CONVERGENCE RATE ACCELERATION OF SUCCESSIVE APPROXIMATION ALGORITHMS IN REAL-TIME CASE

By

I. BENCSIK

Department of Automation, Technical University Budapest

Received July 10, 1973

Presented by Prof. Dr. F. CSÁKI

Introduction

The exact study of convergence rate is due to CIPKIN [1] proving that no successive approximation algorithm giving better result than the least squares method can be conceived. In the literature great many various approximation algorithms based on learning principles are found. The algorithm of SARIDIS [3] is one of the most outstanding. The convergence of these types of algorithms is usually proved for an infinite number of steps with unit probability. In the case of real-time applications a relationship between the expected number of steps and a convergence probability less than one is necessary, based on the error calculation procedure of the MONTECARLO [2] methods.

In this paper an unusual learning principle is presented. The question is whether in real-time applications the expected number of steps obtained in [2] may be ensured in the case of stochastic signals.

The learning principle

The relationship between the less than 1 convergence probability and the expected number of steps is given by [1] for very general conditions, as

$$N_{opt} = \frac{9p}{1-p}. \tag{1}$$

E. g. in the case of $p = 0.95$, $N_{opt} = 170$ and this is also a function of the "arresting conditions". Be the convergence probability of the successive approximation algorithm

$$\mathbf{c}[n] = \mathbf{c}[n-1] + \delta\mathbf{c}[n-1] \tag{2}$$

where n is the number of steps and $\mathbf{c}$ the vector approximating $\mathbf{c}^*$ in an infinite number of steps and in the unimodal case

$$P = \left\{\lim_{n\to\infty} (c[n] - c^*) = 0\right\} = 1 \tag{3}$$

and in a finite number of steps:

$$P\ \{M[F(\mathbf{c}[\mathrm{N}_{opt}] - c^*) = 0]\} = p \tag{4}$$

where F denotes the scalar function, the quality criterion and M the formation of the expected value.

Be τ the time of the calculations needed in one step; then the total calculation time in the convergent case is:

$$T = \mathrm{N}_{opt} \cdot \tau. \tag{5}$$

The problem variables are assumed to be slightly non-stationary stochastic signals with 0 centre and a noise load of finite scatter. According to the experiences the convergence rate much depends on the number and order of magnitude of the variables; for eliminating this dependence, in every case the following normalization procedure was applied:

$$v = \frac{x_m - \bar{x}}{\sigma_x} \tag{6}$$

where the upper stroke denotes the formation of the expected value, x_m the vector of the measured variables and σ_x the scatter. We have used as algorithm a version of the stochastic approximation developed by us, possessing good convergence properties even in the case of high gradient values:

$$\delta c[n] = -\mathbf{R}[n+1]\ \mathrm{arsh}\ \frac{\partial}{\partial c[n]} f[c[n], x[n+1]] \tag{7}$$

where $\mathbf{R}[n]$ is the matrix of the convergence coefficient. The derivations concerning the convergence and the applicability of algorithms of the type (7) are found in [1]. The convergence is especially sensitive for the selection of $\mathbf{R}[n]$ by the usual hyperbolic selection. A more complex selection of $\mathbf{R}[n]$ leads to the increase of τ, the calculation time and on the other hand, — according to the experiences of running the algorithm, — convergence problems arise even so in the real-time case. The hyperbolic selection is intended to ensure the $1/n$-type course of the approximation. The basic idea of the developed algorithm is that we demand the $1/n$-type course of the quantity $|\ \delta c[n]\ |$ and calculate from it in each step the convergence coefficient $\mathbf{R}[n]$ and refrain from all other constraints applying to the convergence coefficient. This means the reversal of the problem as regards the convergence coefficient: if in the $1/n$-

type course the quantity $|\delta c[n]|$ is decreasing faster, then the coefficient $\mathbf{R}[n+1]$ decreases while in the opposite case it increase. The evaluation is easily possible, as the data of the preceding step are always available, but an immediate question is the selection of the initial value $\mathbf{R}[1]$ and the manner of varying $\mathbf{R}[n]$.

The learning strategy

It is obvious, — e. g. in the case of divergence, — that the variation of $\mathbf{R}[n]$ must be faster than $1/n$ for ensuring the $1/n$-type course of the quantity $|\delta c[n]|$. We have chosen arbitrarily the base number of the extreme and mean ratio according to (8):

$$\mathbf{R}[n] = \mathbf{R}[n-1] \cdot 1.62^{\pm 1} \tag{8}$$

where the sign + applies for the "slow" and the sign — for the "fast" case. For the initial value $\mathbf{R}[1]$ we have:

$$r_{ik} = \left| \frac{c_i[0]}{c_i[1]} \right| \tag{9}$$

if $$|c_i[0]| > |c_i[1]|$$

and $$r_{ik} = \left| \frac{c_i[1]}{c_i[0]} \right|$$

respectively, if $$|c_i[1]| > |c_i[0]$$

and in the case of $r_{ik}[1] > 1.62$ we demand that $r_{ik}[1] = 1.62$ applies. In deterministic problems, — e. g. finding extreme values, — this algorithm showed the expected behaviour, but in stochastic cases it slowed down due to the variability of the signals; consequently it is advised to specify a minimum for the elements of $\mathbf{R}[n]$. We have performed this in the following manner:

$$r_{ik\,min} = \frac{1}{5} \sum_{n=6}^{10} r_{ik}[n], \tag{10}$$

i.e. in the first five steps we calculate a "good" initial value, but as according to the experiences the descent from the fifth step on is not $1/n$-type anymore, the demand of the minimum may be abandoned at higher step numbers.

One more condition may be built in against the deceleration of the algorithm, namely:

$$|\delta c[n-1]| - |\delta c[n]| < \lambda |\delta c[n-1]| \tag{11}$$

to be satisfied by

$$\lambda = \frac{1}{1.57} \operatorname{arc\,tg} 5 r_{ik} \min \tag{12}$$

The constant 1.57 is used for the radian conversion to make $\lambda_{max} = 1$. If (11) is not satisfied, then

$$\mathbf{R}[n] = 1.62\mathbf{R}[n-1] \tag{13}$$

and in the opposite case

$$\mathbf{R}[n] = 1.62^{-1}\,\mathbf{R}[n-1]\,. \tag{14}$$

The relationship (14) is applied also in the case of divergence, when

$$|\,\delta c[n-1]\,| < |\,\delta c[n]\,|$$

as well as when the course is faster than the $1/n$-type behaviour, i.e. when

$$|\,\delta c[n]\,| < |\,\delta c\,[n-1]\,|\,\left(1 - \frac{1}{n}\right).$$

Conclusions

In the algorithm obtained in this way we have utilized in the first step a scalar coefficient instead of the convergence coefficient matrix $\mathbf{R}[n]$ and we have examined, instead of the vector-vector function $\mathbf{c}\,[n]$ the $1/n$-type course of the error $\Delta y = y_{measured} - y_{calculated}$ of a scalar vector function of the form

$$y_{sz} = \mathbf{c}^T\mathbf{u}(\mathbf{x})$$

for the problems given by SARIDIS in [3] and for a multivariable half-quadratic formation. The number of steps according to (1) and the convergence probability calculated by the multiple runs for each problem were supplied by the described algorithm.

It is to be noted that on the basis of condition (11) an algorithm assigning a specified number of steps to the decrement Δy of one order of magnitude may also be prepared. The problem contains generally Δy decrements of 4—5 orders of magnitude resulting in $\lambda = 0.3-0.8$ on the basis of the step numbers calculated by (1) (for $p = 0.95$).

Summary

Some of the results of the study of non-asymptotically optimum procedures are reported. In this case the optimum of convergence rate is formulated by the author in [3]. This paper shows an unusual principle of study for cases of stochastic and real time.

References

1. CYPKIN, Y.: mrundlagen der Theorie lernender Systeme. VEB Verlag Technik. Berlin, 1972.
2. Monte Carlo Módszerek. Műszaki Könyvkiadó, 1965.
3. SARIDIS, G. N.: Learning applied to successive approximation algorithms. IEEE Trans. on System Science and Cyb. Vol. SSc-6. No. 2, Apr. 1970. pp. 97—103.
4. FREY, T.: Akadémiai Dr. Értekezés. 1970.

István BENCSIK, H-1521 Budapest