F.R.I. PROJECT RECORD

NO. 2827

EXPERIENCE WITH AUTOMATIC DIFFERENTIATION IN ESTIMATING FOREST GROWTH MODEL PARAMETERS

O. GARCIA

REPORT NO. 21 JUNE 1991

EXECUTIVE SUMMARY

The maximum-likelihood parameter estimation procedures used in FRI's regional growth models involve the numerical optimisation of complicated functions. This process can be costly and time-consuming, especially when dealing with large data sets and testing many variations in the model formulation. In 1988 we developed an automatic differentiation system, called GRAD, that made it possible to use analytical derivatives in the optimisation instead of the more expensive finite differences approximations. This resulted in considerable savings in time and money in the development of the KGM3 and PPM88 growth models on the VAX. It eventually made practical the parameter estimation on microcomputers.

This is one of two papers describing this work presented in January 1991 at a “Workshop on Automatic Differentiation of Algorithms”, in Breckenridge, Colorado, by invitation from the Society for Industrial and Applied Mathematics (SIAM). The second paper describes in greater detail the mathematical and computational techniques used in GRAD. Automatic Differentiation deals with the computer generation of routines for calculating function derivatives, and is becoming increasingly important in optimization, sensitivity analysis, numerical methods, and other applications in a variety of disciplines, including Physics, Chemistry, Meteorology, Oceanography, and Economics. It turned out that GRAD compares very favourably with previously available systems. This work was also presented at a “Workshop on Symbolic Computing in Applied Mathematics”, held in Christchurch in February 1991. The contents of these papers will be included in a book based on the proceedings of the SIAM Workshop.

Experience with Automatic Differentiation in estimating forest growth model parameters *

Oscar García
Forest Research Institute
Rotorua, New Zealand

Abstract

The performance of automatic differentiation in fitting growth models for intensively managed forest plantations is examined. The models consist of a system of stochastic differential equations, and parameters are estimated by maximum- likelihood using a general-purpose variable-metric optimization procedure. Compared to central difference approximations, the use of automatically generated derivatives in the optimization reduced computing time by a factor of 4 on a 80386/80387 microcomputer and by a factor of 6 on a MicroVAX 3500. An automatic differentiation procedure developed by the author was used, and found superior to JAKEF in this type of problem. The results may be relevant to estimation in other complex statistical models.

Introduction

Foresters can influence the development of a forest stand (a homogeneous patch of forest) through a number of silvicultural treatments. Stand density (trees per hectare), can be controlled by the selection of an initial planting density, and by thinnings, which are partial cuts where usually smaller and malformed trees are removed. Density affects the total volume production, the incidence of competition-induced mortality, and the size and timber quality of individual trees. Timber quality can also be improved by pruning lower branches, possibly at some cost in reduced growth. Other management decisions may involve the application of fertilizers and pesticides, different planting or regeneration techniques, the development and use of genetically improved seed, and the timing of the final cut. Mathematical growth models capable of predicting treatment effects are essential for rational forest management, especially in intensively managed production forests.

*Presented at the SIAM Workshop on Automatic Differentiation of Algorithms, Breckenridge, Colorado, January 6-8, 1991.

Over the past decade, 10 regional growth models for radiata pine and one for Douglas-fir have been developed in New Zealand using a methodology based on stochastic differential equations and maximum-likelihood parameter estimation [Garc88a]. These models can predict the behavior of stands subject to a wide range of initial densities, timing and intensity of thinnings, and in some instances pruning, fertilizing, and genetic improvement. Recently, automatic differentiation has been found useful in accelerating parameter estimation.

The nature of the models and estimation procedures is described briefly in the next section. More details can be found in [Garc79, Garc84, Garc89], and a more general discussion of growth modelling in [Garc88b]. [Bard74] is an excellent source for estimation theory and methods. Following this, results of comparisons between the use of difference approximations and of automatically generated analytic derivatives are presented, and the implications discussed.

These results may be relevant to the fitting of complex statistical models in other fields.

Models and estimation

The state of a forest stand is assumed to be adequately described by 3 to 5 state variables: mean diameter, stand height, trees per hectare, and, in some models, measures of ground cover and/or nutrient concentrations. Treatments cause instantaneous changes in the state variables. Between treatments, the state trajectories are modelled by a system of differential equations.

The differential equations are linear on power transformations of the state variables, and can be written as

$$\frac{d\mathbf{x}^{\mathbf{C}}}{dt} = \mathbf{A}\mathbf{x}^{\mathbf{C}} + \mathbf{b},$$

defining

$$\mathbf{x}^{\mathbf{C}} = \exp[\mathrm{C} \ln \mathrm{x}].$$

$\mathbf{x}$ is the state vector, and $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{C}$ are matrices and vectors of parameters to be estimated. Some of these are actually functions, containing unknown parameters, of a site productivity index specific to each stand. Some models include additional functions of state variables multiplying the right-hand-side [Garc89].

The data consists of a few consecutive measurements, at irregular intervals, on a large number of sample plots established in different stands. In order to devise a rational estimation procedure, the data variability is modelled as a perturbation of the differential equations by a Wiener stochastic process. The resulting stochastic differential equations can be integrated analytically to compute the likelihood function, that is, the probability of the model generating the observed data as a function of the parameters. The maximum- likelihood estimates are those values of the parameters that maximize the likelihood [Zack71, Bard74].

The differential equation parameters are estimated by maximum likelihood in two stages. The height growth is treated as a self-contained subsystem, since it can be assumed that the development in stand top height is approximately independent of the other state variables. Therefore, one of the equations involves only the heights, and its parameters are estimated first, together with the site index for each stand. Once these are available, the rest of the parameters are estimated using the likelihood function for the whole system. In addition to the Wiener perturbations, the height growth model also includes other random variables representing measurement errors.

For each data set, there are typically many parameter estimation runs, with variations of the basic model involving different numbers of state variables, the fixing at zero of various subsets of the parameters, the use of multiplier functions, and other detail changes. It is also advisable to repeat the procedure with different starting points to guard against the possibility of local optima.

Computational procedures

The fitting of the height growth model, although simpler than the full model in being univariate, involves the optimization of functions of hundreds of variables, i. e. the different site indices and sometimes nuisance parameters (variances) for each sample plot. A full Newton algorithm, with modifications to ensure convergence, has been implemented for this purpose [Garc83]. Although very large, the Hessian is sparse, with a special structure exploited by partitioning techniques. Analytic first and second derivatives are used, computed by a hand-coded implementation of the automatic differentiation approach discussed below. The procedure has proven to be very reliable and efficient.

For the rest of the parameters, the likelihood function is maximized (or rather, minus the logarithm of the likelihood is minimized) using a general- purpose variable-metric optimization routine. The smaller number of variables (9-20), and the experimental nature of the code, subject to frequent detail changes , did not justify the development of a specialized procedure as in the case of the height model. After some unsuccessful experiences with a difference approximation implementation of Fletcher and Powell's algorithm [Lill70], and with Nelder and Mead's simplex procedure ([ONei71], see [Garc79]), Hatfield Polytechnic's OPVM routine has performed well [Bigg71, Bigg73, NOC76].

OPVM is a Fortran subroutine using a quasi-Newton or variable-metric unconstrained optimization method. It can use analytic first derivatives, or approximate them with central differences in the auxiliary subroutine OPND1. Other more recent optimization routines have not been tested because they lack an essential feature of OPVM: if the function evaluation routine cannot compute the value at a given point, it can set a flag, and OPVM then reduces the step length and tries again. This is necessary because often trial points cause floating point exceptions, typically from out of range arguments in exponentials and other functions. In addition, some parameter values result in complex eigenvalues for the $/bfA$ matrix, unacceptable on physical grounds. The alternative of supplying step bounds is generally unsatisfactory.

The likelihood function to be optimized in these models is fairly complex. As much of the computation as possible is done in a pre-processing step, storing data transformations in an array. Still, the function evaluation subroutines called by the optimization procedure contain some 130 to 180 Fortran statements. This size, together with the expected modifications to the programs, made impractical the coding of analytic derivatives. Therefore, the OPND1 difference approximations were used.

Each function evaluation includes a loop over hundreds, or even thousands of observations. An approximation of the gradient by central differences requires a number of function evaluations equal to twice the number of variables (parameters). With the use of more complex model forms and larger data sets, the necessity of many lengthy over-night runs on mainframe computers (Digital VAX) increased costs and slowed down progress considerably. In 1988, an automatic differentiation procedure was developed in an attempt to improve computing turn-around.

The differentiation procedure, described in a companion paper [Garc91], takes as input a Fortran subroutine that computes the value of a function, and produces as output another Fortran subroutine that computes the derivatives with respect to specified variables. Using the powerful APL language it was possible to implement the system within a very short time. The use of analytic derivatives produced by these computer-generated subroutines greatly reduced computing times, and made it feasible to perform the parameter estimation on microcomputers [Garc89].

Tests and results

The use of analytic derivatives and difference approximations has been compared on three parameter estimation problems (Table 1). Problem A is one of the simpler models, with a moderate amount of data [Garc84]. Problem B is a more complex model with multiplier functions [Garc89], typical of those that motivated the automatic differentiation approach. The runs with A and B started from reasonable estimates, with those for A derived from the solution of B, and vice-versa. Problem C tested a more general form for the covariance of the Wiener process, with additional parameters to be estimated [Garc79, Garc84]. The starting point for C was the parameters estimated with the simpler covariance, and the optimization procedure stopped with only small change in the log-likelihood and failing the convergence test, showing the problem to be ill-conditioned (over- parametrized).

Table 1: Parameter estimation runs

Problem	A	B	C	B, VAX
Variables (parameters)	9	16	18	16
Observations	339	2093	1655	2093
Gradient subroutine generation (min)	6	10	14	-
Difference approximations				
Function calls	70	301	147	348
Gradient calls	45	184	32	210
Time (minutes)	10.8	890.7	89.9	614.8
Analytic derivatives				
Function calls	72	352	145	384
Gradient calls	45	211	33	221
Time (minutes)	2.7	228.5	21.8	99.3
Difference approximations run-time / analytic derivatives run-time	4.0	3.9	4.1	6.2

The computations were performed in double precision on an AT-compatible 20 MHz 80386 microcomputer with an 80387 coprocessor and Microsoft Fortran 4.10. Problem B was also run on a MicroVAX 3500 with VAX Fortran. The problems are reasonably well scaled, with variables and objective value not very far from one. The step size for the finite differences was set to 10^{-2} for runs A and B, and to 10^{-4} for C. Previous experience has shown that the step size is not critical, and that these values are satisfactory.

The third line in Table 1 shows the time taken by the automatic differentiation procedure. This does not include the minor manual editing required to change the subroutine header and assign the gradient. The finite differences run on problem B stopped without reducing all gradient components below the value of 10^{-3} specified in the convergence criterion, although it was very close. The speed-up factors from using analytic derivatives are given in the last row of Table 1. Some additional improvement could be obtained by modifying the optimization routine to make use of the function values computed by the analytic gradients subroutine.

There was no evidence of any improvement in reliability from using analytic derivatives, except perhaps for the difference in satisfying stringent convergence tests just mentioned. The optimization paths in problems A and C were very similar, but in problem B the difference approximations achieved larger reductions of the objective in fewer iterations (Figure 1).

Table 2 shows what Griewank ([Grie88]) calls the work ratio, the ratio of the time required to compute the (analytic) gradient to the time required to compute one function evaluation. Problem D is a model similar to A, but with more free parameters, and with the data set of C. The computations for D were done on a 12 MHz AT-compatible 80286 microcomputer, with the 80287

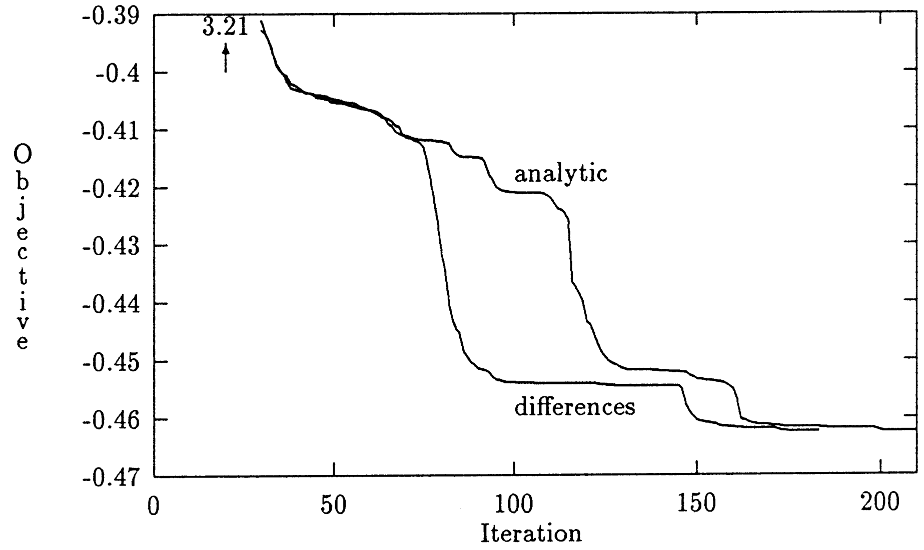

Figure 1: Optimization progress for problem B

Table 2: Ratio of gradient to function evaluation times (work ratio)

Problem	Variables	Observations	Work ratio
A	9	339	3.0
B	16	2093	5.6
C	18	1655	6.4
B, VAX	16	2093	3.7
D	18	1655	5.2
D, data subset	18	100	5.2
Same, JAKEF	18	100	8.0

coprocessor circuitry modified to run at 12 MHz, and Microsoft Fortran 4.10.

The author's automatic differentiation procedure uses what is essentially a forward approach [Grie88, Garc91]. Griewank proved that the reverse mode of differentiation is asymptotically superior to the forward form as the number of variables increases, at least if the possibility of exploiting sparsity is ignored. Therefore, a procedure based on the reverse mode, JAKEF [Hils90], was tried. Unfortunately, the reverse mode requires storage that grows with the number of statements executed in the function evaluation, which in these problems is large due to the loop over the observations. In the 640 K bytes of memory available in a microcomputer under DOS, JAKEF was unable to handle problems of practical size, with 200 or more observations. In addition, a test with a subset of 100 observations showed that the code generated by JAKEF was substantially slower (Table 2).

Discussion and conclusions

I have not tested alternatives to the direct maximization of the likelihood with a variable-metric algorithm. An attractive possibility are the extensions of Fisher's method of scoring [Zack71] discussed by Bard [Bard74] under the name of the Gauss Method. These provide an explicit approximation to the Hessian that can be used in the likelihood optimization. Although more difficult to implement, these techniques might be more efficient. The relative performance of analytic derivatives vs difference approximations, however, is likely to be similar. The same is true for alternative estimation criteria, for example, the maximization of the product of the likelihood and a prior distribution in Bayesian approaches.

The gains from Automatic Differentiation may differ with other optimization procedures. In particular, some implementations of difference approximations start using forward differences, and switch to central differences near the optimum. It is also possible that a derivative-free optimization algorithm, such as Brent's [Bren73] modification of Powell's method, might perform well in this application.

As mentioned before, the speed-up in the estimation of growth model parameters achieved through automatic differentiation was well worthwhile, reducing computing costs and improving turn-around when running on mainframes.

With the increased availability of inexpensive and powerful microcomputers, the preparatory work required by current automatic differentiation procedures is probably not warranted for the casual user, or for one-off or small optimization problems. For large problems that need to be solved repeatedly, however, automatic differentiation can be very useful.

The larger saving in computing time on VAX computers shown here agrees with our previous experience [Garc89]. A possible reason might be differences in the Fortran compilers optimization of the code generated by GRAD.

The forward mode of differentiation may be preferable to the reverse mode in this type of problem. The storage requirements of the reverse approaches can be prohibitive. The superior speed achieved with the author's forward procedure, compared to the reverse method implemented in JAKEF, can be attributed to the exploitation of sparsity, and to the production of stand- alone code, without calls to run-time routines or other overheads.

References

[Bard74] Bard, Y. *Nonlinear Parameter Estimation.* Academic Press, New York. 1974.

[Bigg71] Biggs, M.C. *Minimization algorithms making use of non-quadratic properties of the objective function.* **Journal of the Institute of Mathematics and its Applications 8**, 315-327. 1971.

[Bigg73] Biggs, M.C. *A note on minimization algorithms which make use of non-quadratic properties of the objective function.* **Journal of the Institute of Mathematics and its Applications 12**, 337-338. 1973.

[Bren73] Brent, R.P. *Algorithms for Minimization Without Derivatives.* Prentice-Hall, Englewood Cliffs, N.J. 1973.

[Garc79] García, O. *Modelling stand development with stochastic differential equations.* Pp. 315-333 *in* Elliot D.A. (*Ed.*) *Mensuration for Management Planning of Exotic Forest Plantations.* New Zealand Forest Service, FRI Symposium No. 20. 1979.

[Garc83] García, O. *A stochastic differential equation model for the height growth of forest stands. Biometrics 39*, 1059-1072. 1983.

[Garc84] García, O. *New class of growth models for even-aged stands: Pinus radiata in Golden Downs Forest.* **New Zealand Journal of Forestry Science 14**, 65-88. 1984.

[Garc88a] García, O. *Experience with an advanced growth modelling methodology.* Pp. 668-675 *in* Ek, A.R., Shifley, S.R., and Burk, T.E. (*Eds.*) *Forest Growth Modelling and Prediction.* USDA Forest Service, General Technical Report NC-120. 1988.

[Garc88b] García, O. *Growth modelling - A (re)view.* **New Zealand Forestry 33**(3), 14-17. 1988.

[Garc89] García, O. *Growth Modelling - New developments. In* Nàgumo, H. and Konohira, Y. (Ed.) *Japan and New Zealand Symposium* *on Forestry Management Planning.* Japan Association for Forestry Statistics. 1989.

[Garc91] García, O. *A system for the automatic differentiation of Fortran programs.* Presented at the SIAM Workshop on Automatic Differentiation of Algorithms, Breckenridge, Colorado, January 6-8, 1991.

[Grie88] Griewank, A. *On automatic differentiation.* Argonne National Laboratory, Mathematics and Computer Science Division. Preprint ANL/MCS-P10-1088. 1988.

[Hils90] Hilstrom, K.E. *User guide for JAKEF.* (Available in electronic form from *netlib* on *Internet*).

[Lill70] Lill, S.A. *Algorithm 46: A modified Davidon method for finding the minimum of a function using difference approximation for derivatives.* **The Computer Journal** **13**,111-113. 1970. (corrections in **14**,106).

[NOC76] N.O.C. *OPTIMA - Routines for optimisation problems.* Numerical Optimisation Centre, The Hatfield Polytechnic. 1976.

[ONei71] O'Neill, R. *Algorithm AS47 - Function minimization using a simplex procedure.* **Applied Statistics** **20**,338-346. 1971. (corrections and changes in **23**, 250-252 and **25**,97).

[Zack71] Zacks, S. *The Theory of Statistical Inference.* Wiley, New York. 1971.