ON APPROXIMATE METHODS OF TANGENT HYPERBOLAS [1]

I. KALDO[1] and O. VAARMANN[2]

[1] *Estonian Business School*

Lauteri 3, Tallinn, Estonia

E-mail: indrek.kaldo@ebs.ee

[2] *Tallinn Technical University*

Ehitajate tee 5, Tallinn, Estonia

E-mail: vaarmann@edu.ttu.ee

ABSTRACT

For solving a nonlinear operator equation in Banach space setting approximate variants of the method of tangent hyperbolas are considered. This family of approximate methods includes as special cases methods based on the use of iterative methods to obtain a cheap solution of limited accuracy for associated linear equations at each iteration step as well. A local convergence theorem and rate of convergence for the methods under discussion are given. Computational aspects and possibilities of organizing parallel computation are discussed. Computational experience with various multiprocessors indicates that performance of parallel methods depends critically on efficient load balancing. Problems of allocating subproblems to the processors are also briefly discussed.

Key words: nonlinear equations, Banach spaces, methods with the high order of convergence, approximate variants of methods, parallel computation

1. INTRODUCTION AND BASIC THEOREM

Many real-life problems can be modeled in terms of nonlinear equations

$$F(x) = 0, \tag{1.1}$$

where F is an operator from a Banach space X into another Banach space Y and it is as many times as necessary differentiable.

The use of high order iterative methods for solving (1.1) sometimes enables more rapidly and accurately to calculate values of the model than those with a lower rate of convergence. Computational effort is often one of the basic problems in the solution of real-life problems. The total cost of an iterative method is determined by the number of iterations needed to achieve the required accuracy and the cost of each iteration. For computing a solution with the prescribed accuracy implementation of methods with the high order of convergence require, as a rule, less iterations than methods with a lower convergence order and therefore, likely, require less total arithmetics. Methods with the high order of convergence making full use of the local information (e.g. functional values gradient and Hessian) permit sometimes to win in speed and accuracy. Frequently, only functional values are available and their evaluation needs much computational work, e.g. the solution of inverse problems, the solution of two-point boundary value problems in differential equations by shooting method, computation of proper values for coordination parameters in decomposition-coordination schemes in convex programming. There are also many industrial problems in which the model code requires much computation time and thus must not be called too many times. To save solving of laborious subproblems the implementation of rapidly convergent methods may be useful.

Another important aspect of computation is stability. Whereas even very rough approximation to the operator of second derivatives in the method with the convergence $p \geq 3$ may provide their numerical stability [4; 10] then it is reasonable to develop methods based on a quadratic model

$$F(x_e + d) = F(x_e) + B(x_e)d + H(x_e)dd,$$

where x_e denotes the current iterative point, d the increment of the argument and $B(x) \approx F'(x_e),\ H(x) \approx F''(x)$.

One of the most popular methods with the convergence order three is the method of tangent hyperbolas (or Chebyshev-Halley method)

$$x_{k+1} = x_k - T_k^{-1}\Gamma_k F(x_k), \qquad\qquad (1.2)$$

where $\Gamma_k = [F'(x_k)]^{-1}$ and $T_k = I - \dfrac{1}{2}\Gamma_k F''(x_k)\Gamma_k F(x_k)$ [3; 7; 8; 13]. It can be rewritten as

$$x_{k+1} = x_k - [F'(x_k) - \frac{1}{2}F''(x_k)\Gamma_k F(x_k)]^{-1}F(x_k). \qquad (1.3)$$

The name "the method of tangent hyperbolas" springs from its geometrical interpretation [7] . Differently from above mentioned references in the proof of a convergence theorem here instead of a Kantorovich condition we assume the

existence of $[F'(x)]^{-1}$ for all x the region under consideration. To get more realistic impression of methods under discussion we study their approximate variants as well unlike the papers [3; 7; 8] . The approach adopted by this report is the use of iterative methods to obtain approximations for T_k^{-1} and/or Γ_k or approximate solutions to the corresponding linear equations.

If $A_k \approx \Gamma_k$ and

$$L(x, x - y) \approx F''(x)(x - y), \ \ x, y \in X,$$

then due to $x_k - y_k = A_k F(x_k)$ and

$$L(x_k, A_k) = L(x_k, A_k F(x_k)) = L(x_k, x_k - y_k)$$

it follows from (1.3) that

$$x_{k+1} = x_k - U_k^{-1} A_k F(x_k), \tag{1.4}$$

where

$$U_k = A_k F'(x_k) - \frac{1}{2} A_k L(x_k, A_k).$$

If, in turn to use instead of U_k^{-1} its approximation V_k we get the method

$$x_{k+1} = x_k - V_k A_k F(x_k). \tag{1.5}$$

Further on we shall suppose the existence and boundedness of the operators $[F'(x_k)]^{-1}$. Likewise we assume the existence of such constants $\alpha, \beta, \lambda, \mu, \Lambda$, M, K, G, G_1 and sequences $\{\gamma_{1k}\}$ and $\{\gamma_{2k}\}$ so that the following inequalities are valid

$$\|F'(x)\| \le M, \ \|F''(x)\| \le K, \ \|A_k\| \le \mu_k \le \mu, \ \|A_k^{-1}\| \le \beta_k \le \beta,$$
$$\|V_k A_k\| \le \lambda_k \|F(x_k)\| \le \lambda \|F(x_k)\|, \quad \|V_k\| \le \Lambda_k \le \Lambda, \tag{1.6}$$
$$(\beta, \lambda, \Lambda, \mu < \infty),$$
$$\|I - U_k V_k\| \le \gamma_{1k}, \ max\{\|I - A_k F'(x_k)\|, \ \|I - F'(x_k) A_k\|\} \le \gamma_{2k}.$$

Theorem 1.1. *Let $x_0 \in X$, $S = \{x \in X : \|x - x_0\| \le \rho\}$ and the following condition are valid on ρ :*

1° Operator F is twice Frechet-differentiable;

2° Second derivative satisfies a Lipschitz-condition

$$\|F''(x) - F''(y)\| \le L_2 \|x - y\|;$$

$3°$ $\|F''(x)(x-y) - L(x,x-y)\| \leq G\|x-y\|^2$, $\|L(x,x-y) \leq G_1\|x-y\|$, $G, G_1 \leq \infty$;

$4°$ *There exist* $\Gamma(x)$ *and* $U^{-1}(x)$ *with* $\|\Gamma(x)\| \leq C$ *and* $\|U^{-1}(x)\| \leq C_1$, $C, C_1 < \infty$;

$5°$ $\delta = \delta_0^{(i)} < 1$, $i = 1,2,3,\ldots$ *(the quantity* δ_0 *is defined differently in the cases 1) – 3)).*

Then the following results are valid:

1) *If*

$$\gamma_{ik} \leq \gamma_{i0} < 1, \quad i = 1, 2$$

and $r_1 = \lambda\|F(x_0)\|/(1-\delta) \leq \rho$, *then the equation* $F(x) = 0$ *has a solution* $x^\star$ *in* S $\|x^\star - x_0\| \leq r_1$, *to which the sequence (1.3) converges with*

$$\|x_k - x^\star\| \leq r_1 \delta^k, \ \delta = \delta_0^{(1)};$$

if $\gamma_{i0} \geq \gamma_{i1} \geq \ldots \geq \gamma_{in} \geq \ldots \geq 0$, *and* $\gamma_{ik} \to 0$, *as* $k \to \infty$, *then* $\delta_k^{(1)} \to 0$ *and the sequence (1.3) converges superlinearly with*

$$\|x_k - x^\star\| \leq r_1 \prod_{m=0}^{k-1} \delta_m^{(1)},$$

where

$$\delta_k^{(i)} = \beta_k \mu_k \gamma_{1k} + \frac{1}{2}\gamma_{1k}\mu_k\lambda_k K\|F(x_k)\| + \frac{1}{2}\gamma_{2k}\lambda_k^2\|F(x_k)\|$$

$$+ \frac{1}{2}\gamma_k^2\lambda_k^2 KG_1\|F(x_k)\|^2 + \frac{1}{4}\mu_k^2\lambda_k^2 KG_1\|F(x_k)\|^2 + \frac{1}{6}\lambda_k^3 L_2\|F(x_k)\|^2.$$

2) *If*

$$\gamma_{1k} \leq C_2\|F(x_k)\|, \ \gamma_{2k} \leq \gamma_{20}, \quad \gamma_{20}, C_2 < \infty,$$

$$\delta = \delta_0^{(2)} = d_0^{(2)}\|F(x_0)\| < 1, \quad d = \lim_{k \to \infty} d_k^{(2)} > 0,$$

$$d_k^{(2)} = \beta_k \mu_k C_2 + \frac{1}{2}\gamma_{20}\lambda_k^2 K + \frac{1}{2}(\mu_k\lambda_k KC_2 + \frac{1}{2}\mu_k^2\lambda_k^2 KG_1$$

$$+ \frac{1}{3}\lambda_k^3 L_2)\|F(x_k)\|,$$

then the equation (1.1) has a solution $x^\star$ *in* S, $\|x^\star - x_0\| \leq r_2$, *to which the sequence (1.5) converges quadratically*

$$\|x_k - x^\star\| \leq \lambda H_k^{(2)}(\delta)/d, \ H_k^{(2)}(\delta) = \sum_{i=k}^{\infty} \delta^{2^i}, \ r_2 = \lambda H_0^{(2)}(\delta)/d \leq \rho.$$

3) *If*

$$\gamma_{1k} \le C_3 \|F(x_k)\|^2, \ \ \gamma_{2k} \le C_4 \|F(x_k)\|, \ \ C_3, C_4 < \infty, \ \ r_3 = H_0^{(3)}(\delta)/d \le \rho,$$

where

$$H_k^{(3)}(\delta) = \sum_{i=k}^{\infty} \delta^{3^i}, \ \ \delta = \delta_0^{(3)} = \sqrt{d_0}\|F(x_0)\| < 1$$

and

$$d = d_0^{(3)} = \beta_0 \mu_0 C_3 + \frac{1}{2}\mu_0 \lambda_0 K C_3 + \frac{1}{2}\lambda_0^2 K C_4$$
$$+ \frac{1}{4}\mu_0^2 \lambda_0^2 K G_1 + \frac{1}{2}\lambda_0 \mu_0^2 G + \frac{1}{6}\lambda_0^3 L_2,$$

then the sequence (1.5) *converges cubically*

$$\|x_k - x^\star\| \le (\lambda/\sqrt{d})H_k^{(3)}(\delta).$$

The proof of this theorem rests on a more general theorem from [11]. Since the proof is rather complicated then it is omitted here and shall be presented elsewhere in detail.

There are a lot of methods having a high order of convergence $p > 2$, but in practice they are relatively little exploited. This is partially due to the fact that their computational schemes of the execution of one iteration are laborious, they frequently require the evaluation of derivatives of order greater than one and a good initial guess since their advantages become evident in the close vicinity of the solution. It is known that methods with the high order convergence are in many cases more efficient than Newton type methods as applied to solving nonlinear integral and differential equations [3].

2. PARTICULAR METHODS

Further we shall consider some possibilities how to avoid the evaluation of F'' and thereby probably to reduce computational cost.

Approximating the term $F''(x_k)A_k F(x_k)$ by the expression

$$L(x_k, A_k) = 2[F'(x_k) - F'(x_k - \frac{1}{2}A_k F(x_k))],$$

and using

$$F'(x_k) - \frac{1}{2}F''(x_k)\Gamma_k F(x_k) \approx F'(x_k - \frac{1}{2}L(x_k, A_k)) = F'(x_k - \frac{1}{2}A_k F(x_k))$$

the method (1.2) becomes

$$x_{k+1} = x_k - [F'(x_k - \frac{1}{2}A_k F(x_k))]^{-1} F(x_k), \qquad (2.1)$$

Since

$$[F'(x_k - \frac{1}{2}A_k F(x_k))]^{-1} = \{A_k^{-1}[A_k F(x_k) - \frac{1}{2}A_k L(x_k, A_k)]\}^{-1} = U_k^{-1} A_k,$$

then (2.1) can be rewritten as

$$x_{k+1} = x_k - U_k^{-1} A_k F(x_k). \qquad (2.2)$$

If $\|I - U_k V_k\| \leq \gamma_{1k} = O(\|F(x_k)\|^2)$ and $\|I - F'(x_k)A_k\| \leq \gamma_{2k} = O(\|F(x_k)\|)$ then by Theorem 1.1 the method (2.1) is cubically convergent.

The method (2.1) with $A_k = \Gamma_k$ coincides with the well-known midpoint method. It is shown in [1] that midpoint method can be used for solving equations with non-differentiable operators.

Another possibility to get rid of the evaluation of F'' is to replace it by a fixed bilinear operator:

$$x_{k+1} = x_k - \left(I - \frac{1}{2}A_k \Phi A_k F(x_k)\right)^{-1} A_k F(x_k), \qquad (2.3)$$

where $\Phi : X \times X \to Y$ is a general bounded bilinear operator. The method (2.3) has similar computational cost as the Newton method but it remains faster than the Newton method as shown in [5].

Replacing in (2.1) A_k by

$$[F'(\Theta_{k-1})]^{-1} \quad \text{and} \quad [F'(x_k) - \frac{1}{2}A_k F(x_k)]^{-1}$$

by $[F'(\Theta_k)]^{-1}$, where

$$\Theta_k = \begin{cases} x_0, & \text{if } k = 0, \\ x_k - \frac{1}{2}[F'(\Theta_{k-1})]^{-1} F(x_k), & \text{if } k \geq 1, \end{cases}$$

the method (2.1) transforms to

$$x_{k+1} = x_k - [F'(\Theta_k)]^{-1} F(x_k), \qquad (2.4)$$

having the convergence order equal to $1 + \sqrt{2}$.

Indeed,

$$\|I - A_k F'(x_k)\| = \|F'(\Theta_{k-1})]^{-1}(F'(\Theta_{k-1}) - F'(x_k)\|$$

$$\leq \frac{3}{2} CK \|F'(\Theta_{k-1})]^{-1}\|\|F(x_{k-1})\|,$$

i.e. $\gamma_{2k} = O(\|F(x_{k-1})\|)$ and it can be shown that in this case the prevailing term in the Taylor series for $F(x_{k+1})$ is $O(\|F(x_{k-1})\|\|F(x_k)\|^2)$ which implies that convergence rate is equal to $1 + \sqrt{2}$ (see for [9]).
Let W_k be an operator which approximates the inverse $[F'(x_k - \frac{1}{2} A_k F(x_k))]^{-1}$, then W_k can be written as $W_k = V_k A_k$ with $V_k \approx U_k^{-1}$ because of $[F'(x_k - \frac{1}{2} A_k F(x_k))]^{-1} = U_k^{-1} A_k$ and $A_k = [F'(\Theta_{k-1})]^{-1}$.
If $\|I - U_k V_k\| = O(\|F(x_{k-1})\|\|F(x_k)\|^2)$, i.e. if the approximation error is not greater than $O(\|F(x_{k-1})\|\|F(x_k)\|)$, then the convergence order equal to $1 + \sqrt{2}$ is preserved.

3. ORGANIZING PARALLEL COMPUTATIONS

One of the potential ways to reduce the total time needed for computing a solution with a prescribed accuracy is the use of parallel computation. Methods with the high order convergence offer various possibilities to organize parallel computation. Further, we shall discuss some aspects of parallel computation concerning rapidly convergent method as applied to the solution of problems of moderate size in finite dimensional spaces. The process is organized by designating the computers (processors) as the master (that coordinates the process of the other computers) and slaves (that execute different tasks). In the beginning of the computational process the researcher allocates a certain number of subproblems to the processors. Computational experience with various multiprocessors indicate that performance of parallel methods depends critically on efficient load balancing. Herewith, communication cost must be considered. Sometimes the communication complexity is higher than the computational complexity, in other words, more time is spent routing data among processors than actually manipulating data [2].

One possibility to save arithmetic operations on a sequential computer when using an inexact Newton method is to recalculate the A_k explicitly after a certain number of iteration steps while at the intermediate steps, say at l ones, to use the formula

$$A_{k+i} u_i = A_{k+i-1}[2I - F'(x_{k+i} A_{k+i-1}]u_i, \; i = 1, \ldots, l, \quad (3.1)$$

where $u_i = F(x_{k+i})$ and $A_{k+i} u_i$ is calculated by matrix-vector multiplication, i.e. the linear operator A_{k+i} is defined by the recurrence formula (3.1) [6]. This idea will be used by us for organizing parallel computations.

First we consider a Newton-type method

$$x_{k+1} = x_k - A_k F(x_k), \tag{3.2}$$

$$A_{k+1} = A_k(2I - F'(x_k)A_k), \tag{3.3}$$

where x_0 and A_0 are initial approximations to the solution $x^\star$ and $[F'(x^\star)]^{-1}$ respectively, i.e. $A_{k+1} =\approx [F'(x_k)]^{-1}$, while values of expression like (3.1) can be computed using parallel linear algebra computation. If only one processor is in use and instead of the execution the formula (3.3) to compute $A_k = [F'(x_k)]^{-1}$, then we get from (3.2) the usual sequential Newton method. If we apply two processors and take $A_k = [F'(x_{k-1})]^{-1}$ while using one processor to compute x_{k+1} by the formula (3.2) and the other one to compute $F'(x_{k-1})]^{-1}$, then we obtain a parallel variant of the Newton-type method

$$x_{k+1} = x_k - [F'(x_{k-1})]^{-1}F(x_k). \tag{3.4}$$

Method (3.4) has two serious disadvantages:
* its convergence order is $\frac{1}{2}(1 + \sqrt{5}) \approx 1,618$;
* the processors are unequally loaded and one of them stays idle.
Note, that its convergence rate follows easily from Theorem 1.1.

To alleviate the problem of unequal load balancing we consider the following accelerated synchronous parallel Newton-type method

$$x_{k+1} = x_k - A_k \sum_{i=0}^{p-1}(I - F'(x_k)A_k)^i F(x_k), \tag{3.5}$$

$$A_{k+1} = A_k \sum_{i=0}^{q-1}(I - F'(x_k)A_k)^i, \ A_k = [F'(x_{k-1})]^{-1}, \tag{3.6}$$

which according to the convergence Theorem 1.1 (see also [11]) has the convergence order equal to two provided $p, q \geq 2$. The use of the modified formula (3.5) instead of (3.2) permits to increase the convergence order at the cost of additional computational work of order $O(n^2)$ using matrix-vector multiplication. Matrix multiplications are attainable by means of scalar products that are independent and therefore can be performed simultaneously. There are different types of matrix multiplications that are suitable for parallel execution on computers with certain given structure [2].

Let us now consider a synchronous parallel variant of (2.1)

$$y_k = x_k - \frac{1}{2}A_k F(x_k), \tag{3.7}$$

$$x_{k+1} = x_k - B_k F(x_k), \tag{3.8}$$

where

$$A_k = \Gamma_{k-1} \sum_{i=0}^{p-1} (I - F'(x_k)\Gamma_{k-1})^i,$$

$$B_k = \Gamma_{k-1} \sum_{i=0}^{q-1} (I - F'(y_k)\Gamma_{k-1})^i,$$

are computed by matrix-vector multiplication on the master processor and Γ_{k-1} is computed explicitly on another processor. From Theorem 1 [11] it follows that method (3.7), (3.8) has the convergence rate equal to 3 provided $p, q \geq 6$. For $p \geq 1$ and $q \geq 2$ the convergence order is equal to 2.

4. CONCLUDING REMARKS

The use of parallel algorithms under discussion permits to obtain, as by-product, approximations to inverse operators of F'. Knowledge of $[F']^{-1}$ or its approximation is desirable in many cases. It facilitates to evaluate a condition number and thereby to estimate stability of iterative processes, to obtain error bounds, to pass easily on to interval-arithmetic versions based on Krawczyk transformation, etc.

Although iterative methods with the convergence order $p > 2$ have interest in their own, they seem to be very useful in polyalgorithmic computational schemes. The property of global convergence is a criterion for robustness of the algorithm. A widespread strategy is the use of a Newton-type method if it works, otherwise, to switch on a slower but more sure global method, e.g. on a method based on the steepest descent direction. A combination of the steepest descent method with the midpoint method or its variants with $A_k \approx \Gamma_k$ is expected to be more efficient in the sense of stability than one with a Newton-type method, and that benefit should probably repay the additional computational cost per iteration. The reason for doing so is that the Newton method and its numerous variants are unable to perform efficiently (converge quickly) or even can break down if F' is ill-conditioned or singular, since they are based on a linear model but iterative methods of order $p \geq 3$ use at least a quadratic model. For the same reason, they perform efficiently in the globalization of iterative processes if one uses a continuation strategy. As in generally known, homotopy methods suffer from the disadvantage that the Jacobian at some iterative points may become singular [12]. A more thorough discussion of polyalgorithmic approach and a few results of computational experiments concerning the methods under discussion can be found in [11].

Asynchronous parallel methods using the same idea for parallelization seem to be more promising and those will be discussed in the future.

REFERENCES

[1] K. Argyros. An error analysis for the midpoint method. *Tamkang Journal of Mathematics*, **30(2)**, 71 – 83, 1999.

[2] J. Blazewicz, K. Ecker, B. Plateau and D. Trystram. *Handbook on Parallel and Distributing Processing*. Springer Verlag, 2000.

[3] B. Döring. Iterative lösung gewisser Randwertprobleme und Integralgleichnungen. *Applicae Matematiky*, **23**, 1 – 31, 1978.

[4] G.-P. Ehle and H. Schwetlick. Discretized Euler-Chebyshev multistep method. *SIAM J. Numer. Anal.*, **13(3)**, 433 – 443, 1976.

[5] J.A. Ezquerro and M.A. Hernandez. Newton-type iterations approximations of Chebyshev method. *Southwest Journal of Pure and Applied Mathematics (Electronic Journal)*, **1**, 44 – 55, 1998.

[6] O. Hald. On a Newton-Moser type method. *Numer. Math.*, **23**, 411 – 426, 1975.

[7] M. A. Mertvetsova. Analog of the process of tangent hyperbolas for general functional equations (in russian). *Dokl. AN SSSR*, **88(4)**, 611 – 614, 1953.

[8] R. E. Moore and Shen Zuhe. An interval version of chebyshev's method for nonlinear operator equations. *Nonlinear Analysis, Theory, Methods and Applications*, **7(1)**, 21 – 34, 1983.

[9] J.W. Schmidt. Eine übertragung der Regula Falsi auf Gleichungen in Banachräumen II. Nictlineare Gleichungssysteme. *ZAMM*, **43(3)**, 97 – 110, 1963.

[10] R.D. Schnabel and P.D. Frank. Tensor method for nonlinear equations. *SIAM J. Numer. Anal.*, **21(5)**, 815 – 843, 1984.

[11] O. Vaarmann. On high order iterative methods for nonlinear problems. In: Th.M. Rassias(Ed.), *Nonlinear Mathematical Analysis and Applications*, FL: Hadronic Press, Palm Harbor, 275 – 299, 1998.

[12] L.T. Watson. Numerical linear algebra aspects of globally convergent homotopy methods. *SIAM Rev.*, **28(4)**, 529 – 545, 1986.

[13] Z. P. Zykova. A study of a family of iterative methods with the convergence order three, 1984. PhD Thesis, Leningrad, LGU (in Russian)

Apie hiperbolinio tangento aproksimacijos metodus

I. Kaldo, O. Vaarmann

Netiesinių operatorinių lygčių Banacho erdvėje sprendimui nagrinėjami kraštinių hiperbolių metodo variantai. Šių metodų šeima apima specialiuosius metodus, pagrįstus iteraciniais metodais, kurie įgalina gauti blogą sprendinį su tam tikru tikslumu, sprendžiant susijusias tiesines lygtis kiekvienoje iteracijoje. Pateikta lokalaus konvergavimo teorema bei konvergavimo greitis. Svarstoma skaičiavimų išlygiagretinimo galimybės. Skaičiavimo eksperimentas su įvairiais multi-procesoriais rodo, kad lygiagrečiųjų metodų vykdymas priklauso iš esmės nuo atliekamo darbo išbalansavimo.