NGR-44-006-089

TM: J M Lewallen
Manned

AERO-ASTRONAUTICS REPORT NO. 78

MODIFIED QUASILINEARIZATION METHOD FOR SOLVING NONLINEAR EQUATIONS

by

A. MIELE, S. NAQVI, AND A.V. LEVY

RICE UNIVERSITY

970

Modified Quasilinearization Method for Solving Nonlinear Equations[1]

by

A. MIELE[2], S. NAQVI[3], and A.V. LEVY[3]

Abstract. This paper presents a general method for solving nonlinear equations of the form $\varphi(x) = 0$, where x and φ are n-vectors. The method is based on the consideration of the performance index P, the cumulative error in the equations.

A modified quasilinearization algorithm is generated by requiring the first variation of the performance index δP to be negative. This algorithm differs from the ordinary quasilinearization algorithm because of the inclusion of the scaling factor or stepsize α in the system of variations. The main property of the modified quasilinearization algorithm is the descent property: if the stepsize α is sufficiently small, the reduction in P is guaranteed. Convergence to the desired solution is achieved when the inequality $P \leq \epsilon$ is met, where ϵ is a small, preselected number.

The variations per unit stepsize $\Delta x/\alpha = A$ are governed by a system of n linear equations, which is solved by Gaussian elimination. Several numerical examples are presented. They illustrate (i) the simplicity as well as the rapidity of convergence of the modifed quasilinearization algorithm and (ii) the importance of stepsize control.

[1] This work was supported by the National Science Foundation, Grant No. GP-18522.

[2] Professor of Astronautics and Director of the Aero-Astronautics Group, Department of Mechanical and Aerospace Engineering and Materials Science, Rice University, Houston, Texas.

[3] Graduate Student in Aero-Astronautics, Department of Mechanical and Aerospace Engineering and Materials Science, Rice University, Houston, Texas.

NGR-44-006-089

1. Introduction

In recent years, considerable attention has been devoted to the solution of nonlinear equations, algebraic or transcendental (see, for example, Refs. 1-10). The most direct way to solve these equations is to employ quasilinearization, that is, to replace the nonlinear system by one that is linear in the perturbations about a nominal point. The resulting algorithm is called ordinary quasilinearization algorithm.

The main advantage of the ordinary quasilinearization algorithm is simplicity and rapidity of convergence if the nominal point is in the neighborhood of the solution point. There are cases, however, where ordinary quasilinearization diverges due to excessive magnitude of the variations. This is why it is convenient to imbed the linearized system into a more general system by means of the scaling factor α, $0 \leq \alpha \leq 1$, applied to each forcing term. The resulting algorithm is called modified quasilinearization algorithm.

At first glance, the above imbedding procedure may seem arbitrary (see, for example, Refs. 7 and 10). However, a rigorous conceptual justification can be given through the consideration of the performance index P: this is the cumulative error in the equations. By computing the first variation of P and requiring δP to be negative, one generates the modified quasilinearization algorithm. The main property of this algorithm is the descent property: if the stepsize α is sufficiently small, the reduction in P is guaranteed. In addition, the performance index P can also be employed as a convergence criterion: the algorithm is terminated when P becomes smaller than some preselected value.

2. Modified Quasilinearization

Consider a system described by the equation

$$\varphi(x) = 0 \tag{1}$$

where x and φ are n-vectors. Assume that the first derivative of the function φ with respect to the vector x exists and is continuous. Also, assume that a solution to (1) exists. The problem is to find the vector x which solves Eq. (1).

2.1. Performance Index. In general, the system (1) is nonlinear, so that approximate methods must be employed. In this connection, let the performance index P be defined as[4]

$$P = \varphi^T(x)\varphi(x) \tag{2}$$

The scalar function P measures the error in the constraint; therefore, P = 0 for any x satisfying Eq. (1) and P > 0 otherwise. When approximate methods are employed, they must ultimately lead to a state x such that

$$P \leq \epsilon \tag{3}$$

where the quantity ϵ is a small, preselected number.

2.2. Modified Quasilinearization. Here, we present a modification of the quasilinearization algorithm which has a descent property in the performance index P. Consider a nominal point x and a varied point $\tilde{x}$ such that

$$\tilde{x} = x + \Delta x \tag{4}$$

---

[4] The superscript T denotes the transpose of a matrix.

The passage from the nominal point to the varied point causes the performance index P to change. To first order, we see that

$$\delta P = 2\varphi^T(x)\,\delta\varphi(x) \tag{5}$$

where the symbol $\delta(\ldots)$ denotes the first variation.

Next, consider the system of variations defined by

$$\delta\varphi(x) = -\alpha\varphi(x) \tag{6}$$

where α is a scaling factor (or stepsize) in the range

$$0 \le \alpha \le 1 \tag{7}$$

Consequently, the first variation of the performance index P becomes

$$\delta P = -2\alpha\varphi^T(x)\varphi(x) \tag{8}$$

and, in the light of the definition (2), is equivalent to

$$\delta P = -2\alpha P \tag{9}$$

Note that, for any nominal point x not satisfying Eq. (1),

$$P > 0 \tag{10}$$

Therefore, for α positive, one has

$$\delta P < 0 \tag{11}$$

This is the basic descent property of the algorithm defined by Eq. (6): it guarantees that, if α is sufficiently small,

$$\tilde{P} < P \tag{12}$$

2.3. System of Variations. To first order, the change of the function $\varphi(x)$ is related to the change of the state Δx as follows:[5]

$$\delta\varphi(x) = \varphi_x^T(x)\Delta x \tag{13}$$

where the matrix φ_x is n x n. Consequently, Eq. (6) can be rewritten as

$$\varphi_x^T(x)\Delta x + \alpha\varphi(x) = 0 \tag{14}$$

For a given value of α, Eq. (14) is equivalent to n scalar equations which are linear in the n components of the vector Δx. The associated algorithm is called modified quasi-linearization.

For $\alpha = 1$, Eq. (14) becomes identical with that of ordinary quasilinearization, that is, the equation obtained by linearizing Eq. (1) about a nominal point x. While modified quasilinearization exhibits the descent property (11)-(12), this is not necessarily the case with ordinary quasilinearization. This means that, if Eq. (14) is employed with $\alpha = 1$, the performance index P may actually increase when passing from the nominal point x to the varied point $\tilde{x}$.

2.4. Coordinate Transformation. To simplify the problem, we introduce the auxiliary variable

$$A = \Delta x/\alpha \tag{15}$$

---

[5] The matrix $\varphi_x(x)$ is defined so that its ith column is the gradient of the ith scalar component of $\varphi(x)$ with respect to the vector x.

and rewrite Eq. (14) in the form

$$\varphi_x^T(x)A + \varphi(x) = 0 \tag{16}$$

This vector equation is equivalent to n scalar equations which are linear in the n components of the vector A. With A known and the stepsize α specified (see the following section), the correction Δx is obtained from Eq. (15), and the varied point $\tilde{x}$ is computed from Eq. (4).

2.5. Determination of the Stepsize. After combining Eqs. (4) and (15), we obtain the relation

$$\tilde{x} = x + \alpha A \tag{17}$$

Since x is given and A is known from Eq. (16), Eq. (17) yields a one-parameter family of solutions, the parameter being the stepsize α. For this one-parameter family, the performance index P becomes a function of the form

$$P = P(\alpha) \tag{18}$$

At $\alpha = 0$, the slope of this function is negative and is given by

$$P_\alpha(0) = -2P(0) \tag{19}$$

The function (18) exhibits a relative minimum with respect to α, that is, a point where

$$P_\alpha(\alpha) = 0 \tag{20}$$

This point can be determined by means of a one-dimensional search (for example, using quadratic interpolation, cubic interpolation, or quasilinearization). Ideally, this procedure should be used iteratively until the modulus of the slope satisfies the following inequality:

$$|P_\alpha(\alpha)| \leq \theta \tag{21}$$

where θ is a small, preselected number.

Since the rigorous determination of α takes time on a computer, one might renounce solving Eq. (20) with a particular degree of precision and determine the stepsize in a noniterative fashion. To this effect, we first assign the value

$$\alpha = 1 \tag{22}$$

to the stepsize; this corresponds to full quasilinearization of Eq. (1) and is the value which would solve Eq. (20) exactly, should the function (1) be linear. Of course, the stepsize (22) is acceptable only if

$$P(\alpha) < P(0) \tag{23}$$

Otherwise, the previous value of α must be replaced by some smaller value in the range (7) (for example, using a bisection process) until Ineq. (23) is met. This is guaranteed by the descent property (11)-(12).

2.6. Summary of the Algorithm. In the light of the previous discussion, we summarize the modified quasilinearization algorithm as follows:

(a) Assume a nominal point x. At this point, determine the vector $\varphi(x)$ and the matrix $\varphi_x(x)$.

(b) Determine the vector A by solving Eq. (16).

(c) Consider the one-parameter family of solutions (17) and perform a one-dimensional search on the function (18); specifically, perform a bisection process on α (starting from $\alpha = 1$), and continue the process until Ineq. (23) is satisfied.

(d) With the stepsize α known, compute the varied point $\tilde{x}$ from (17).

(e) With $\tilde{x}$ known, the iteration is completed. The varied point $\tilde{x}$ becomes the nominal point x for the next iteration. That is, return to (a) and iterate the algorithm.

(f) The algorithm is terminated when the stopping condition (3) is satisfied.

3. Numerical Examples[6]

In order to illustrate the theory, several numerical examples were developed using a Burroughs B-5500 computer and double-precision arithmetic. The algorithm was programmed in FORTRAN IV.

Convergence was defined as follows:

$$P \leq 10^{-20} \tag{24}$$

and the number of iterations at convergence N_* was recorded. Conversely, nonconvergence was defined by means of the inequalities

(a) $$N \geq 100 \tag{25}$$

or

(b) $$N_S \geq 20 \tag{26}$$

or

(c) $$M \geq 0.4 \times 10^{69} \tag{27}$$

Here, N is the iteration number, N_S is the number of bisections of the stepsize α (starting from $\alpha = 1$) required to satisfy Ineq. (23), and M is the modulus of any of the quantities employed in the algorithm. Satisfaction of Ineq. (25) indicates divergence or extreme slowness of convergence; in turn, satisfaction of Ineq. (26) indicates extreme smallness of the displacement Δx; finally, satisfaction of Ineq. (27) indicates exponential overflow for the Burroughs B-5500 computer: the computer program is automatically stopped.

---

[6] For simplicity, the symbols employed in this section denote scalar quantities.

Example 3.1. Consider the nonlinear system

$$x^2 - y - 8 = 0 \quad , \quad x^4 - y^2 = 0 \tag{28}$$

which has solutions

$$x = 2 \quad , \quad y = -4 \tag{29}$$

and

$$x = -2 \quad , \quad y = -4 \tag{30}$$

Assume the following nominal coordinates:

$$x = y = 1/2 \tag{31}$$

which do not satisfy the system (28). Starting with these nominal coordinates, we employed the algorithm of Section 2. Convergence to the solution (29) was achieved in $N_* = 8$ iterations (see Table 1). The converged coordinates are

$$x = 2.0000 \quad , \quad y = -4.0000 \tag{32}$$

Table 1. Results pertaining to Example 3.1.

N	α	P
0	–	0.6×10^{2}
1	1/16	0.6×10^{2}
2	1/4	0.4×10^{2}
3	1/4	0.1×10^{2}
4	1/2	0.2×10^{1}
5	1	0.5×10^{-1}
6	1	0.3×10^{-5}
7	1	0.7×10^{-15}
8	1	0.5×10^{-34}

Example 3.2. Consider the nonlinear system

$$x^2 - x^3y^5z + z^3 - 1 = 0$$

$$y \sin\left(\frac{1}{2}\pi x\right) - 1 = 0 \tag{33}$$

$$2xy^2z^3 + y \exp(1-x) - 3 = 0$$

a known solution of which is

$$x = y = z = 1 \tag{34}$$

Assume the following nominal coordinates:

$$x = y = z = 1/4 \tag{35}$$

which do not satisfy the system (33). Starting with these nominal coordinates, we employed the algorithm of Section 2. Convergence to the solution (34) was achieved in $N_* = 7$ iterations (see Table 2). The converged coordinates are

$$x = 1.0000 \ , \quad y = 1.0000 \ , \quad z = 1.0000 \tag{36}$$

Table 2. Results pertaining to Example 3.2.

N	α	P
0	–	0.7×10^{1}
1	1/2	0.5×10^{1}
2	1	0.1×10^{0}
3	1	0.1×10^{-2}
4	1	0.1×10^{-6}
5	1	0.9×10^{-11}
6	1	0.4×10^{-19}
7	1	0.9×10^{-36}

Example 3.3. Consider the nonlinear system

$$
\begin{aligned}
&(x - u)^2 + (y - w)^2 + (x + y + u + w)^2 - 16 = 0 \\
&x \sin(\tfrac{1}{2}\pi u) + y \cos(\tfrac{1}{2}\pi w) - 1 = 0 \\
&x + y^2 + u^3 + w^4 - 4 = 0 \\
&x + 2y + 3u + 4w - 10 = 0
\end{aligned}
\tag{37}
$$

a known solution of which is

$$x = y = u = w = 1 \tag{38}$$

Assume the following nominal coordinates:

$$x = y = u = w = 3 \tag{39}$$

which do not satisfy the system (37). Starting with these nominal coordinates, we employed the algorithm of Section 2. Convergence to the solution (38) was achieved in $N_* = 25$ iterations (see Table 3). The converged coordinates are

$$x = 0.9999 \;, \quad y = 1.0000 \;, \quad z = 1.0000 \;, \quad u = 0.9999 \tag{40}$$

Table 3. Results pertaining to Example 3.3.

N	α	P	N	α	P
0	–	0.3×10^{5}	13	1	0.2×10^{-6}
1	1/4	0.2×10^{5}	14	1	0.1×10^{-7}
2	1	0.1×10^{5}	15	1	0.9×10^{-9}
3	1	0.2×10^{4}	16	1	0.6×10^{-10}
4	1/2	0.1×10^{4}	17	1	0.3×10^{-11}
5	1/4	0.8×10^{3}	18	1	0.2×10^{-12}
6	1	0.1×10^{3}	19	1	0.1×10^{-13}
7	1	0.4×10^{1}	20	1	0.9×10^{-15}
8	1	0.1×10^{0}	21	1	0.5×10^{-16}
9	1	0.8×10^{-2}	22	1	0.3×10^{-17}
10	1	0.1×10^{-2}	23	1	0.2×10^{-18}
11	1	0.6×10^{-4}	24	1	0.1×10^{-19}
12	1	0.4×10^{-5}	25	1	0.9×10^{-21}

Example 3.4. Consider the nonlinear system

$$(x - y)^2 + (y - z)^2 + (2z - u - w)^2 = 0$$

$$x^2 + y^2 + z^2 + u^2 + w^2 - 5 = 0$$

$$(x - 1)^2 + (y - 2)^2 + w^4 - 2 = 0 \tag{41}$$

$$x + 2y^2 + 3z^3 + 4u^4 + 5w^5 - 15 = 0$$

$$x^2 + xyz - u^3 - 1 = 0$$

a known solution of which is

$$x = y = z = u = w = 1 \tag{42}$$

Assume the following nominal coordinates:

$$x = y = z = 2 \quad , \quad u = w = 3 \tag{43}$$

which do not satisfy the system (41). Starting with these nominal coordinates, we employed the algorithm of Section 2. Convergence to the solution (42) was achieved in $N_* = 25$ iterations (see Table 4). The converged coordinates are

$$x = 1.0000 \;,\; y = 1.0000 \;,\; z = 0.9999 \;,\; u = 1.0000 \;,\; w = 1.0000 \tag{44}$$

Table 4. Results pertaining to Example 3.4.

N	α	P	N	α	P
0	–	0.2×10^{7}	13	1	0.2×10^{-6}
1	1/16	0.2×10^{7}	14	1	0.1×10^{-7}
2	1	0.3×10^{6}	15	1	0.1×10^{-8}
3	1	0.8×10^{5}	16	1	0.7×10^{-10}
4	1	0.1×10^{5}	17	1	0.4×10^{-11}
5	1	0.1×10^{4}	18	1	0.2×10^{-12}
6	1	0.8×10^{2}	19	1	0.1×10^{-13}
7	1	0.4×10^{1}	20	1	0.1×10^{-14}
8	1	0.2×10^{0}	21	1	0.6×10^{-16}
9	1	0.1×10^{-1}	22	1	0.4×10^{-17}
10	1	0.1×10^{-2}	23	1	0.2×10^{-18}
11	1	0.7×10^{-4}	24	1	0.1×10^{-19}
12	1	0.4×10^{-5}	25	1	0.1×10^{-20}

4. Remarks

The following remarks are pertinent to the previous theoretical development.

Remark 4.1. If the stepsize is set at the constant value $\alpha = 1$, the modified quasilinearization algorithm of Section 2 reduces to the ordinary quasilinearization algorithm. While modified quasilinearization exhibits the descent property (11)-(12), this is not necessarily the case with ordinary quasilinearization. Therefore, in ordinary quasilinearization, the performance index P may actually increase when passing from the nominal point x to the varied point $\tilde{x}$.

With reference to the examples of Section 3, computer runs were made employing both modified quasilinearization and ordinary quasilinearization: in Table 5, the number of iterations at convergence N_* is indicated and, as the table shows, the experimental evidence is in favor of modified quasilinearization. It is emphasized that the above conclusion was obtained through particular examples and that, consequently, the subject requires further investigation.

Table 5. Number of iterations at convergence.

	N_*	
	$\alpha \leq 1$	$\alpha = 1$
Example 3. 1	8	12
Example 3. 2	7	11
Example 3. 3	25	66
Example 3. 4	25	Nonconvergence (a)

Remark 4.2. The fundamental property of the modified quasilinearization algorithm is the descent property (11)-(12). This local property guarantees the decrease of the performance index P when passing from the nominal point x to the varied point $\tilde{x}$. However, it does not guarantee convergence; that is, it does not guarantee that $P \to 0$ as $N \to \infty$. This is due to the fact that convergence depends on the analytical nature of the function $\varphi(x)$ and on the nominal point x chosen in order to start the algorithm.

The above point can be illustrated with reference to Example 3.4 and the following nominal coordinates:

$$x = 1 \;,\quad y = 2 \;,\quad z = 3 \;,\quad u = 4 \;,\quad w = 5 \tag{45}$$

Ordinary quasilinearization fails to converge because of exponential overflow [nonconvergence (c)]. Modified quasilinearization fails to converge because of excessive number of stepsize bisections [nonconvergence (b)].

5. Discussion and Conclusions

In this paper, a general method for solving nonlinear equations of the form $\varphi(x) = 0$ is presented, where x and φ are n-vectors. The method is based on the consideration of the performance index P, the cumulative error in the equations.

A modified quasilinearization algorithm is generated by requiring the first variation of the performance index δP to be negative. The algorithm has the form $\tilde{x} = x + \alpha A$. Here, α, $0 \leq \alpha \leq 1$, is the stepsize and A is obtained by solving a system of n linear equations by Gaussian elimination.

The main property of the modified quasilinearization algorithm is the descent property: if the stepsize α is sufficiently small, the reduction in P is guaranteed. Not only is P employed as a guide during progression of the algorithm, but also as a convergence criterion: the algorithm is terminated when the performance index P becomes smaller than some preselected value.

Several numerical examples are presented. They illustrate (i) the simplicity as well as the rapidity of convergence of the algorithm and (ii) the importance of stepsize control.

References

1. DAVIDENKO, D. F., On a New Method of Numerical Solution of Systems of Nonlinear Equations (in Russian), Doklady Akademiia Nauk SSSR, Vol. 88, No. 4, 1953.

2. ROSENBROCK, H. H., An Automatic Method for Finding the Greatest or Least Value of a Function, Computer Journal, Vol. 3, No. 3, 1960.

3. FREUDENSTEIN, F., and ROTH, B., Numerical Solution of Systems of Nonlinear Equations, Journal of the Association for Computing Machinery, Vol. 10, No. 4, 1963.

4. KIZNER, W., A Numerical Method for Finding Solutions of Nonlinear Equations, SIAM Journal on Applied Mathematics, Vol. 12, No. 2, 1964.

5. POWELL, M. J. D., A Method for Minimizing a Sum of Squares of Nonlinear Functions without Calculating Derivatives, Computer Journal, Vol. 7, No. 4, 1965.

6. BARNES, J. G. P., An Algorithm for Solving Nonlinear Equations on the Secant Method, Computer Journal, Vol. 8, No. 1, 1965.

7. BROYDEN, C. G., A Class of Methods for Solving Nonlinear Simultaneous Equations, Mathematics of Computation, Vol. 19, No. 92, 1965.

8. DEIST, F. H., and SEFOR, L., Solution of Systems of Nonlinear Equations by Parameter Variation, Computer Journal, Vol. 10, No. 1, 1967.

9. BROYDEN, C. G., Quasi-Newton Methods and Their Application to Function Minimization, Mathematics of Computation, Vol. 21, No. 99, 1967.

10. BROYDEN, C. G., A New Method of Solving Nonlinear Simultaneous Equations, Computer Journal, Vol. 12, No. 1, 1969.