DEMONSTRATIO MATHEMATICA
Vol. XI No 3 1978

Jerzy Trojan

HOW TO DECREASE THE COMBINATORY COMPLEXITY

Introduction

We consider the solution of a nonlinear scalar equation

$$f(x) = 0, \tag{1}$$

where $f: D \subset \mathbb{C} \to \mathbb{C}$. Assume that f is analytic in a neighbourhood of a simple zero α, $f(\alpha) = 0 \neq f'(\alpha)$. We approximate α by an iteration. Suppose that we can compute the standard information on f, i.e.

$$\mathfrak{N}(f,x) = \left\{f(x), f'(x), \ldots, f^{(s)}(x)\right\} \tag{2}$$

for a given integer s, $s \geqslant 1$. We deal with a one-point iteration without memory φ which defines the sequence of successive approximations $\{x_i\}$ by

$$x_{n+1} = \varphi(x_n, \mathfrak{N}(f,x_n)), \quad n = 0,1,\ldots \tag{3}$$

where x_0 is a given initial approximation, see [2].

The measure of "goodness" of the iteration (3) is defined by the complexity index [3]

$$z(\varphi) = \frac{\mathrm{comp}(\mathfrak{N}) + \mathrm{comb}(\varphi)}{\log_2 p(\varphi)}, \tag{4}$$

where $\mathrm{comp}(\mathfrak{N})$ is the complexity (cost) of computing $\mathfrak{N}(f,x)$, $\mathrm{comb}(\varphi)$ is the combinatory complexity of combining the information $\mathfrak{N}(f,x_n)$ to produce the next approximation x_{n+1} and $p(\varphi)$ is the order of the iterations φ.

From the computational complexity point of view we want to find an iteration φ with minimal complexity index. Let

$$(5)\qquad z(\Phi_s) = \inf_{\varphi\in\Phi_s} z(\varphi),$$

where Φ_s is the class of iterations which use the standard information (2). It is well-known [1], [2] that $p(\varphi) \leqslant s+1$ for any φ from Φ_s. Since any φ has to use $x_n, f(x_n),\ldots, f^{(s)}(x_n)$ at least once at the n-th step, taking the complexity as the total number of arithmetic operations we have

$$(6)\qquad \mathrm{comb}(\varphi) \geqslant s.$$

We recall the interpolatory iteration I_s which was defined in [4].

Algorithm 1. Given x_n, set $z_0 = x_n$; perform $m = \lceil \log_2(s+1)\rceil$ Newton steps for the interpolatory polynomial

$$P_s(x) = f(x_n) + f'(x_n)(x-x_n) + \ldots + \frac{1}{s!} f^{(s)}(x_n)(x-x_n)^s$$

as follows:

for $i = 0(1)m-1$ compute $P_s(z_i)$, $P_s'(z_i)$ and $z_{i+1} = z_i - P_s'(z_i)^{-1}\cdot P_s(z_i)$;

set $x_{n+1} = z_m$.

It is easy to verify that the combinatory complexity of I_s is roughly $4(s+1)\log_2(s+1)$. From this we get the estimate [3]

$$(7)\qquad \frac{\mathrm{comp}(\mathfrak{N}) + s}{\log_2(s+1)} \leqslant z(\Phi_s) \leqslant \frac{\mathrm{comp}(\mathfrak{N}) + 4(s+1)\log_2(s+1)}{\log_2(s+1)}.$$

If $\mathrm{comp}(\mathfrak{N}) \gg s\log_2 s$ then we have a pretty tight bounds on $z(\Phi_s)$. However for large s it can happen that $\mathrm{comp}(\mathfrak{N})$

linearly depends on s and the combinatory complexity might dominate. Thus, we wish to answer the following questions:

Does there exist an iteration whose combinatory complexity is essentially less than $O(s\log_2 s)$?

What is the minimal combinatory complexity of an iteration of order $s+1$?

In the next section we exhibit an iteration φ^* of order $s+1$ with linear combinatory complexity, i.e. $\mathrm{comb}(\varphi^*) = O(s)$. Due to (6) this means that φ^* has minimal complexity (exact to asymptotic constant) which answers our problem.

Minimal combinatory complexity

Assume that $s+1 = 2^{m-1} + k$, $0 < k \leq 2^{m-1}$. Define a sequence of numbers $\{\mu_i\}$, $\mu_i = 2^{i+1}-1$, $i = 0(1)m-1$. Let $P_{\mu_i}(x)$ be a polynomial of degree $\leq \mu_i$ defined by

$$P_{\mu_i}(x) = b_0 + b_1(x-x_n) + \dots + b_{\mu_i}(x-x_n)^{\mu_i},$$

where $b_j = \frac{1}{j!} f^{(j)}(x_n)$ $j = 0(1)s$, $b_j = 0$ for $j > s$.

Assume that x_n is a sufficiently close approximation to α. Then it is obvious that $P'_{\mu_i}(x) \neq 0$ for x close to α and there exists a point α_i for which $P_{\mu_i}(\alpha_i) = 0$ and

$$\alpha - \alpha_i = O((x_n - \alpha)^{\mu_i+1}) = O(e^{2^{i+1}}), \quad (8)$$

where $e = x_n - \alpha$, for $i = 1(1)m-2$.
For $i = m-1$ we have $\alpha_{m-1} - \alpha = O(e^{s+1})$.

We are now in position to propose the following iteration φ^* defined by the algorithm

A l g o r i t h m 2. Given x_n, set $z_0 = x_n$; for $i = 0(1)m-1$ compute $P_{\mu_i}(z_i)$, $P_{\mu_i}(z_i)$;

$$z_{i+1} = z_i - P_{\mu_i}(z_i)^{-1} \cdot P_{\mu_i}(z_i);$$

set $x_{n+1} = z_m$.

Comparing with Algorithm 1 we see that we still have m Newton steps but applied to the polynomials P_1, P_3, $P_7, \ldots,$ $P_{\mu_{m-1}}$ of increasing degrees.

T h e o r e m . For the iteration φ^* we have

$$p(\varphi^*) = s + 1,$$

$$\operatorname{comb}(\varphi^*) = 2^{m+2} + 4s - 4m - 6 \leqslant 12s - 4\log_2(s+1) - 6.$$

P r o o f. Let $e_i = z_i - \alpha$ and $d_i = \alpha_i - \alpha$. From (8) we get $e_1 = z_1 - \alpha = z_1 - \alpha_0 + \alpha_0 - \alpha = \alpha_0 - \alpha = d_0 = O(e^2)$ since P_1 is a polynomial of first degree and $z_1 = \alpha_0$. Assume by induction that $e_j = O\left(e^{2^j}\right)$ for $j \leqslant i$. Then

$$e_{i+1} = z_{i+1} - \alpha = z_{i+1} - \alpha_i + \alpha_i - \alpha = O(z_i - \alpha_i)^2 + d_i =$$

$$= O(z_i - \alpha + \alpha - \alpha_i)^2 + d_i =$$

$$= O\left(O\left(e^{2^i}\right) + O\left(e^{2^{i+1}}\right)\right)^2 + O\left(e^{2^{i+1}}\right) =$$

$$= O\left(e^{2^i}\right)^2 + O\left(e^{2^{i+1}}\right) = O\left(e^{2^{i+1}}\right) \quad \text{for} \quad i = 0(1)m-2.$$

For $i = m - 1$ we have

$$e_m = z_m - \alpha = z_m - \alpha_{m-1} + \alpha_{m-1} - \alpha =$$

$$= O(e^{2^m}) + O(e^{s+1}) = O(e^{s+1})$$

which means that $x_{n+1} - \alpha = O(x_n - \alpha)^{s+1}$. Hence $p(\varphi^*) = s + 1$.

We compute the combinatory complexity of φ^*. We apply m steps of Newton iteration. In every step except the first one we compute the values P_{μ_i} and P'_{μ_i} by Horner's scheme performing $4\mu_i - 2$ multiplications and additions. Newton's iteration itself needs 2 arithmetic operations. Thus the total number of operations is equal to

$$\mathrm{comb}(\varphi^*) = 2 + \sum_{i=1}^{m-2} 4\mu_i + 4s = 2^{m+2} + 4s - 4m - 6.$$

Since $s+1 = 2^{m-1} + k, \quad 0 < k \leqslant 2^{m-1}$ we can estimate

$\mathrm{comb}(\varphi^*) \leqslant 12s - 4\log_2(s+1) - 6$

and this completes the proof.

Applying our theorem we can improve the estimate (7), namely

$$(9) \qquad \frac{\mathrm{comp}(\mathfrak{N}) + s}{\log_2(s+1)} \leqslant z(\Phi_s) \leqslant \frac{\mathrm{comp}(\mathfrak{N}) + 12s}{\log_2(s+1)} .$$

Note that both sides of (9) are very tight for every s.

The presented algorithm can be used also for the multivariate case. This will be reported in another paper.

REFERENCES

[1] M. Jankowski, H. Woźniakowski: O złożoności obliczeniowej w analizie numerycznej, Matematyka Stosowana 5 (1975) 5-27.

[2] J.F. Traub: Iterative methods for the solution of equations, New York 1964.

[3] J.F. Traub, H. Woźniakowski: Strict lower and upper bounds on iterative computational complexity, in Analytic computational complexity, New York (1976) 15-34.

[4] H. Woźniakowski: Maximal order of multipoint iterations using n evaluations, in Analytic computational complexity, New York (1976) 75-109.

INSTITUTE OF MATHEMATICS, SILESIAN TECHNICAL UNIVERSITY, GLIWICE

Received November 16, 1977.