A Minimization Algorithm for a Class of Functions

L. R. Welch[1]
Communications Systems Research Section

Let N be a positive integer and $A_0, \cdots, A_N$ be non-negative numbers with at least one positive. Define

$$G(x) = \frac{N}{x} + \sum_{k=0}^{N} A_k x^k$$

The problem is to compute $z > 0$ with

$$G(z) = \min_{x>0} G(x)$$

This article gives a simple algorithm requiring $\lceil (3/2)(\ln N/\ln 2) \rceil + 8$ evaluations of a polynomial of degree $N+1$ and 6 evaluations of its derivative. This algorithm is required to optimize the DSN resource allocation process.

I. Introduction

Let N be a positive integer and $A_0, \cdots, A_N$ be non-negative numbers with at least one positive. Define

$$G(x) = \frac{N}{x} + \sum_{k=0}^{N} A_k x^k$$

The problem is to compute $z > 0$ with

$$G(z) = \min_{x>0} G(x)$$

This article gives a simple algorithm requiring $\lceil (3/2) \times (\ln N/\ln 2) \rceil + 8$ evaluations of a polynomial of degree $N+1$ and 6 evaluations of its derivative. This algorithm is required to optimize the DSN resource allocation process.

II. Analysis

The derivative of G is

$$G'(x) = \frac{-N}{x^2} + \sum_{k=1}^{N} A_k k x^{k-1}$$

$$= \frac{-N + \sum_{k=1}^{N} A_k k x^{k+1}}{x^2}$$

[1]Consultant from USC Electrical Engineering Department.

Let $B_0 = -N$, $B_1 = 0$, $B_k = (k-1)A_{k-1}$ for $k \geqq 2$, and

$$f(x) = \sum_{k=0}^{N+1} B_k x^k$$

Then $f(z) = 0$ and the problem becomes one of finding the roots of $f(x) = 0, x > 0$.

III. The Algorithm

Step I: Define

$$\sup = \exp\left(\min_{\substack{2 \le k \le N+1 \\ B_k \neq 0}} \left[\frac{1}{k} \ln \frac{N}{B_k}\right]\right)$$

$$\text{GLB} = \exp\left(\min_{\substack{2 \le k \le N+1 \\ B_k \neq 0}} \left[\frac{1}{k} \ln \frac{1}{B_k}\right]\right)$$

$$L = \left\lceil \frac{3}{2}\left(\frac{\ln N}{\ln 2}\right) \right\rceil + 2$$

Step II:

Let $x = \dfrac{\sup + \text{GLB}}{2}$.

If $f(x) \geqq 0$, set $\sup = x$.

If $f(x) < 0$, set $\text{GLB} = x$.

Do this calculation L times.

Set $x = \sup$.

Step III (Newton's method):

Replace x by $x - \dfrac{f(x)}{f'(x)}$.

Do this calculation six times.

Set $z = x$.

IV. Results

The relative error in z is less than 10^{-40} if the arithmetic is executed with infinite precision, and the relative error in $G(z)$ will be smaller. If the arithmetic is executed with s decimal digits of accuracy, the relative error will be on the order of $N^2 10^{-s}$, $s < 40$.

V. Validity of the Algorithm

Since $f(0) = -N$ and $f'(x), f''(x)$ are positive for $x > 0$, there is a unique root on the positive axis. The solution can be bounded as follows: Define

$$\sup = \exp\left(\min_{\substack{2 \leqq k \leqq n \\ B_k \neq 0}} \left[\frac{1}{k} \ln \frac{N}{B_k}\right]\right) = \left(\frac{N}{B_{k_0}}\right)^{1/k_0}$$

$$\text{GLB} = \exp\left(\min_{\substack{2 \leqq k \leqq n \\ B_k \neq 0}} \left[\frac{1}{k} \ln \frac{1}{B_k}\right]\right) = \left(\frac{1}{B_{k_1}}\right)^{1/k_1}$$

Then the following statements are true:

(A) $\text{GLB} \leqq z \leqq \sup$

Proof: If $0 \leqq z < \text{GLB}$, then

$$f(z) = -N + \sum_{k=2}^{N+1} B_k z^k < -N + \sum_{k=2}^{N+1} B_k \cdot \frac{1}{B_k} \leqq 0$$

Thus, $f(z) < 0$, a contradiction. If $z > \sup$, then

$$f(z) = -N + \sum_{k=2}^{N+1} B_k z^k > -N + B_{k_0} \cdot \frac{N}{B_{k_0}} = 0$$

Thus, $f(z) > 0$, a contradiction.

(B) $\dfrac{\sup - \text{GLB}}{z} < N^{1/2}$

Proof:

$$\sup = \left(\frac{N}{B_{k_0}}\right)^{1/k_0} \leqq \left(\frac{N}{B_{k_1}}\right)^{1/k_1} = (N)^{1/k_1}\,\text{GLB}$$

Now

$$\frac{\sup - \text{GLB}}{z} < \frac{\sup}{\text{GLB}} \leqq N^{1/k_1} \leqq N^{1/2}$$

(C) After step II of the algorithm,

$$\frac{\sup - \text{GLB}}{z} < \frac{1}{2N}$$

Proof: Each step of the algorithm divides ($\sup - \text{GLB}$) by two. Therefore,

$$\frac{\sup - \text{GLB}}{z} < N^{1/2}\, 2^{-L} < N^{1/2}\, 2^{-((3/2)(\ln N/\ln 2)+1)} = \frac{1}{2N}$$

(D) Since $f'(x)$ and $f''(x)$ are positive for $x > 0$, if $x_0, x_1, \cdots$ are computed using Newton's method beginning with $x_0 > z$, the x's form a monotone sequence decreasing to z.

(E) $|x_t - z| < \dfrac{2z}{N}\left(\dfrac{1}{2}\right)^{2^t}$

Proof:

$$x_{t+1} - z = x_t - z - \frac{f(x_t)}{f'(x_t)} = \frac{(x_t - z) f'(x_t) - f(x_t)}{f'(x_t)}$$

Using Taylor's theorem with the remainder for the numerator,

$$|x_{t+1} - z| = \frac{\dfrac{(x_t - z)^2}{2}\left[f''(\epsilon) + (\epsilon - z) f^{(3)}(\epsilon)\right]}{f'(x_t)}$$
$$< \frac{(x_t - z)^2}{2} \frac{\left[f''(x_t) + (x_t - z) f^{(3)}(x_t)\right]}{f'(x_t)}$$

However, for polynomials of degree N or less with all positive coefficients, $zh'(z) \leqq Nh(z)$. This gives

$$(x_{t+1} - z) < \frac{(x_t - z)^2}{2} \frac{N}{z}\left(1 + \frac{N}{z}(x_t - z)\right)$$

Use of $(x_0 - z) < (z/N) \cdot \frac{1}{2}$ and the above inequality produces the statement (E).