FAST SIMULATED ANNEALING*

Harold Szu
Naval Research Laboratory, Code 5709, Washington, D.C. 20375-5000

Abstract

Simulated annealing is a stochastic strategy for searching the ground state. A fast simulated annealing (FSA) is a semi-local search and consists of occasional long jumps. The cooling schedule of FSA algorithm is inversely linear in time which is fast compared with the classical simulated annealing (CSA) which is strictly a local search and requires the cooling schedule to be inversely proportional to the logarithmic function of time. A general D dimensional Cauchy probability for generating the state is given. Proofs for both FSA and CSA are sketched. A double potential well is used to numerically illustrate both schemes.

INTRODUCTION

When the classical energy/cost function $C(\vec{x})$ has a single minimum, the conventional method can provide the unique ground state, and any method of gradient descent can approach the minimum. However, when $C(\vec{x})$ has multiple extrema, a nonconvex optimization technique that allows tunnelling and variable sampling and accepting hill-climbing for escaping from local minima is required. To illustrate the concept, we first consider a serial processing. If a ball is rolling over a hilly terrain inside a box, one must shake the box gently enough in the vertical direction of perturbations that the ball cannot climb up the global minimum valley and sufficiently vigorously along the horizontal direction of sampling to escape from local minimum valleys. Thus, a strategy of variable perturbations is needed. We secondly consider a concurrent parallel processing. A molten solid having random thermal energy must be gradually cooled down in order to reach the (globally minimum energy) crystalline state. Thus, a thermal random noise is useful when it is gradually quenched. These algorithms may be called simulated annealing, or Monte Carlo method when a constant noise temperature is assumed as first proposed by Metropolis et al.[1] for computer simulation of hard-disc phase transitions. Recently, Kirkpatrick et al.[2] in classical systems and Ceperley and Alder[3] in quantum systems have investigated a general and powerful computing technique for changing noise temperature and sampling grid sizes. A necessary and sufficient condition for the convergence to the global minimum has been proven in 1984 by Geman and Geman[4] for the classical simulated annealing (CSA) based on a strictly local sampling. It is required that the time schedule of changing the fluctuation variance, described in terms of the artificial cooling temperature $T_a(t)$, which could be different from the true thermodynamic temperature T, is inversely proportional to a logarithmic function of time given a sufficient high initial temperature T_0.

$$T_a(t)/T_0 = 1/log(1+t)\,. \tag{1}$$

Such an artificial temperature cooling schedule is too slow to be practical. Instead, for arbitrary $T_o \neq 0$ the FSA has

$$T_c(t)/T_0 = 1/(1+t) \tag{2}$$

D-DIMENSIONAL CAUCHY PROBABILITY

Basically, the algorithm has three parts (1) States are generated with a probability density that has a Gaussian-like peak and Lorentzian wings that imply occasional long jumps

*NRL Invention Patent Case

0094-243X/86/1510420-6$3.00 Copyright 1986 American Institute of Physics

among local sampling. (2) The canonical ensemble for a state acceptance probability allows occasional hill-climbing among descents. (3) An artificial cooling temperature enters both (1) and (2) as a control parameter of noise. FSA turns out to be better than any algorithm based on any bounded variance distribution, which is equivalent to the Gaussian diffusion process by the central limiting theorem. Starting in a random state, at each time step a new state is generated according to the generating probability. If this new state has lower cost it becomes the new state of the machine. If it has higher cost it becomes the new state with the probability determined by the acceptance function. Otherwise the old state is retained. Both the acceptance and generating functions vary according to the cooling schedule. When $T_a = 0$, it is a gradient descent method. Since the diffusion process used for the strictly local strategy is artificial, it can be replaced with a semi-local search with an occasional long jump among local diffusions described by a Lorentzian distribution defined in the D dimension as follows

$$g_c(\vec{x}) = (2\pi)^{-D} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} d\vec{k} \exp(-i\vec{k}\vec{x}) \exp(-c|\vec{k}|) = c/[(\vec{x}^2 + c^2)]^{(D+1)/2} \quad (3)$$

which has the Cauchy characteristic function

$$\chi(\vec{k}) = \exp(-c|\vec{k}|) \quad (4)$$

The parameter c is the temperature parameter $T_c(t)$ which decreases according to a cooling schedule to be determined. The Cauchy distribution implies an occasional long jump among essentially local sampling over the phase space. This proper tradeoff between local search and semi-global search allows a fast annealing schedule.

PROOFS OF COOLING SCHEDULES FOR FSA AND CSA

One of the most significant consequences of such a trade off observation is that we are able to prove generally the cooling schedule to be inversely proportional to the time, rather than to the logarithmic function of time[4]. Since a rigorous theorem based on a stochastic Markovian chain will be published elsewhere[6], we shall compare (CSA) with (FSA) and sketch the essential proofs for both cooling schedules in the arbitrary D-dimensional vector space. In FSA we separate the state-generating from the state-visiting, while the actual visit is decided by the hill-climbing acceptance criterion based on the canonical ensemble of a specific Hamiltonian. FSA demands the state-generating to be infinite often in time (i.o.t.), but CSA requires the state visiting to be i.o.t. Let the state-generating probability at the cooling temperature $T_c(t)$ at the time t and within a neighborhood be (bounded below by) $\geqq g_t$. Then the probability of not generating a state in the neighborhood is obviously (bounded above by) $\leqq [1 - g_t]$. To insure a globally optimal solution for all temperatures, a state in an arbitrary neighborhood must be able to be degenerated i.o.t., which does not, however, imply the ergodicity that requires actual visits i.o.t. To prove that a specific cooling schedule maintains the state-generation i.o.t., it is easier to prove the *negation* of the *converse*, i.e. the *impossibility* of *never* generating a state in the neighborhood after an arbitrary time t_0, namely such a negation probability vanished

$$\prod_{t=t_0}^{\infty} [1 - g_t] = 0 \quad (5)$$

Taking logarithm of (5) and Taylor expansion (noting that $\log 0 = -\infty$, $\log(1 - g_t) \approx -g_t$), to prove (5) is equivalent to prove (6),

$$\sum_{t=t_0}^{\infty} g_t = \infty \quad (6)$$

We can now verify those cooling schedules satisfying Eq. (6) in the D-dimension neighborhood for an arbitrary size $|\Delta\vec{x}_0|$ and t_0.

(i) Bounded variance type CSA: there exists an initial T_0 and for $t > 0$

$$T_a(t) = T_0/\log(t) \tag{7}$$

$$g_t \approx \exp\left[-\frac{|\Delta\vec{x}_0|^2}{T_a(t)}\right] T_a(t)^{-D/2} \tag{8}$$

$$\sum_{t=t_0}^{\infty} g_t \geq \exp\left[-\log(t)\right] = \sum_{t=to}^{\infty} \frac{1}{t} = \infty \tag{9}$$

(ii) Unbounded variance type *GSA*: For arbitrary $T_0 > 0$

$$T_c(t) = T_0/t \tag{10}$$

$$g_t \approx \frac{T_c(t)}{\left[T_c^2(t) + |\Delta\vec{x}_0|^2\right]^{(D+1)/2}} \approx \frac{T_0}{t\,|\Delta\vec{x}_0|^{D+1}} \tag{11}$$

$$\sum_{t=t_0}^{\infty} g_t \approx \frac{T_0}{|\Delta\vec{x}_0|^{D+1}} \sum_{t=t_0}^{\infty} \frac{1}{t} = \infty \tag{12}$$

So any neighborhood is visited i.o.t. and the cooling schedule algorithm is admissible. The advantage of using Cauchy distribution in D-dimensions is that the ability to take advantage of locality is preserved, but the presence of a small number of very long jumps allows faster escape from local minima. As a result we can be much less cautious in our cooling. In fact, we can cool as fast as $T_c(t) = T_0/t$ for any $T_0 > 0$. Because the rate of convergence of the annealing algorithm is bounded by the temperature, this means that the algorithm can converge much faster.

EXAMPLE OF NONCONVEX OPTIMIZATIONS

In order to illustrate both FSA and CSA we choose a one dimensional simple double well potential as the classical energy

$$C(x) = x^4 - 16x^2 + 5x \tag{13}$$

as illustrated in Fig. 1. In order to appreciate the analogy with the transition probability in quantum mechanical we plot both the normal distribution and the Cauchy/Lorentzian distribution over the shallower valley representing a trapping in the valley. While the wing of Lorentzian probability has reached the other, deeper, valley, the normal distribution has negligible value there and thus has less chance to escape. A higher temperature implies a faster sampling in a much more "coarse grained" fashion. As the temperature is gradually reduced, the Cauchy machine searches through the state space with more refined sampling. An artificial control temperature within the search state space is the state-space-search generating temperature T_C, which is different to a thermodynamic temperature along the energy/cost function in a canonical ensemble. For simplicity, we let the two be proportional or equal to each other $T_C = T$ without causing any confusion.

We apply stochastic optimization to the simple cost function (13). Apart from the automatic learning aspects, Boltzmann machines may be characterized by (i) bounded generating probability density (thermal diffusion), e.g., Gaussian

$$G(x) \cong \exp\left(-x^2/T_a(t)^2\right) \tag{14}$$

(ii) an inversely logarithmic update cooling schedule Eq. (1) and (iii) the canonical hill-climbing acceptance probability (putting the Boltzmann constant $K_B = 1$, i.e. $C = H/K_B$)

$$\exp(-C_{t+1}/T_a)/[\exp-C_{t+1}/T_{a)} + \exp(-C_t/T_a)] = (1 + \exp(\Delta C/T_a(t)))^{-1}. \tag{15}$$

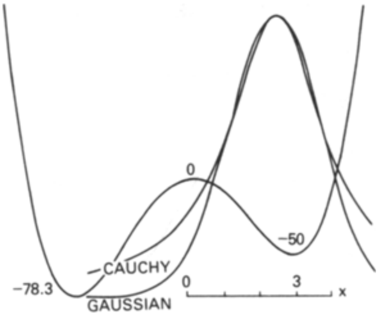

Fig. 1 — Gaussian probability density versus Cauchy probability density plotted over a simplified model of cost functions

Where $\Delta C = C_{t+1} - C_t$ is the increase of cost incurred by a transition. The resulting cost at each time step shows the validity of the inversely logarithmic cooling schedule (1) (also plotted as the dotted line in Fig. 2.). The energy axis is the vertical axis which shows the first minimum, the zero, and the second minimum at the level of the horizontal axis as visited by several thousand trials.

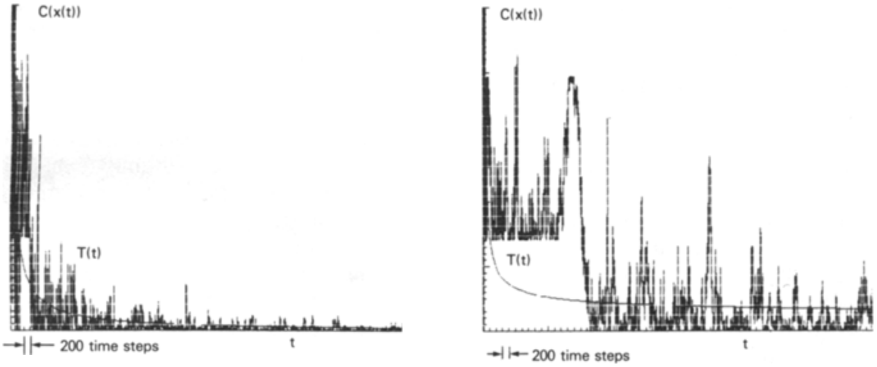

Fig. 2 — Actual cost $C(x(t))$ and cooling schedule $T(t)$ used in Boltzmann and Cauchy Machine are plotted against time steps $t = 1$ to 12000, where a tic mark is 200 time steps

Then we define the Cauchy Machine which replaces (i) the generating probability (14) with the Cauchy/Lorentzian distribution:

$$G_C\,(x) = T(t)/(T(t)^2 + x^2) \qquad (16)$$

Then, (ii) the update cooling schedule may be inversely linear in time, Eq. (2). For the sake of comparison, we use the identical hill-climbing acceptance probability (15) except $T_a(t)$ is replaced by $T_c(t)$. The success of the simulation shown in the right hand side of Fig. 3 supports our universal theorem of convergence for Cauchy machines, namely that the process finds the optimum with the cooling schedule Eq. (2). We shall first plot both the free-space random walk with displacements having the normal distribution together with the Cauchy random walk, and the actual random walk within the potential walls. It is evident that there is no bound on the variance of fluctuations of the Cauchy distribution. This provides us the opportunity of occasionally sampling the state space from one extreme to the other. Obviously, the Cauchy Machine is better in reaching and staying in the global minimum (shown in the left) as compared with the corresponding tracks generated by the Boltzmann machine (in the right).

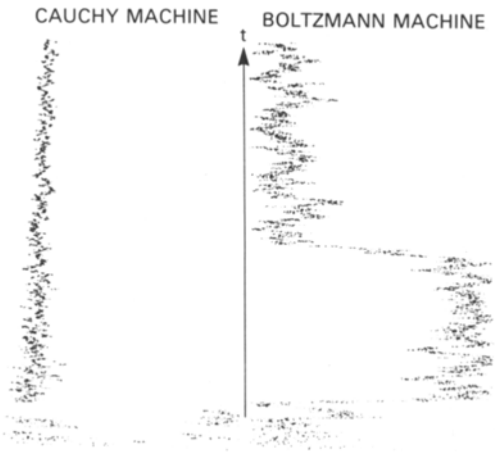

Fig. 3 — Comparison between Cauchy Machine search with Boltzmann Machine search (shown in right) within the identical cost constraint. The vertical axis is time steps and the horizontal is the displacement x.

We now turn to a conceptually simple but computationally complex application in a higher dimensional phase space. Given 100 points which have been randomly scattered from five lines (the ground truth), the problem is that to discover a best fit of those 100 points with five lines. This class of perception of random dot problem may be called "*unlabeled* (unknown correspondences between lines and points) mixture of densities" and to rediscover those labels and means is known as "*unsupervised* learning" in Computer Vision. In nonstochastic version, it is computationally complex or NP-difficult because there are 10^{10} possible ways to assign 100 random points to 5 groups, $(100)^5$ for a (dumb) exhaustive search. Furthermore, there exists ambiguous results and the unique solution cannot be guaranteed by the conventional gradient descent methodologies, which have no ability of de-trapping from local minima associated with each ambiguous result. A computationally complex NP-complete problem is the traveling salesman[2] problem which may be heuristically solved by the stochastic method of simulated annealing algorithm, and which would guarantee a unique solution if appropriate cooling schedules have been followed. The present example of "unsupervised learning" is solved using the standard Maximum likelihood formalism and FSA, and the result is presented.[5]

Basically, the Cauchy distribution helps us to preserve a local search and an occasional long jump in speed up the state-generation at an artificial noise temperature T_c which is conveniently separated from the thermodynamic temperature T used in physical distribution function $\exp(-H/k_B T)$ for the occasional hill climbing acceptance criterion of those generated states. Such a computational saving is comparable to what Tukey and Cooley did to the N^2 operations needed for 2D DFT with the observation of the harmonic pairing/butterfly giving the $N \log N$ operations needed for FFT. The computational saving of FSA when compared with CSA is similar to that FFT which revolutionized signal processing compared with DFT. Thus we can call it fast simulated annealing (FSA) which we hope should significantly broaden the applicability of simulated annealing to neural network computing[7] and physics problems.

References

1. N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, *J. Chem. Phys.* **21**, 1087-1092, June 1953.
2. S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, *Science* **220**, 671-680, 13 May 1983.
3. D. Ceperley and B. Alder, Science **231**, 555-560, 7 Feb. 1986.

4. S. Geman and D. Geman, *IEEE Trans, Patt, Anan. Mach. Int.*, **PAMI-6** (No. 6), 721-741, Nov. 1984.
5. H.H. Szu and R.L. Hartley, "Simulated Annealing with Cauchy Probability" submitted to Optics Letter.
6. R.L. Hartley and H.H. Szu, "Generalized Simulated Annealing," submitted for publication.
7. H.H. Szu, "Neural Network Models for Computing," to appear in Applied Optics.