Simulated Annealing for MAP State Estimation

A. Kehagias

October 11, 2001

Abstract

A simulated annealing method is presented for Maximum A Posteriori state estimation. Specifically, I consider the problems of smoothing and filtering. A minimax entropy interpretation is proposed to motivate the method.

1 Introduction

The problem addressed in this paper is the following. Consider a discrete time stochastic dynamic system with state vector x_n taking values in R^N, obeying the equation

$$x_n = f(x_{n-1}) + u_n, \tag{1}$$

and observation vector $y_n \in R^M$

$$y_n = g(x_n) + v_n. \tag{2}$$

Here, $u_n, v_n, n = 1, 2, ...$ are zero mean, white and independent of each other stochastic processes, with probability laws $p_{u_n}(u), p_{v_n}(v)$ respectively. Furthermore, we assume a random initial state x_0, with probability law $p_{x_0}(x)$, and independent of u_n, v_n. When the context is clear, I will also write $p(u_n), p(v_n), p(x_0)$.

Let $X^n = [x_0, ..., x_n], Y^n = [y_1, ..., y_n]$. The Maximum A Posteriori Estimation problem is to find vectors $Z \in R^{N \cdot n}, z \in R^N$ such that for given Y_n,

$$p_{X_n}(Z \mid Y_n), p_{x_n}(z \mid Y_n)$$

are maximized. (Here the vertical bars indicate conditional probabilities.) Maximization of the first probability is the smoothing problem, maximization of the second probability is the filtering problem.

2 Notation

Unless otherwise specified, lowercase subscripted letters refer to stochastic processes that take values in R^N, e.g. x_n. The same letters, with a bar or hat, refer to random variables, e.g. $\hat{x}_m$. Uppercase superscripted letters, refer to stochastic processes made up by a sequence of R_N-valued stochastic processes . E.g. $X^{t,n} = [x_{t,1}, ..., x_{t,n}]$. Bars and hats refer, again, to random variables.

In general, I use the following naming convention: an **estimator** is a s.p., e.g. $X^{t,n}$ and an **estimate** is a r.v., e.g. $\hat{X}^{m,n}$. For example, we could have the **estimator** converging to an **estimate**: $\lim_{t\to t_m} X^{t,n} = \hat{X}^{m,n}$. We have two levels of time here: time indicated by $m, n, n+1, ...$ is the time in which the dynamical system (1)-(2) evolves; time indicated by t is the time in which the estimation algorithm runs.

3 Smoothing - Preliminaries

Consider a fixed number of observations, n. Note that

$$p(X^n \mid Y^n) = \frac{p(Y^n \mid X^n)p(X^n)}{p(Y^n)}, \tag{3}$$

$$p(X^n) = p(x_n \mid x_{n-1})p(x_{n-1} \mid p(x_{n-2})...p(x_0) =$$
$$p_{u_n}(x_n - f(x_{n-1})p_{u_{n-1}}(x_{n-1} - f(x_{n-2}))...p(x_0), \tag{4}$$

$$p(Y^n \mid X^n) = p_{v_n, v_{n-1}, ..., v_1}(y_n - g(x_n), ..., y_1 - g(x_1)) =$$
$$p_{v_n}(y_n - g(x_n))p_{v_{n-1}}(y_{n-1} - g(x_{n-1}))...p_{v_1}(y_1 - g(x_1)). \tag{5}$$

Hence, for given Y^n, maximization of $p(X^n \mid Y^n)$ is equivalent to maximization of $p(Y^n \mid X^n)p(X^n) = U^*(X^n)$. But, we have

$$U^*(X^n) = p_{u_n}(x_n - f(x_{n-1}))p_{u_{n-1}}(x_{n-1} - f(x_{n-2}))...p_{x_0}(x_0)\cdot$$
$$p_{v_n}(y_n - g(x_n))...p_{v_1}(y_1 - g(x_1)). \tag{6}$$

Define $U(X^n) = -logU^*(X^n)$: the negative log-likelihood function of X^n. Maximization of $U^*(X^n)$ is equivalent to minimization of $U(X^n)$. By defining functions $U_0(.) = -log\, p_{x_0}(.), U_m(.) = -logp_{u_m}(.), V_m(.) = -log\, p_{v_m}(.)$,

we can write

$$U(X^n) = U_0(x_0) + \sum_{m=1}^{n} \left[U_m(x_m(x_m - f(x_{m-1})) + V_m(y_m - g(x_m)) \right]. \quad (7)$$

Now, take the usual aproximation of the continuous valued variables x_m by discrete variables x_m, taking values $x_m \in \Lambda = \{\lambda_1, ..., \lambda_L\}$. Assume the approximation is good enough that minimization of $U(X^n)$ over $R^{N \cdot n}$ is equivalent to minimization over $\Omega = \Lambda^n$.

4 Smoothing by Simulated Annealing

The method of simulated annealing was introduced by Kirkpatrick [4] for combinatorial optimization problems. Geman and Geman [3] proved important convergence results and used the method for image restoration. The present work is very much in line with their paper.

The function $U(X^n)$ has a number of global minima $X_1^n, ..., X_K^n \in R^{N \cdot n}$. The smoothing problem is to find one of them. Simulated annealing solves the smoothing problem in the following way: we will consider a stochastic process $X^{t,n}$ which will be constructed in such a way as to ensure that the estimator $X^{t,n}$ will converge to one of the minima of U. More exactly, define the random variable $\hat{X}^n$ which takes values in the set $\{X_1^n, ..., X_K^n\}$ with a uniform probability. Then we will construct $X^{t,n}$ in such a way that it converges in distribution to $\hat{X}^n$, as $t \to \infty$.

To achieve this convergence, perform the following algorithm:

Homogeneous Annealing Algorithm

Given a positive function of time, $T(t)$,
and a random variable $\hat{Z}$, taking values in Ω,

choose $X^{0,n}$ arbitrarily and, given $X^{t,n}$,
choose $X^{t+1,n}$ in the following way:

Pick $\hat{Z} \in \Omega$ randomly and, given $T(t)$, compute q:

$$q = \frac{exp(-\frac{U(\hat{Z})}{T(t)})}{exp(-\frac{U(X^{t,n})}{T(t)})}. \quad (8)$$

`If` $q \geq 1$ $X^{t+1,n} = \hat{Z}$ `with probability 1.`
`If` $q < 1$ $X^{t+1,n} = \hat{Z}$ `with probability q (else` $X^{t+1,n} = X^{t,n}$`).`
`Continue for the next` t.

In this way we generate a stochastic process $X^{t,n}$. In [3] it is proven that for $X^{t,n} \rightarrow \hat{X}_n$, it is sufficient that two conditions are satisfied. The first condition has to do with the probability law of $hatZ$ and further details can be found in [3]. The second condition is on $T(t)$, the "temperature" parameter: $T(t) \rightarrow 0$ at a slow enough pace - specifically

$$T(t) \geq \frac{K}{logt}. \tag{9}$$

If the above conditions hold, then

$$\lim_{t \rightarrow \infty} X^{t,n} = \hat{X}^n$$

(the limit is in the distribution sense). In practice we start the algorithm at a temperature T_i and end at $T_f = \epsilon > 0$. Assuming that at any time we use the lowest admissible value of $T(t) = K/logt$ we have a total execution time of

$$\Delta t = exp(K/T_f) - exp(K/T_i). \tag{10}$$

Here, let us consider only annealing schedules that are piecewise constant:

$$T(t) = \begin{cases} T_1 & t_0 \leq t \leq t_1 \\ T_2 & t_1 \leq t \leq t_2 \\ ... & ... \\ T_n & t_{n-1} \leq t \leq t_n \end{cases}$$

and respect the logarithmic bound of eq.(9). Then the execution time is essentiallly the same as in eq.(10).

Furthermore from [3] we know that for $t_1 - t_0,\ t_2 - t_1, ...$ big enough the stochastic process $X^{t,n} \rightarrow \hat{X}^{m,n}$ (for $t \rightarrow t_m$) where $\hat{X}^{m,n}$ is a random variable (the estimate) with

$$\pi_i^m = p(X^m = X_i) = \frac{exp(-\frac{1}{T_m}U(X_i)}{\sum_{X \in \Omega}^{J} exp(-\frac{1}{T_m}U(X)} \qquad (X_i \in \Omega) \tag{12}$$

So, in the limit $t \rightarrow t_m$, the estimate has a distribution $\pi^m = \{\pi_1, ..., \pi_j\}$ and, accordingly, an average log-likelihood

$$E(-U(\hat{X}^{m,n})) = G_m. \tag{13}$$

Define

$$G(t) = G_m \quad for \quad t_{m-1} \leq t \leq t_m. \tag{14}$$

Now, Jaynes [1,2] proves that the equilibrium distribution π^m is the maximum entropy distribution of all distributions satisfying (13). Obviously, G_m in (13) is a measure of the likelihood of $\hat{X}^{m,n}$ (as an estimate of X^n).

What happens then in our annealing scheme is the following: For a given temperature we pick the max entropy estimate subject to fixed average likelihood. The max entropy estimate is the most honest, or most efficient one to choose (see [2]). However, we actually want an estimate with small entropy (ideally, in the limit we would like to have an estimate with zero entropy - it would take just one value with probability 1 and that value would maximize U).

This is resolved, as temperature drops to lower and lower levels. First of all, as temperature decreases, expected likelihood increases. This is true because

$$G(t) = \frac{d}{d(-1/T(t))}(log \sum_{X \in \Omega} exp(-\frac{1}{T(t)}U(X)) \tag{12}$$

$$\frac{dG}{dT} \leq 0. \tag{16}$$

And secondly, for the entropy, defined by

$$S(t) = \sum_{X \in \Omega} \frac{1}{T(t)} U(X) \frac{exp(-\frac{1}{T(t)}U(X))}{C(T(t))} \; - \; logC(T(t)), \tag{17}$$

$$\frac{dS}{dT} \geq 0. \tag{18}$$

At every temperature step then ($t_{m-1} \leq t \leq t_m$), we effectively fix a likelihood level G_m and pick the maximum entropy estimate with such likelihood. When going to a lower temperature, we increase the likelihood ($T_m \downarrow$, so $G_m \uparrow$) and decrease the maximum entropy. This is then a minimax procedure on the entropy, with a simultaneous increase of the likelihood of the estimator. In the limit of $T(t) \to 0$ we get a very likely estimate with a low entropy.

5 Filtering

For filtering we would like to maximize $p_{x_n}(z \mid Y^n)$. It turns out this is a rather difficult task. In a sense, it is better anyway to maximize $p_{X^n}(Z \mid Y^n)$,

but we also have to work harder. The power of linear systems filtering [5] is in that it can be done in a recursive way. Simulated annealing can be used for nonlinear problems but it is not obvious how to do this recursively. The general idea of recursive filtering is to use some of the previous work in the next step.

We now want to minimize

$$U(X^{n+1}) = -logp_{x_0}(x_0) - \sum_{m=1}^{n} [logp_{u_m}(x_m - f(x_{m-1}) + logp_{v_m}(y_m - g(x_m)]$$

$$-logp_{u_{n+1}}(x_{n+1} - f(x_n)) - logp_{v_{n+1}}(y_{n+1} - g(x_{n+1}). \tag{19}$$

For the purpose of this discussion, use the following terminology: At time n we have an estimate $\hat{X}^n$ and at time n+1 an estimate $\hat{X}^{n+1} = [\bar{X^n}, \hat{x}_{n+1}]$. We will call $\bar{X}^n$ the "first part" of $\hat{X^{n+1}}$ (or $X^{t,n+1}$) and $\bar{x}_{n+1}$ the "last part". We will use "first" and "last" in the same sense for the estimator $X^{t,n+1}$.

$\hat{X}^n$, $\bar{X}^n$ are estimates of the same quantity. It is intuitively obvious, that the new observation y_{n+1} will not completely invalidate our old estimate, and, in fact, we can be fairly confident that $\hat{X}^n \simeq \bar{X}^n$. At any rate, it is a good idea to start the new annealing algorithm with the old estimate as initial value for the first part of $X^{t,n+1}$.

On the other hand, the choice of the initial temperature, as indicated in the previous section, is related to the confidence we have in the estimate. If we consider a piecewise constant cooling schedule, as we go to lower and lower temperatures, we consider classes of more and more likely estimates.

Furthermore our state of higher confidence in the first part of the estimate must be reflected in the annealing scheme we will use.

This is feasible by the Jaynes interpretation of maximum entropy methods. In place of one constraint (eq.(13)) on the likelihood of $\hat{X}^{m,n+1}$, let us use two constraints, one on the likelihood of $\hat{X}^{m,n}$ and one on the likelihood of $\hat{x}_{m,n+1}$. To be more specific, define the functions

$$U_1(X^n) = -log\, p_{x_0}(x_0) -$$

$$\sum_{m=1}^{n} [log\, p_{u_m}(x_m - f(x_{m-1})) + logp_{v_m}(y_m - g(x_m))], \tag{20}$$

$$U_2(x_n, x_{n+1}) = -log\, p_{u_{n+1}}(x_{n+1} - f(x_n)) - logp_{v_{n+1}}(y_{n+1} - g(x_{n+1})). \tag{21}$$

U_1 is the negative log-likelihood of $\hat{X}^n$, U_2 is the negative log-likelihood of $\hat{x}_{n+1}$ conditioned on $\hat{x}_n$. If we set fixed levels for the expectations of these two functions

$$E(-U_1(\hat{X}^{m,n})) = G_m^1, \tag{22}$$

$$E(-U_2(\hat{x}_{m,n}, \hat{x}_{m,n+1})) = G_m^2, \tag{23}$$

then the maximum entropy estimate $\hat{X}^{m,n+1}$ satisfying (22), (23) has the following distribution

$$p(X^{m,n+1} = (z_1, ..., z_{n+1})) =$$

$$\frac{1}{C(T_1, T_2)} \cdot exp(-\frac{1}{T_1}U_1(z_1, ..., z_n) - \frac{1}{T_2}U_2(z_n, z_{n+1})). \tag{24}$$

Assume $max|\frac{U_2(Z)}{U_1(Z)}| \ll 1$. (this would be no problem for laws occuring in practice, e.g. Gaussians).Then, computing $\frac{dG_m^1}{dT_1}$, $\frac{d(G_m^1 - G_m^2)}{dT_1}$, we find they are always negative. That is, when we decrease T_1, we increase G_m^1 and, furthermore, we increase it more than G_m^2. The conclusion is: an estimate $\hat{X}^m$ with a probability law given by (24), is more reliable in its first part: $(\hat{x}_{m,1}, ..., \hat{x}_{m,n})$, than in its last: $\hat{x}_{m,n+1}$. This agrees well with the way we want to weigh the parts of $\hat{X}^{m,n+1}$. How to get an estimate with the law (24)? From [3] we know it is the limit of a stochastic process $X^{t,n+1}$, generated by an algorithm like the one described in the previous section, the only difference being in that we now have two functions: $T_1(t), T_2(t)$ and the quantity q is given by

$$q = \frac{exp(-\frac{1}{T_1(t)}U_1(z_1, ..., z_n) - \frac{1}{T_2(t)}U_2(z_n, z_{n+1}))}{exp(-\frac{1}{T_1(t)}U_1(X^{t,n}) - \frac{1}{T_2(t)}U_2(x_{t,n}, x_{t,n+1}))}, \tag{25}$$

With $T_1(t) = T_1,\ T_2(t) = T_2$, the algorithm gives, in the limit $t \to t_m$, a r.v. with the law (24). With $T_1(t), T_2(t)$ as in (11), it is an annealing algorithm; call it Inhomogeneous Annealing Algorithm.

So we will get the max entropy estimate in a given class. But what we are really interested is a fast annealing schedule that will yield the minimum of $U = U_1 + U_2$. To achieve this there are two points to arrange.

First, as we proceed in the computation, we would like the difference in the likelihood of the two parts of the estimator to decrease. (This is because our initial uninformed guess $x_{t,n+1},\ t = 0$ gets closer to the optimal value as $t \to \infty$.) An obvious way to have the difference decrease, is to gradually diminish the difference $\Delta T = T_m^1 - T_m^2$. Specifically, take $T_m^2(t) =$

$h(t)T_m^1(t),\ h(t) \downarrow 1$. This has the effect of gradually equalizing the confidence we have in the two parts of the estimate. In addition it allows us to start the system at a lower temperature (hence fewer iterations will be needed). We only want to keep the last part of the system at a high temperature and we can cool this faster.

Second, inhomogeneous annealing of the function $U = U_1 + U_2$ according to (25) is equivalent to homogeneous annealing of the function $U^* = U_1 + (T_1/T_2) \cdot U_2$. Consequently, the actual minimum to be reached will be a minimum of U^*. However, again from [3], we know that the convergence is not influenced by initial values. So, suppose at a given time, late in the annealing schedule, when $T_m^1/T_m^2 \simeq 1$, $T_m^1, T_m^2 \simeq 0$, we fix T_m^1, T_m^2 and keep them constant for the rest of the process. The whole process, then will yield a good approximation to a minimum of $U^* \simeq U$.

The above two observations indicate that applying the inhomogeneous annealing algorithm at time n+1 with initial values as determined from time n, T_1 lower than T_2, T_2 cooling faster and asymptotically approaching T_1, will yield a good approximation to the minimum of $p(X^{n+1} \mid Y^{n+1})$ in a shorter time then what would be required had we done the problem completely from scratch.

6 Conclusion

We have looked at two problems: MAP smoothing and MAP filtering of observations of nonlinear dynamical systems. These are both hard problems for which no satisfactory and general solution exists to date. Smoothing can be done as a straightforward extension of the Homogeneous Annealing Algorithm. The ease of ipmplementation of this algorithm for any kind of nonlinear system and the guaranteed convergence make it very attractive. Filtering is a harder problem, but the Inhomogeneous Annealing Algorithm makes a more efficient computation possible and retains simplicity of implementation and guaranteed convergence.

7 References

1. E.T. Jaynes, "On the Rationale of Maximum-Entropy Methods", Proc. IEEE, Vol.70, No.9, Sept. 1982.

2. E.T. Jaynes, "Prior Probabilities", IEEE SSC-4, No.3, Sep. 1968.

3. S. Geman, D. Geman, "Stochastc Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", IEEE PAMI-6, No.6, Nov. 1984.

4. S. Kirkpatrick, C.D. Gellatt Jr., M.P. Vecchi, "Optimization by Simlated Annealin"g, IBM ThomasWatson Research Center, NY 1982.

5. R. Kalman, "New Results in Filtering Theory", ASME J. of Basic Eng., March 1960.