] R. F. Voss, "Fractals in Nature: From Characterization to Simulation", in *The Science of Fractal Images* (H. O. Peitgen and D. Saupe, eds.), New York, Springer-Verlag, 1988.