Time difference simultaneous perturbation method

Y. Maeda

Indexing terms: Neural networks, Stochastic approximation, Simultaneous perturbation

The authors propose a heuristic recursive algorithm to find the minimum point of a function without using the gradient of the function. This algorithm is based on the time diiference and simultaneous perturbation method. However, it does not need an additional measurement of the function to update the estimated point.

Introduction: In this Letter we consider a simple optimisation problem: finding the optimal parameter that minimises a function, while supposing that we cannot use the gradient of the function. The most basic recursive approach to the problem is the finite difference method; we add a small perturbation to a point and make measurements at the point and the perturbed point. Using a difference approximation, we can estimate the gradient of the function at the point. This estimated gradient is used to modify the point. However, if the dimension of the parameter n increases, (n large) we have to repeat this procedure n times.

The simultaneous perturbation (SP) technique overcomes some of these problems. Instead of adding perturbation to all parameters one by one, the SP technique adds perturbations to all parameters simultaneously. Then, we can obtain an estimated gradient such as the finite difference. In this procedure twice the number of measurements are required, even if n is large. This method was originally proposed by Spall [1 – 3], and independently devised by the author [4] and Alespector *et al.* [5]. Furthermore, some applications of this method are reported [6 – 8]. In this method we need twice the measurements of the function.

We propose a heuristic method based on a time difference and SP technique. Basically we use the SP idea. However, we do not make the additional measurement with a perturbed point. When we update an estimator of the optimal point, we use an estimated gradient vector and a perturbation vector simultaneously as a modifying quantity of the point. The estimated gradient of the function derives from past measurements using the SP technique. The perturbation is necessary to estimate the gradient in the next iteration.

Time difference simultaneous perturbation method: Let $\mathbf{u} \in \Re^n$ be a parameter vector, and $J(u)$ be a function to be minimised. We would like to find an optimal point $\mathbf{u}_* = \min_{\mathbf{u}} J(\mathbf{u})$. We propose the following algorithm:

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \alpha \Delta \mathbf{u}_t + c\mathbf{s}_t \qquad (1)$$

$$\Delta u_{t,i} = \begin{cases} \Delta u_{max} & \text{if } \frac{J(\mathbf{u}_t)-J(\mathbf{u}_{t-1})}{(cs_{t-1,i})} > \Delta u_{max} \\ -\Delta u_{max} & \text{if } \frac{J(\mathbf{u}_t)-J(\mathbf{u}_{t-1})}{(cs_{t-1,i})} < -\Delta u_{max} \\ \frac{J(\mathbf{u}_t)-J(\mathbf{u}_{t-1})}{(cs_{t-1,i})} & \text{otherwise} \end{cases} \qquad (2)$$

where $\Delta u_{t,i}$ denotes the ith component of the vector Δu_t. α is a positive gain coefficient. s_t is a sign vector whose components are +1 or –1 and $E(s_t) = 0$. $s_{t-1,i}$ represents the ith component of the vector s_t and $E(s_{t-1,i}\, s_{t-1,j}) = 0$ $(i \neq j)$. c (>0) denotes the magnitude of the perturbation. By means of the term cs_t, random perturbations $+c$ or $-c$ are added to all parameters. Signs of perturbations are different in different parameters.

In eqn. 1, the vector Δu_t represents an estimated gradient vector derived from the second procedure, eqn. 2. Perturbations are added to all parameters simultaneously by the third term of eqn. 1. Since the expectation of the sign vector is zero, the estimator u_t is updated only by the second term in terms of expectation.

In eqn. 2 the difference between the values of the function at time t and time $(t-1)$ is divided by the magnitude of the perturbation. This gives an estimated gradient. However, $J(u_t)$ includes the effect of Δu_{t-1}. When we expand $J(u_t)$ at u_{t-1}, there exists u_m such that

$$\begin{aligned} J(\mathbf{u}_t) = J(\mathbf{u}_{t-1}) &- (\alpha \Delta \mathbf{u}_{t-1} - c\mathbf{s}_{t-1})^T J'(\mathbf{u}_{t-1}) \\ &+ (\alpha \Delta \mathbf{u}_{t-1} - c\mathbf{s}_{t-1})^T J''(\mathbf{u}_m)(\alpha \Delta \mathbf{u}_{t-1} - c\mathbf{s}_{t-1}) \end{aligned} \qquad (3)$$

Therefore, since $s_{t-1,i} = \pm 1$, we have

$$\begin{aligned} \Delta u_{t,i} &= \frac{J(\mathbf{u}_t) - J(\mathbf{u}_{t-1})}{c} s_{t-1,i} \\ &= \frac{(\alpha \Delta \mathbf{u}_{t-1} + c\mathbf{s}_{t-1})^T J'(\mathbf{u}_{t-1})}{c} s_{t-1,i} \\ &+ \frac{(\alpha \Delta \mathbf{u}_{t-1} + c\mathbf{s}_{t-1})^T \mathbf{J}''(\mathbf{u}_m)(\alpha \Delta \mathbf{u}_{t-1} + c\mathbf{s}_{t-1})}{c} s_{t-1,i} \\ &= \frac{(\alpha \Delta u_{t-1,1} + cs_{t-1,1})}{c} s_{t-1,i} J'_1(\mathbf{u}_{t-1}) + \cdots \\ &+ \frac{(\alpha \Delta u_{t-1,i} + cs_{t-1,i})}{c} s_{t-1,i} J'_i(\mathbf{u}_{t-1}) + \cdots \\ &+ \frac{(\alpha \Delta u_{t-1,n} + cs_{t-1,n})}{c} s_{t-1,i} J'_n(\mathbf{u}_{t-1}) \end{aligned}$$

$$+\frac{(\alpha\Delta \mathbf{u}_{t-1}+c\mathbf{s}_{t-1})^T \mathbf{J}''(\mathbf{u}_m)(\alpha\Delta \mathbf{u}_{t-1}+c\mathbf{s}_{t-1})}{c} s_{t-1,i} \quad (4)$$

where $J_i'(\mathbf{u}_{t-1}) = \partial J(\mathbf{u}_{t-1})/\partial u_{t-1,i}$. $E(s^3_{t-1,i}) = 0$ because $s^2_{t-1,i} = 1$ and $E(s_{t-1,i}) = 0$. Thus, the expectation of the last term of the right-hand side of eqn. 4 is zero. Moreover, since $E(s_{t-1,i}\, s_{t-1,j}) = 0$ $(i \neq j)$ and $E(s_{t-1,i}) = 0$, taking the expectation of eqn. 4 gives

$$E(\Delta u_{t,i}) = J_i'(\mathbf{u}_{t-1}) \quad (5)$$

That is, the modifying vector Δu_t is the gradient of the function in the terms of the expectation, i.e. this procedure is a stochastic gradient method similar to the SP method. Details of this method is shown in Fig. 1.

begin
- Measure a value of the function $J(\mathbf{u}_t)$.
- Subtract the value $J(\mathbf{u}_t)$ from the previous value of $J(\mathbf{u}_{t-1})$.
  (* Obtain time difference of the function *)
  for $i := 1$ to n do
  begin
  - Multiply the time difference by $\alpha s_{t-1,i}/c$.
    (* $\alpha \Delta u_{t,i}$ *)
  - Generate new sign randomly. (* $s_{t,i}$ *)
  - Add the perturbation to $\alpha \Delta u_{t,i}$. (* $\alpha \Delta u_{t,i} + c s_{t,i}$ *)
  - Update the parameter.

  end;

  Renew the iteration (* $t = t + 1$ *)

end.

Fig. 1 *Algorithm*

Simulation: To confirm the viability of this method, we apply this to a simple multidimensional optimisation problem. The function is as follows:

$$J(\mathbf{u}) = u_1^2+2u_2^2+3u_3^2+4u_4^2+5u_5^2+u_1u_5+u_2u_4+u_1u_3 \quad (6)$$

As initial values, we set up $u_{1,1} = u_{1,2} = u_{1,3} = u_{1,4} = u_{1,5} = 5.0$. The result of the time difference simultaneous perturbation method using eqns. 1 and 2 is shown in Fig. 2. The magnitude of the perturbation c is 0.01 and α is 0.001. Δu_{max} is 0.1. These are determined empirically.

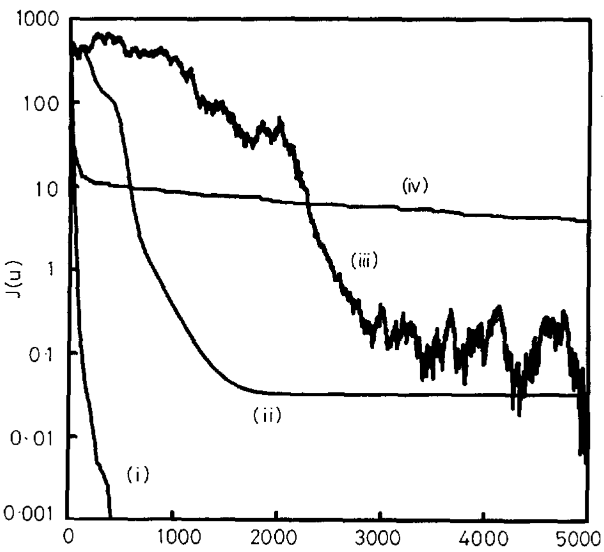

Fig. 2 *Simulation result*

(i) SP
(ii) finite difference
(iii) time difference simultaneous perturbation
(iv) random search

Moreover, to compare with our result, we show results using the simple random search, the finite difference technique and the SP technique. In the random search used here, random vectors in $[-5\ 5]^5$ are generated using uniform distribution. The result shown in Fig. 2 is an average for 20 trials.

In the finite difference method, a difference approximation is used instead of the gradient. We fix on the constant perturbation c (=0.1). The gain coefficient is 0.01. To guarantee the convergence of the estimator, the perturbation has to tend to zero, whether or not it is quadric. As was expected, the estimator did not converge to zero.

Details of the SP used here are described in [4]. In this case the gain coefficient was 0.01 and the magnitude of the perturbation was 0.01. As a result, the SP method is superior to the other methods. The time difference SP method is better than the random search but inferior to the SP and the finite difference.

However, as we stated before, there are some applications where the SP or the finite difference methods are not applicable. We can provide another option. Like the other methods, this algorithm contains some adjustable coefficients. The suitable coefficients depend on an unknown objective function. However, we did find a heuristic strategy.

The role of Δu_{max} is preventing a remiss modification via an error in the estimated gradient. However, too small a value of Δu_{max} will cause a slower convergence when the estimated gradient is nearly correct or an estimator is far from an optimal point. Conversely, in the neighbourhood of the optimal point, a smaller Δu_{max} is preferable.

Eqn. 2 gives the estimated gradient using SP. Then, we must add only perturbation to the parameters and measure the value of the function. This process is eqn. 1 in my method. However, eqn. 1 includes not only the perturbation in the third term but also the modification term of the second term. This causes the error in the estimated gradient. In this sense it seems that a smaller coefficient α and/or a larger perturbation give a better estimated gradient. However, if the perturbation term is larger than the modification term in eqn. 1, modification of the parameters is not carried out properly because of the perturbation term. Good results were obtained when the magnitudes of the modification term and the perturbation term are balanced. The gradient of the function varies, depending on the position of the estimator. However, that would be a clue to determining the gain coefficient α and the perturbation c.

To guarantee the convergence of this algorithm, the perturbation c must tend to zero, because the perturbation always disturbs movements of the estimator to a proper direction in eqn. 1, and the constant perturbation does not give an exact estimation of the gradient in eqn. 2. Since the perturbation is constant in this example, it was foreseen that the estimator will remain in the neighbourhood of the optimal point.

Conclusion: A heuristic optimisation method has been proposed, based on a combination of time difference and the SP method. The ordinary finite difference needs n times measurements, and the SP method needs twice the measurements to obtain a modifying quantity for all parameters in every iteration. However, the time difference simultaneous perturbation method proposed here needs no additional measurement. We can anticipate some applications of this technique, such as neural networks. At the same time, proving the convergence of the algorithm and accelerating the convergence speed are important. For stochastic approximation we need further study.

19 February 1996
Electronics Letters Online No: 19960637

Y. Maeda (*Kansai University, Faculty of Engineering, Department of Electrical Engineering, 3-3-35, Yamate-cho, Suita-chi, Osaka 564 Japan*)

Y. Maeda: Currently visiting the Department of Electrical and Computer Engineering, University of California at Irvine, CA, USA

References

1 SPALL, J.C.: 'A stochastic approximation technique for generating maximum likelihood parameter estimates'. Proc. of the 1987 American Control Conference, 1987, pp. 1161–1167

2 SPALL, J.C.: 'A stochastic approximation algorithm for large-dimensional systems in the Kiefer-Wolfowitz setting'. Proc. of the 27th IEEE Conference on Decision and Control, 1988, pp. 1544–1548

3 SPALL, J.C.: 'Multivariable stochastic approximation using a simultaneous perturbation gradient approximation', *IEEE Trans.*, 1992, **AC-37**, pp. 332–341

4 MAEDA, Y., HIRANO, H., and KANATA, Y.: 'A learning rule of neural networks via simultaneous perturbation and its hardware implementation', *Neural Netw.*, 1995, **8**, pp. 251–259

5 ALSPECTOR, J., MEIR, R., YUHAS, B., JAYAKUMAR, A., and LIPPE, D.: 'A parallel gradient descent method for learning in analog VLSI neural networks', *in* HANSON, S.J., COWAN, J.D., and LEE, C. (Eds.): 'Advances in neural information processing systems 5' (Morgan Kaufmann Publisher, San Mateo, CA, 1993), pp. 836–844

6 SPALL, J.C., and CRISTION, J.A.: 'Nonlinear adaptive control using neural networks: Estimation with a smoothed form of simultaneous perturbation gradient approximation', *Stat. Sin.*, 1994, **4**, pp. 1–27

7 SPALL, J.C., and CHIN, D.C.: 'A model-free approach to optimal signal light timing for system-wide traffic control'. Proc. of the 1994 IEEE Conference on Decision and Control, 1994, pp. 1868–1875

8 MAEDA, Y., and KANATA, Y.: 'A learning rule of neural networks for neuro-controller'. Proc. of the 1995 World Congress of Neural Networks, 1995, Vol. 2, pp. II-402–II-405