Tatra Mt. Math. Publ. **39** (2008), 115–123

TATRA MOUNTAINS
Mathematical Publications

OPTIMAL EXPERIMENTAL DESIGN AND QUADRATIC OPTIMIZATION

Rebecca Haycroft — Luc Pronzato — Henry P. Wynn —
— Anatoly A. Zhigljavsky

ABSTRACT. A well known gradient-type algorithm for solving quadratic optimization problems is the method of Steepest Descent. Here the Steepest Descent algorithm is generalized to a broader family of gradient algorithms, where the step-length γ_k is chosen in accordance with a particular procedure. The asymptotic rate of convergence of this family is studied. To facilitate the investigation we re-write the algorithms in a normalized form which enables us to exploit a link with the theory of optimum experimental design.

Introduction

The steepest descent algorithm in $\mathbb{R}^d$ has been shown to be equivalent to a special algorithm for updating measures on the real line, see, e.g., [4]. The connection is that when the steepest descent algorithm is applied to the minimization of the quadratic function

$$f(x) = \frac{1}{2}(Ax, x) - (x, y)\,, \tag{1}$$

where (x, y) is the inner product, it can be translated to the updating of measures in $[m, M]$ where

$$m = \inf_{\|x\|=1} (Ax, x)\,, \qquad M = \sup_{\|x\|=1} (Ax, x)$$

with $0 < m < M < \infty$; m and M are the smallest and largest eigenvalues of A, respectively. The research has developed from the well known result, due to Akaike [1], that for standard steepest descent the renormalized iterates $\frac{x_k}{\sqrt{\|x_k\|}}$ converge to the two-dimensional space spanned by the eigenvectors corresponding to the eigenvalues m and M.

2000 Mathematics Subject Classification: Primary 62K05; Secondary 65K05, 37N40.
Keywords: gradient algorithms, steepest descent algorithm, rate of convergence, design of experiments, optimality criteria.

Let $g(x) = Ax - y$ be the gradient of the objective function (1). The steepest descent algorithm is $x_{k+1} = x_k - \frac{(g_k,g_k)}{(Ag_k,g_k)} g_k$. Using the notation $\gamma_k = \frac{(g_k,g_k)}{(Ag_k,g_k)}$, we write the algorithm as $x_{k+1} = x_k - \gamma_k g_k$. This can be rewritten in terms of the gradients as

$$g_{k+1} = g_k - \gamma_k A g_k \ . \tag{2}$$

The main objective of the paper is studying the family of algorithms (2) where the step-length γ_k is chosen in a general way. To facilitate this study we first rewrite the algorithm (2) in a different (normalized) form and then make a connection with the theory of optimum experimental design.

Renormalized version of gradient algorithms

Let us convert (2) into a "renormalized" version. First note that

$$(g_{k+1}, g_{k+1}) = (g_k, g_k) - 2\gamma_k (Ag_k, g_k) + \gamma_k^2 (A^2 g_k, g_k) \ . \tag{3}$$

Letting $r_k = \frac{(g_{k+1},g_{k+1})}{(g_k,g_k)}$ and dividing (3) through by (g_k, g_k) gives

$$r_k = 1 - 2\gamma_k \frac{(Ag_k, g_k)}{(g_k, g_k)} + \gamma_k^2 \frac{(A^2 g_k, g_k)}{(g_k, g_k)} \ . \tag{4}$$

The value of r_k can be considered as a rate of convergence of algorithm (2) at iteration k. Other rates which are asymptotically equivalent to r_k can be considered as well, see [4] for a discussion. The asymptotic rate of convergence of the gradient algorithm (2) can be defined as $R = \lim_{k\to\infty} (r_1 \cdot \ldots \cdot r_k)^{1/k}$. Of course, this rate may depend on the initial point x_0 or, equivalently, on g_0.

To simplify the notation, we need to convert to moments and measures. Since we assume that A is a positive definite d-dimensional square matrix, we can assume, without loss of generality, that A is a diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$; the elements $\lambda_1, \ldots, \lambda_d$ are the eigenvalues of the original matrix such that $0 < m = \lambda_1 \le \ldots \le \lambda_d = M$. Then for any vector $g = \big(g_{(1)}, \ldots, g_{(d)}\big)^T$ we can define

$$\mu_\alpha(g) = \frac{(A^\alpha g, g)}{(g, g)} = \frac{(\Lambda^\alpha g, g)}{(g, g)} = \frac{\sum_i g_{(i)}^2 \lambda_i^\alpha}{\sum_i g_{(i)}^2} \ .$$

This can be considered as the αth moment of a distribution with mass $p_i = \frac{g_{(i)}^2}{\sum_j g_{(j)}^2}$ at λ_i, $i = 1, \ldots, d$. Using the notation $\mu_\alpha^{(k)} = \mu_\alpha(g_k)$, where g_k are the iterates in (2), we can rewrite (4) as

$$r_k = 1 - 2\gamma_k \mu_1^{(k)} + \gamma_k^2 \mu_2^{(k)}. \tag{5}$$

For the steepest descent algorithm γ_k minimizes $f(x_k - \gamma g_k)$ over γ and we have $\gamma_k = \frac{1}{\mu_1^{(k)}}$ and $r_k = \frac{\mu_2^{(k)}}{\mu_1^{(k)2}} - 1$. Write $z_k = \frac{g_k}{\sqrt{(g_k, g_k)}}$ for the normalized gradient and recall that $p_i = \frac{g_{(i)}^2}{\sum_j g_{(j)}^2}$ is the ith probability corresponding to a vector g. The corresponding probabilities for the vectors g_k and g_{k+1} are

$$p_i^{(k)} = \frac{(g_k)_{(i)}^2}{(g_k, g_k)} \qquad \text{and} \qquad p_i^{(k+1)} = \frac{(g_{k+1})_{(i)}^2}{(g_{k+1}, g_{k+1})} \qquad \text{for } \ i = 1, \ldots, d\,.$$

Now we are able to write down the re-normalized version of (2), which is the updating formula for p_i $(i = 1, \ldots, d)$:

$$\begin{aligned} p_i^{(k+1)} &= \frac{(1-\gamma_k\lambda_i)^2}{(g_k, g_k) - 2\gamma_k(Ag_k, g_k) + \gamma_k^2(A^2 g_k, g_k)} p_i^{(k)} \\ &= \frac{(1-\gamma_k\lambda_i)^2}{1 - 2\gamma_k\mu_1^{(k)} + \gamma_k^2\mu_2^{(k)}} p_i^{(k)}. \qquad (6) \end{aligned}$$

When two eigenvalues of A are equal, say $\lambda_j = \lambda_{j+1}$, the updating rules for $p_j^{(k)}$ and $p_{j+1}^{(k)}$ are identical so that the analysis of the behaviour of the algorithm remains the same when $p_j^{(k)}$ and $p_{j+1}^{(k)}$ are confounded. We may thus assume that all eigenvalues of A are distinct.

A multiplicative algorithm for optimal design

Optimization in measure spaces covers a variety of areas and optimal experimental design theory is one of them. These areas often introduce algorithms which typically have two features: the measures are re-weighted in some way and the moments play an important role. Both features arise, as we have seen, in the above algorithms; see also [2], [3] and [5] for examples of other algorithms of this type.

In classical optimal design theory for the linear regression model $y_j = \alpha + \beta x_j + \varepsilon_j, x_j \in [m, M]$, one is interested in functionals of the moment matrix $M(\xi)$ of a design measure ξ:

$$M(\xi) = \begin{pmatrix} \mu_0 & \mu_1 \\ \mu_1 & \mu_2 \end{pmatrix}, \qquad (7)$$

where $\mu_\alpha = \mu_\alpha(\xi) = \int x^\alpha \,\mathrm{d}\xi(x)$ are the αth moments of the measure ξ and $\mu_0 = 1$.

In the theory of optimum design the directional (Fréchet) derivative "towards" a discrete measure ξ_x mass 1 at a point x is of importance. This is

$$\frac{\partial}{\partial\alpha}\Phi\left(M\big[(1-\alpha)\xi+\alpha\xi_x\big]\right)\Big|_{\alpha=0}=\operatorname{tr}\Big(\overset{\circ}{\Phi}(\xi)M(\xi_x)\Big)-\operatorname{tr}\Big(\overset{\circ}{\Phi}(\xi)M(\xi)\Big), \qquad (8)$$

where

$$\overset{\circ}{\Phi}(\xi)=\frac{\partial\Phi}{\partial M}\Big|_{M=M(\xi)}=\begin{pmatrix}\frac{\partial\Phi}{\partial\mu_0} & \frac{1}{2}\frac{\partial\Phi}{\partial\mu_1}\\ \frac{1}{2}\frac{\partial\Phi}{\partial\mu_1} & \frac{\partial\Phi}{\partial\mu_2}\end{pmatrix}.$$

Here Φ is a functional on the space of 2×2 matrices usually considered as an optimality criterion to be maximized with respect to ξ. The first term on the right hand side of (8) is

$$\varphi(x,\xi)=\operatorname{tr}\Big(\overset{\circ}{\Phi}(\xi)M(\xi_x)\Big)=\begin{pmatrix}1, & x\end{pmatrix}\overset{\circ}{\Phi}(\xi)\begin{pmatrix}1\\ x\end{pmatrix}=\frac{\partial\Phi}{\partial\mu_0}+x\frac{\partial\Phi}{\partial\mu_1}+x^2\frac{\partial\Phi}{\partial\mu_2}\,.$$

A class of optimal design algorithms is based on the multiplicative updating of the weights of the current design measure $\xi^{(k)}$ with some function of $\varphi(x,\xi)$. We show below how algorithms in this class are related to the gradient algorithms (2) in their re-normalized form (6).

Assume that our measure is discrete and concentrated on $[m,M]$. Assume also that $\frac{\partial\Phi(M)}{\partial\mu_2}>0$; then $\varphi(x,\xi)$ has a well-defined minimum

$$c(\xi)=\min_{x\in\mathbb{R}}\varphi(x,\xi)=\frac{\partial\Phi}{\partial\mu_0}-B(\xi)\,,\quad\text{where}\quad B(\xi)=\frac{1}{4}\frac{\left(\frac{\partial\Phi}{\partial\mu_1}\right)^2}{\left(\frac{\partial\Phi}{\partial\mu_2}\right)}\,.$$

Let $\xi(x)$ be the mass at a point x and define the re-weighting at x by

$$\xi^{'}(x)=\frac{\varphi(x,\xi)-c(\xi)}{b(\xi)}\,\xi(x)\,, \qquad (9)$$

where $b(\xi)$ is a normalizing constant

$$b(\xi)=\int_m^M\big(\varphi(x,\xi)-c(\xi)\big)\xi(\mathrm{d}x)=\operatorname{tr}\Big[M(\xi)\,\overset{\circ}{\Phi}(\xi)\Big]-c(\xi)\,.$$

Let us define $\gamma=\gamma(\xi)=\gamma(\mu_1,\mu_2)$ as

$$\gamma=\gamma(\xi)=\frac{-2\frac{\partial\Phi}{\partial\mu_2}}{\frac{\partial\Phi}{\partial\mu_1}}\,. \qquad (10)$$

Then

$$\varphi(x,\xi)-c(\xi)=B(\xi)\left(1-\gamma(\xi)x\right)^2.$$

The normalization ensures that the measure $\xi^{'}$ is a probability distribution. We obtain that the re-weighting formula (9) can be equivalently written as

$$\xi^{'}(x) = \frac{(1-\gamma x)^2}{1-2\gamma\mu_1+\gamma^2\mu_2}\,\xi(x)\,. \tag{11}$$

This is exactly the same as the general gradient algorithm in its renormalized form (6). To see that, we simply write the updating formula (11) iteratively

$$\xi^{(k+1)}(x) = \frac{(1-\gamma_k x)^2}{1-2\gamma_k\mu_1^{(k)}+\gamma_k^2\mu_2^{(k)}}\,\xi^{(k)}(x)\,.$$

Optimum design gives the worst rate of convergence

Let $\Phi = \Phi\big(M(\xi)\big)$ be an optimality criterion, where $M(\xi)$ is as in (7). Associate with it a gradient algorithm with step-length $\gamma(\mu_1,\mu_2)$ as given by (10).

Let ξ^* be the optimum design for Φ on $[m,M]$; that is,

$$\Phi\big(M(\xi^*)\big) = \max_{\xi}\Phi\big(M(\xi)\big)$$

where the maximum is taken over all probability measures supported on $[m,M]$. Note that ξ^* is invariant for one iteration of the algorithm (11); that is, if $\xi=\xi^*$ in (11) then $\xi'(x)=\xi(x)$ for all $x\in\operatorname{supp}(\xi)$.

In accordance with (5), the rate associated with the design measure ξ is defined by

$$r(\xi) = 1-2\gamma\mu_1+\gamma^2\mu_2 = \frac{b(\xi)}{B(\xi)}\,.$$

Assume that the optimality criterion Φ is such that the optimum design ξ^* is non-degenerate (that is, ξ^* is not just supported at a single point). Note that if $\Phi(M)=-\infty$ for any singular matrix M, then this condition is satisfied.

Since the design ξ^* is optimum, all directional derivatives are non-positive:

$$\frac{\partial}{\partial\alpha}\,\Phi\Big[M\big((1-\alpha)\xi^*+\alpha\xi(x)\big)\Big]\Big|_{\alpha=0^+}\le 0\,,$$

for all $x\in[m,M]$. Using (8), this implies

$$\max_{x\in[m,M]}\varphi(x,\xi^*)\le t^* = \operatorname{tr}\Big[M(\xi^*)\overset{\circ}{\Phi}(\xi^*)\Big]\,.$$

Since $\varphi(x,\xi^*)$ is a quadratic convex function of x, this is equivalent to $\varphi(m,\xi^*)$ $\le t^*$ and $\varphi(M,\xi^*)\le t^*$. As

$$\int_m^M \varphi(x,\xi^*)\xi^*(\mathrm{d}x) = t^*$$

this implies that ξ^* is supported at m and M. Since ξ^* is non-degenerate, ξ^* has positive masses at both points m and M and

$$\varphi(m,\xi^*) = \varphi(M,\xi^*) = t^*.$$

As $\varphi(x,\xi^*)$ is quadratic in x with its minimum at $\frac{1}{\gamma}$, this implies that

$$\gamma^* = \gamma\big(\mu_1(\xi^*),\, \mu_2(\xi^*)\big) = \frac{2}{(m+M)}\,.$$

The rate $v(\xi^*)$ is therefore

$$v(\xi^*) = \frac{b(\xi^*)}{B(\xi^*)} = \frac{t^* - c(\xi^*)}{B(\xi^*)} = (1-m\gamma^*)^2 = (1-M\gamma^*)^2 = R_{\max}\,,$$

where

$$R_{\max} = \frac{(M-m)^2}{(M+m)^2}\,. \tag{12}$$

Assume now that the optimum design ξ^* is degenerate and is supported at a single point x^*. Note that since $\varphi(x,\xi^*)$ is both quadratic and convex, x^* is either m or M. Since the optimum design is invariant in one iteration of the algorithm (11), γ^* is constant and

$$\max_{\xi} r(\xi) = \max\big[(1-m\gamma^*)^2, (1-M\gamma^*)^2\big] \ge R_{\max}$$

with the inequality replaced by an equality if and only if $\gamma^* = \frac{2}{(M+m)}$.

Some special cases

A few examples of gradient algorithms (2) worthy of mention are given below.

The steepest descent algorithm corresponds to the case when $\Phi(\xi)$ is the D-optimality criterion $\Phi\big(M(\xi)\big) = \mu_2 - \mu_1^2$ with a step length equal to $\gamma_k = \frac{1}{\mu_1^{(k)}}$. It is well-known that the asymptotic rate of the steepest descent algorithm is always close to the value $R_{\max}$ defined in (12). The asymptotic behaviour of the steepest descent algorithm has already been extensively studied, see, e.g., [4].

The steepest descent algorithm with relaxation is also known in literature on optimization. For this algorithm, $\gamma_k = \frac{\varepsilon}{\mu_1^{(k)}}$, where ε is some fixed positive number. This algorithm can be associated with the optimality criterion

$$\Phi\big(M(\xi)\big) = \varepsilon\mu_2 - \mu_1^2\,. \tag{13}$$

It is known that for suitable values of the relaxation parameter ε this algorithm has a faster convergence rate than the ordinary steepest descent algorithm. However, the reasons why this occurs were not previously known.

It can be shown that if the relaxation coefficient ε is either small ($\varepsilon < \frac{4Mm}{(M+m)^2}$) or large ($\varepsilon > 1$), then for almost all starting points the algorithm asymptotically behaves as if it has started at the worst possible initial point. Equivalently, for these values of ε the sequence of designs converges to the optimal design for the criterion (13).

Moreover, if the relaxation parameter is either too small ($\varepsilon < \frac{2m}{m+M}$) or too large ($\varepsilon > \frac{2M}{m+M}$), then the rate of the steepest descent algorithm with relaxation becomes worse than $R_{\max}$, the worst-case rate of the standard steepest descent algorithm. This is related to the fact that for these values of ε the optimal design for the criterion (13) is degenerate (that is, it is concentrated at a single point). As a consequence, we also obtain a well-known result that if the value of the relaxation coefficient is either $\varepsilon < 0$ or $\varepsilon > 2$, then the steepest descent algorithm with relaxation diverges.

When $\frac{4Mm}{(m+M)^2} < \varepsilon \leq 1$ the relaxed steepest descent algorithm does not converge to the optimum design and its renormalized version (6) asymptotically exhibits either cyclic or chaotic behaviour. It is within this range of ε that improved asymptotic rates of convergence are observed. The behaviour of the asymptotic rate R is shown in Fig. 1, where we display the asymptotic rates in the case $\frac{M}{m} = 10$. In this figure we assume that $d = 100$ and all the eigenvalues are equally spaced. We have established numerically that the dependence on the dimension d is insignificant as long as $d \geq 10$. In addition, choosing equally spaced

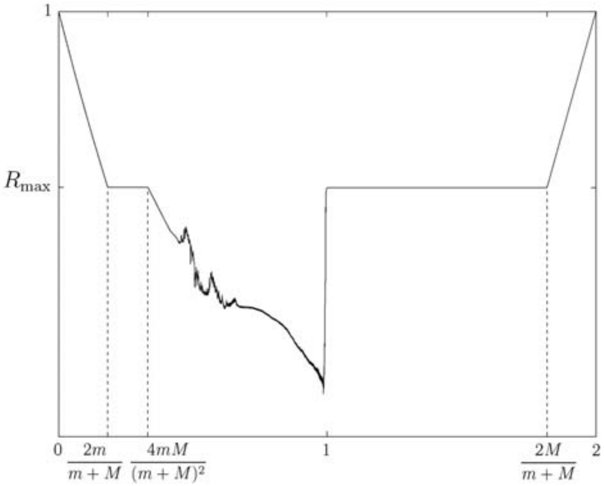

FIGURE 1. Asymptotic rate of convergence as a function of ε for the steepest descent algorithm with relaxation ε.

eigenvalues is effectively the same as choosing eigenvalues uniformly distributed on $[m, M]$ and taking expected values of the asymptotic rates.

The convergence rates of all gradient-type algorithms depend on, amongst other things, the condition number $\rho = \frac{M}{m}$. As one would expect, an increase in ρ gives rise to a worsening rate of convergence. Fig. 2 shows the effect of increasing the value of ρ on the rates of convergence for the steepest descent algorithm with relaxation coefficients $\varepsilon = 0.9$ and 0.99.

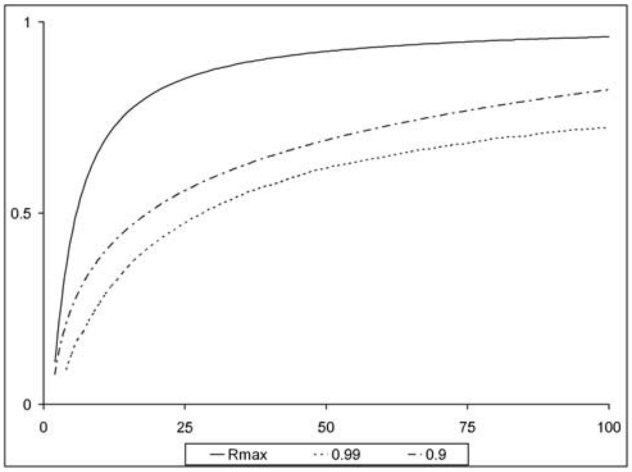

FIGURE 2. Asymptotic rate of convergence as a function of ρ for steepest descent with relaxation coefficients $\varepsilon = 0.9$ and $\varepsilon = 0.99$.

Any optimization criterion $\Phi\big(M(\xi)\big)$ such that $\frac{\partial\Phi}{\partial\mu_2} > 0$ creates an optimization algorithm of the form (2). Some of these algorithms can be very efficient. For example the family of Φ_p-optimality criteria $\Phi_p\big(M(\xi)\big) = \big(\mathrm{tr}M^{-p}(\xi)\big)^{\frac{1}{p}}$ creates very efficient optimization algorithms. Another useful generalization of the steepest descent algorithm is the family of so-called α-root algorithms related to the criteria $\Phi\big(M(\xi)\big) = \mu_2^{\alpha} - \mu_1^{2\alpha}$. For values of α slightly larger than 1 the resulting optimization algorithms have been found to be extremely efficient.

REFERENCES

[1] AKAIKE, H.: *On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method*, Ann. Inst. Statist. Math. Tokyo **11** (1959), 1–16.

[2] MANDAL, S.—TORSNEY, B.: *Construction of optimal designs using a clustering approach*, J. Statist. Plann. Inference **136** (2006), 1120–1134.

[3] MANDAL, S.—TORSNEY, B.—CARRIERE, K. C.: *Constructing optimal designs with constraints*, J. Statist. Plann. Inference **128** (2005), 609–621.

[4] PRONZATO, L.—WYNN, H. P.—ZHIGLJAVSKY, A. A.: *Dynamical Search*, Chapman & Hall/CRC, Boca Raton, 2000.

[5] TORSNEY, B.—MANDAL, S.: *Multiplicative algorithms for constructing optimizing distributions: further developments*. In: Proceedings of the 7th International Workshop on Model-Oriented Design and Analysis—MODA '04 (A. Di Bucchianico et al., eds.), Heeze, The Netherlands, Physica-Verlag, Heidelberg, 2004, pp. 163–171.

Received September 29, 2006

Rebecca Haycroft
Anatoly A. Zhigljavsky
Cardiff University
School of Mathematics
Senghennydd Road
Cardiff CF24 4AG
UNITED KINGDOM
E-mail: haycroftrj@cf.ac.uk
zhigljavskyaa@cf.ac.uk

Luc Pronzato
Laboratoire I3S
2000 route des Lucioles B.P. 121
F-06903 Sophia Antipolis
FRANCE
E-mail: pronzato@i3s.unice.fr

Henry P. Wynn
London School of Economics
Dept. of Statistics
London WC2A 2AE
UNITED KINGDOM
E-mail: h.p.wynn@lse.ac.uk