$n_i, \vec{n}$ = unit normal to a surface element
N_c = capillary vector; cf. Eq. 46
N_c = capillary number, $|\vec{N}_c|$
p = pressure
$p_\alpha, \alpha = 1,2$ = pressure in phase α
p_c = capillary pressure
p_o = integration constant; cf. Eq. 21
$q_i, \vec{q}$ = filter velocity
$\vec{q}^{(\alpha)}, \vec{q}_\alpha$ = filter velocity of phase $\alpha = 1,2$
$s_\alpha, \alpha = 1,2$ = saturation of phase α
s_{or} = residual oil saturation
$\vec{u}$ = velocity of interface
$\vec{v}, v_i$ = fluid velocity
$\vec{v}^{(\alpha)}, \vec{v}_\alpha$ = fluid velocity of phase $\alpha = 1,2$
V = volume of regions used in volume averaging; cf. Eq. 1

Greek Letters

α = interfacial tension
α^* = effective interfacial tension
$\partial_i, i = x,y,z$ = partial derivatives
η = porosity
η_α = fractional volume occupied by phase $\alpha = 1,2$
π_{ik} = momentum flux density tensor
ρ = mass density
σ_{ik} = viscosity stress tensor
μ = viscosity

Subscripts and Superscripts

$\alpha, (\alpha) = 1,2$ = phase α
$i,j,k = x,y,z$ = cartesian components

Special Symbols

$\langle\ \rangle$ = average over V; cf. Eq. 1
$\overline{(\ )}$ = occupied by the phase; cf. Eq. 2

LITERATURE CITED

Chuoke, R. L., P. van Meurs, and C. von der Poel, "The instability of slow, immiscible, viscous liquid-liquid displacements in permeable media," *Trans. Am. Inst. Min. Eng.*, **216**, 188 (1959).

Clampitt, R. L., and P. D. Fleming, "Research digs into micellar flooding processes," *Oil and Gas J.* (Jan. 14, 1980).

Hubbard, M. K., "Darcy's law and the field equations of the flow of underground fluids," *A.I.M.E. Petrol. Trans.*, **207**, 222 (1956).

Larson, R. G., L. E. Scriven, and H. T. Davis, "Percolation theory of residual phases in porous media," *Nature*, **268**, 409 (1977).

Melrose, J. R., and C. F. Brandner, "Role of capillary forces in determining microscopic displacement efficiency for oil recovery by water-flooding," *J. Can. Pet. Tech.*, **13**, 54 (1974).

Newman, S. P., "Theoretical derivation of Darcy's Law," *Acta. Mech.*, **25**, 153 (1977).

Ng, K. M., H. T. Davis, and L. E. Scriven, "Visualization of blob mechanics in flow through porous media," *Chem. Eng. Sci.*, **33**, 1009 (1978).

Ng, K. M., and A. C. Payatakes, "Stochastic simulation of the motion, breakup and stranding of oil ganglia in water-wet granular porous media during immiscible displacement," *AIChE J.*, **26**, 419 (1980).

Patel, J. G., M. G. Hedge, and J. C. Slattery, "Further discussion of two-phase flow in porous media," *AIChE J.*, **18**, 1062 (1972).

Pathak, P., P. H. Winterfeld, H. T. Davis, and L. E. Scriven, "Rock structure and transport therein: unifying with Voroni models and percolation concepts," First Joint SPE/DOE Symp. on Enhanced Oil Recovery, Tulsa, OK (April 20–23, 1980).

Payatakes, A. C., K. M. Ng, and R. W. Flumerfelt, "Oil ganglion dynamics during immiscible discplacement: model formulation," *AIChE J.*, **26**, 430 (1980).

Slattery, J. C., "Flow of viscoelastic fluids through porous media," *AIChE J.*, **13**, 1066 (1967).

Slattery, J. C., "Single-phase flow through porous media," *AIChE J.*, **15**, 866 (1969).

Slattery, J. C., "Two-phase flow through porous media," *AIChE J.*, **16**, 345 (1970).

Spanos, T. J. T., "The surface conditions for viscous-displacement in a homogeneous porous medium," *Can. J. Phys.*, **57**, 1738 (1979).

Whitaker, S., "The equations of motion in porous media," *Chem. Eng. Sci.*, **21**, 291 (1966).

Whitaker, S., "Advances in the theory of fluid motion in porous media," *Ind. Eng. Chem.*, **61**, 14 (1969).

Manuscript received January 11, 1982; revision received October 5, and accepted October 20, 1982.

Simplification of Quasilinearization Method for Parameter Estimation

An alternate development of the quasilinearization method for parameter estimation is presented to enable a more efficient implementation of the algorithm. Similarity of this algorithm to Gauss-Newton method is shown and attention is given to systems having a nonlinear relationship between the observed and state variables. To overcome the problem of a small region of convergence, the use of direct search optimization is proposed for the first few iterations, followed by the simplified quasilinearization algorithm.

NICOLAS KALOGERAKIS and REIN LUUS

Department of Chemical Engineering and Applied Chemistry, University of Toronto, Toronto, Ontario, Canada

SCOPE

Off-line parameter estimation in systems described by sets of ordinary differential equations is very important in process modelling, simulation and optimization. Among the most common methods are the quasilinearization method (Bellman and Kalaba, 1965; Lee, 1968), the Gauss-Newton method (Bard, 1970, 1974), and Marquardt's modification of Gauss-Newton method (Marquardt, 1963; Bard, 1970).

Quasilinearization is best known for its fast quadratic convergence to the optimum, but a major problem is its small region of convergence (Seinfeld and Gavalas, 1970; Seinfeld and La-

pidus, 1974). Hwang and Seinfeld (1972) proposed the use of Gauss-Newton method instead of quasilinearization claiming that the former is computationally more efficient than the latter. Bard (1970) compared several gradient methods and concluded that Marquardt's modification of Gauss-Newton method is the best. Ramaker et al. (1970) used Marquardt's modification in quasilinearization to expand the region of convergence. However, they found that the efficiency of the algorithm is greatly influenced by the starting value of the directional parameter and the rate at which it is varied.

Numerous other approaches have been used to overcome the problem of the small region of convergence. Donnelly and Quon (1970) proposed a procedure whereby the given data are perturbed if divergence occurs and a series of problems are solved until the model trajectories match the original data. Nieman and Fisher (1972) incorporated linear programming into quasilinearization and suggested the solution of a series of constrained parameter estimation problems where the search is restricted in a small parameter space around the chosen initial guess. Recently, Wang and Luus (1980) proposed the use of shorter data length to substantially enlarge the region of convergence, thereby also avoiding the problem of convergence to nonglobal optima. More specifically, only portions of the data length are used during the early iterations when the estimated values for the parameters are far from their optimum values. Once the parameter values are close to the optimum, the entire data length is used to refine the final estimates.

Hwang and Seinfeld (1972) proposed to overcome the problem of multiple optima by the use of different weighting along the data length. They concluded that when changes in the weighting have no effect on the estimated parameter values, the global optimum has been reached.

Bergman et al. (1976) used quasilinearization and Gauss-Newton method to estimate parameters in a biomedical model. They found that both methods produced the same results. Xuyen and Svrcek (1977) reported that Gauss-Newton and quasilinearization methods are equivalent.

In this work a somewhat different development of the quasilinearization method is presented and the computational requirements are examined. It is found that the integration of the particular solution is redundant and, therefore, unnecessary. It is shown that both quasilinearization and Gauss-Newton method yield identical results, unless there is a nonlinear relationship between the observed and state variables. In this case, quasilinearization behaves differently when at the end of each iteration the new parameter estimates are obtained by solving the resulting set of nonlinear algebraic equations. Finally, as an alternative to using shorter data length to enlarge the region of convergence, we propose the use of direct search optimization for the first few iterations to obtain a sufficiently good estimate for the parameters to serve as a starting point for the simplified quasilinearization algorithm.

CONCLUSIONS AND SIGNIFICANCE

By means of a simple transformation a redundant set of differential equations is removed from the quasilinearization method. This simplified quasilinearization method is shown to be very similar to the Gauss-Newton method. Both algorithms require the same number of differential equations to be integrated at each iteration and produce the same new estimate of the unknown parameter vector when there is a linear relationship between the observed and state variables. In the case of a nonlinear observational relationship, linearization of the output was found to be not only easier to program and computationally faster, but also yielded a larger region of convergence than direct substitution of the nonlinear output relationship into the performance index. When prior information about the values of the unknown parameters is unavailable, the use of direct search optimization to give a good initial guess appears to be a practical and convenient approach. The computational effort required by the LJ optimization procedure (Luus and Jaakola, 1973) compares favorably to that used by the simplified quasilinearization method and only a few iterations are required to yield sufficiently good values for the parameters to ensure convergence by the quasilinearization method.

PROBLEM STATEMENT

Let us consider a dynamic system described by

$$\frac{dx(t)}{dt} = f(x(t),k) \tag{1}$$

$$x(t_o) = x_o \tag{2}$$

where x is the n-dimensional state vector, x_o the given initial state and k a p-dimensional vector of constant parameters. The m-dimensional output vector $y(t)$ is related to the state vector by an explicit relationship

$$y(t) = h(t,x(t)) \tag{3a}$$

In many engineering applications Eq. 3a assumes the linear form

$$y(t) = C(t)x(t) \tag{3b}$$

where $C(t)$ is the $m \times n$ observation matrix. Although C can change with time, for most situations it is a constant matrix.

Given the measurements of the output vector, $\hat{y}(t_i)$, $i = 1,2,\ldots,N$, the problem is to estimate the unknown parameter vector k, which minimizes the weighted sum of squares of deviations

$$S = \sum_{i=1}^{N} [\hat{y}(t_i) - y(t_i)]^T Q(t_i)[\hat{y}(t_i) - y(t_i)] \tag{4}$$

where $Q(t_i)$ is an $m \times m$ positive definite, symmetric weighting matrix and, in the general case, can have different values along the data length. We assume that f in Eq. 1 has continuous first partial derivatives with respect to its arguments.

QUASILINEARIZATION METHOD

Suppose we have an estimate $k^{(j)}$ of the unknown parameter vector at the jth iteration; then Eq. 1 becomes

$$\frac{dx^{(j)}(t)}{dt} = f[x^{(j)}(t),k^{(j)}] \tag{5}$$

At the next iteration with the parameter vector $k^{(j+1)}$ Eq. 1 becomes

$$\frac{dx^{(j+1)}(t)}{dt} = f[x^{(j+1)}(t),k^{(j+1)}] \tag{6}$$

By using Taylor series expansion on the right hand side of Eq. 6 and retaining only the linear terms, we obtain

$$\frac{dx^{(j+1)}(t)}{dt} = f[x^{(j)}(t),k^{(j)}] + \left(\frac{\partial f^T}{\partial x}\right)^T [x^{(j+1)}(t) - x^{(j)}(t)] + \left(\frac{\partial f^T}{\partial k}\right)^T [k^{(j+1)} - k^{(j)}] \quad (7)$$

where the partial derivatives are evaluated at $x^{(j)}(t)$.

Equation 7 is linear in $x^{(j+1)}(t)$ and $k^{(j+1)}$ and upon integration will have the structure

$$x^{(j+1)}(t) = g(t) + G(t)k^{(j+1)} \quad (8)$$

where $g(t)$ is an n-dimensional vector and $G(t)$ is an $n \times p$ matrix. Differentiating Eq. 8 with respect to t and equating the resulting equation to Eq. 7 yields

$$\frac{dg(t)}{dt} = f[x^{(j)}(t),k^{(j)}] + \left(\frac{\partial f^T}{\partial x}\right)^T [g(t) - x^{(j)}(t)] - \left(\frac{\partial f^T}{\partial k}\right)^T k^{(j)} \quad (9)$$

and

$$\frac{dG(t)}{dt} = \left(\frac{\partial f^T}{\partial x}\right)^T G(t) + \left(\frac{\partial f^T}{\partial k}\right)^T \quad (10)$$

Since the initial state is given, it follows from Eq. 8 that the initial conditions for Eqs. 9 and 10 are $g(t_o) = x_o$ and $G(t_o) = 0$ respectively.

Thus, given the initial state x_o and the parameter vector $k^{(j)}$, Eqs. 5, 9 and 10 can be solved simultaneously to yield $g(t)$ and $G(t)$. With $k^{(j+1)}$, yet to be determined, Eq. 8 gives an explicit expression for $x^{(j+1)}(t)$. When the output vector is related to the state vector by Eq. 3b, Eq. 4 becomes

$$S = \sum_{i=1}^{N} [\hat{y}(t_i) - C(t_i)(g(t_i) + G(t_i)k^{(j+1)})]^T \times Q(t_i)[\hat{y}(t_i) - C(t_i)(g(t_i) + G(t_i)k^{(j+1)})] \quad (11)$$

We now choose $k^{(j+1)}$ by minimizing this expression by considering the stationary condition $\partial S/\partial k^{(j+1)} = 0$ which yields

$$\left[\sum_{i=1}^{N} G^T(t_i)C^T(t_i)Q(t_i)C(t_i)G(t_i)\right]k^{(j+1)} = \sum_{i=1}^{N} G^T(t_i)C^T(t_i)Q(t_i)[\hat{y}(t_i) - C(t_i)g(t_i)] \quad (12)$$

Since $Q(t_i)$ is positive definite, Eq. 12 yields the minimum of S. Equation 12 is a linear equation and can be solved by Gaussian elimination to yield $k^{(j+1)}$.

Since linearization has been used, $k^{(j+1)}$ obtained by solving Eq. 12 is an approximation to the best parameter vector. Therefore, with this value as $k^{(j)}$ the above procedure can be repeated to yield another $k^{(j+1)}$ and thus a sequence of parameter vectors can be obtained. It is found that the sequence $k^{(0)}, k^{(1)}, k^{(2)} \ldots$ converges rapidly to the optimum if the initial guess $k^{(0)}$ is sufficiently good. This procedure constitutes the well-known quasilinearization algorithm which is based on the linearization of the differential equation, Eq. 1, around the trajectory $x^{(j)}(t)$, resulting from the choice of $k^{(j)}$. The total number of differential equations that are integrated at each iteration is $n(p + 2)$.

We observe, however, that Eq. 9 is really redundant since $g(t)$ can be obtained by alternate means. Since Eq. 8 is the result of linearization around the nominal trajectory $x^{(j)}(t)$ resulting from $k^{(j)}$, if we let $k^{(j+1)}$ be $k^{(j)}$, the LHS of Eq. 8, $x^{(j+1)}(t)$, becomes $x^{(j)}(t)$ and $g(t)$ is obtained immediately through rearrangement; i.e.,

$$g(t) = x^{(j)}(t) - G(t)k^{(j)} \quad (13)$$

The use of Eq. 13 leads to a simplification, since the number of differential equations to be integrated at each iteration is reduced to $n(p + 1)$. The algorithm of this simplified quasilinearization method is thus the following:

(1) Select an initial guess $k^{(j)}$ and put the iteration index $j = 0$.
(2) Integrate Eqs. 5 and 10 simultaneously to obtain $x^{(j)}(t)$ and $G(t)$.
(3) Obtain $g(t_i)$, $i = 1,2,\ldots,N$ from Eq. 13.
(4) Obtain $k^{(j+1)}$ from Eq. 12.
(5) If $\|k^{(j+1)} - k^{(j)}\|$ is less than a preset criterion terminate the iteration; otherwise set $k^{(j)} = k^{(j+1)}$, increment j by one and go back to step (2).

We shall now proceed to show that there is a direct relationship of this algorithm to Gauss-Newton method.

GAUSS-NEWTON METHOD

In the development of the Gauss-Newton method, linearization with respect to $k^{(j)}$ is performed on the output vector $y(t)$, rather than the differential equation. Therefore, we write the linearized output vector

$$y^{(j+1)}(t) = y^{(j)}(t) + \left(\frac{\partial y^T}{\partial k}\right)^T (k^{(j+1)} - k^{(j)}) \quad (14)$$

For the case of Eq. 3b, Eq. 14 yields

$$y^{(j+1)}(t) = C(t)x^{(j)}(t) + C(t)\left(\frac{\partial x^T}{\partial k}\right)(k^{(j+1)} - k^{(j)}) \quad (15)$$

Here $(\partial x^T/\partial k)^T$ is the $n \times p$ sensitivity matrix. To obtain the sensitivity matrix we differentiate both sides of Eq. 1 with respect to k and change the order of differentiation to yield

$$\frac{d}{dt}\left(\frac{\partial x^T}{\partial k}\right)^T = \left(\frac{\partial f^T}{\partial x}\right)^T\left(\frac{\partial x^T}{\partial k}\right)^T + \left(\frac{\partial f^T}{\partial k}\right)^T \quad (16)$$

Since the initial state x_o is independent of the parameter vector k, it follows that

$$\left(\frac{\partial x^T}{\partial k}\right)^T = 0 \text{ at } t = t_o \quad (17)$$

Inspecting Eqs. 16 and 10, it is obvious that both $G(t)$ and $(\partial x^T/\partial k)^T$ satisfy the same differential equation and have the same initial condition. Therefore

$$\left(\frac{\partial x^T}{\partial k}\right)^T = G(t) \quad (18)$$

Thus, Eq. 15 can be rewritten as

$$y^{(j+1)}(t) = C(t)x^{(j)}(t) + C(t)G(t)(k^{(j+1)} - k^{(j)}) \quad (19)$$

Substituting $y^{(j+1)}(t)$ into the performance index and setting $\partial S/\partial k^{(j+1)} = 0$ we obtain a set of linear algebraic equations

$$\left[\sum_{i=1}^{N} G^T(t_i)C^T(t_i)Q(t_i)C(t_i)G(t_i)\right](k^{(j+1)} - k^{(j)}) = \sum_{i=1}^{N} G^T(t_i)C^T(t_i)Q(t_i)(\hat{y}(t_i) - C(t_i)x^{(j)}(t_i)) \quad (20)$$

Equation 20 is then solved by Gaussian elimination to yield $k^{(j+1)} - k^{(j)}$ and thus $k^{(j+1)}$ is obtained.

As is seen, the only difference between the Gauss-Newton and the quasilinearization method appears to be the nature of the equation yielding $k^{(j+1)}$. Let us substitute Eq. 13 into Eq. 12 to yield

$$\left[\sum_{i=1}^{N} G^T(t_i)C^T(t_i)Q(t_i)C(t_i)G(t_i)\right]k^{(j+1)} = \sum_{i=1}^{N} G^T(t_i)C^T(t_i) \times Q(t_i)[\hat{y}(t_i) - C(t_i)x^{(j)}(t_i) + C(t_i)G(t_i)k^{(j)}] \quad (21)$$

and by taking the last term on the right hand side of Eq. 21 to the left side, Eq. 20 is obtained. Thus the methods are computationally identical if the output is linearly related to the state.

NONLINEAR OBSERVATION RELATIONSHIP

Let us now consider the situation when the output vector is related to the state vector through the nonlinear relationship given by Eq. 3a. Substituting $x^{(j+1)}$ from Eq. 8 into Eq. 3a and the resulting equation into the performance index, the stationary condition $\partial S/\partial k^{(j+1)} = 0$ yields

$$\sum_{i=1}^{N} G^T(t_i)\left(\frac{\partial h^T(t_i, x^{(j+1)}(t_i))}{\partial x^{(j+1)}(t_i)}\right) \times Q(t_i)\,(\hat{y}(t_i) - h(t_i, x^{(j+1)}(t_i))) = 0 \quad (22)$$

Equation 22 represents a set of p nonlinear algebraic equations for $k^{(j+1)}$. These nonlinear algebraic equations can be solved either directly (Seinfeld and Lapidus, 1974) or by means of linearization of the output vector around the trajectory $x^{(j)}(t)$ (Emig and Köppner, 1974; Xuyen and Svrcek, 1977) whereby Eq. 22 is reduced to a set of p linear algebraic equations followed by solution of these linear equations. In the first case, Eq. 22 can be solved numerically by Newton's method, for which as an initial guess for $k^{(j+1)}$ we use the previous value, i.e., $k^{(j)}$. If $k^{(j)}$ is close to the optimum, $k^{(j+1)}$ will not differ much from $k^{(j)}$, and therefore convergence problems for the Newton's method are not expected. However, when $k^{(j+1)}$ is far from $k^{(j)}$, the use of $k^{(j)}$ as an initial guess may cause difficulties, since the root-finding algorithm may diverge, or it may converge to another value which does not correspond to the minimum of the performance index.

If on the other hand, we choose to linearize the output vector around $x^{(j)}(t)$, such difficulties do not arise. Linearization of Eq. 3a around $x^{(j)}(t)$ gives

$$h(t, x^{(j+1)}(t)) = h(t, x^{(j)}(t)) + \left(\frac{\partial h^T}{\partial x}\right)^T (x^{(j+1)}(t) - x^{(j)}(t)) \quad (23)$$

where the partial derivative is evaluated at $x^{(j)}(t)$. By substituting Eqs. 8 and 13 into Eq. 23 we get

$$h(t, x^{(j+1)}(t)) = h(t, x^{(j)}(t)) + \left(\frac{\partial h^T}{\partial x}\right)^T G(t)[k^{(j+1)} - k^{(j)}] \quad (24)$$

and thus Eq. 22 is reduced to the linear equation

$$\left[\sum_{i=1}^{N} G^T(t_i)\left(\frac{\partial h^T}{\partial x}\right) Q(t_i)\left(\frac{\partial h^T}{\partial x}\right)^T G(t_i)\right](k^{(j+1)} - k^{(j)}) = \sum_{i=1}^{N} G^T(t_i)\left(\frac{\partial h^T}{\partial x}\right) Q(t_i)(\hat{y}(t_i) - h(t_i, x^{(j)}(t_i))) \quad (25)$$

By using the Gauss-Newton method such options do not arise, since the method requires the linearization of the output vector around $k^{(j)}$. Thus, we have

$$h(t, x^{(j+1)}(t)) = h(t, x^{(j)}(t)) + \left(\frac{\partial h^T}{\partial x}\right)^T \left(\frac{\partial x^T}{\partial k}\right)^T (k^{(j+1)} - k^{(j)}) \quad (26)$$

Substitution of $h(t, x^{(j+1)}(t))$ into the performance index and the use of the stationary condition $\partial S / \partial k^{(j+1)} = 0$, yields Eq. 25. Therefore, when we linearize the output vector, quasilinearization and Gauss-Newton method yield the same results.

The estimate that is obtained by solving numerically Eq. 22 is different from that obtained by Eq. 25. In fact one should expect to obtain a better value for k from the latter, since it represents more accurately the dependence of the performance index on $k^{(j+1)}$. Therefore, one should expect the output linearization approach to require a larger number of iterations to converge to the optimum. However, the programming effort is less and any difficulties associated with the root-finding procedure are avoided. A numerical comparison of the computational effort by these two approaches is given in Example 2.

PROCEDURE FOR GOOD INITIAL ESTIMATES

A good initial guess for the parameter vector is usually required to ensure convergence of the quasilinearization or Gauss-Newton method. A considerable amount of work has been directed to solve the problem of the small region of convergence or the problem of convergence to nonglobal optima (Ramaker et al., 1970; Hwang and Seinfeld, 1972; Donnelly and Quon, 1970; Nieman and Fisher, 1972).

Recently, Wang and Luus (1980) proposed the use of shorter data length to substantially enlarge the region of convergence to the global optimum. Although a significant increase in the size of the region of convergence is achieved, there may still be occasions where the region of convergence is too small with respect to the range of knowledge of the unknown parameters. In such cases one must resort to some preliminary work to establish a sufficiently good initial guess before the use of the quasilinearization or Gauss-Newton method.

Direct search optimization provides a convenient means of obtaining a good initial guess for the parameter vector. The procedure we have found to be most reliable is the LJ optimization procedure (Luus and Jaakola, 1973) which uses random search points and region contraction. The method is easy to program and handles the problem of multiple optima with high reliability (Wang and Luus, 1977, 1978). In most cases only a few iterations are required to arrive sufficiently close to the global minimum to ensure convergence with the quasilinearization or Gauss-Newton method. Furthermore, the use of only a section of the data keeps the computation time reasonably small, even though a larger number of function evaluations are done at each iteration. The listing of the LJ optimization program has been given by Jaakola and Luus (1974), and therefore, no detailed comments about the minimization procedure are necessary. It is noted, however, that it is unnecessary to integrate the equations for the entire portion of the data chosen for each set of parameter values. Once the performance index becomes greater than the smallest performance index found up to that point, a new set of parameter values can be chosen.

As shown by Wang and Luus (1978) for the LJ optimization procedure, the size of the initial search region is not very important provided that it is chosen to be adequately large. Constraints are frequently imposed on the parameters from physical considerations, so that the initial search region may be chosen to cover the entire feasible range, or simply taken to be of reasonable size. In the following numerical examples the initial search region of a parameter is taken to be its initial value, unless otherwise specified.

Throughout, one hundred random points are taken at each iteration, the search region is reduced by 5% after each iteration and a maximum of 20 iterations are used. As will be shown with numerical examples, the computational effort is about the same as required by the quasilinearization or Gauss-Newton method.

NUMERICAL RESULTS

Example 1

To compare the simplified quasilinearization to the standard version and to the Gauss-Newton method, we choose the model of an isothermal CSTR with complex reactions used by Lapidus and Luus (1967) and Rao and Luus (1972) for optimal control studies. The governing equations are

$$\begin{aligned}
\frac{dx_1}{dt} &= k_5 - qx_1 - k_1x_1x_2 - k_4x_1x_6\sqrt{0.9}; & x_1(0) &= 0.1883\\
\frac{dx_2}{dt} &= 7.0 - qx_2 - k_1x_1x_2 - 2k_2x_2x_3; & x_2(0) &= 0.2507\\
\frac{dx_3}{dt} &= 1.75 - qx_3 - k_2x_2x_3; & x_3(0) &= 0.0467\\
\frac{dx_4}{dt} &= -qx_4 + 2k_1x_1x_2 - k_3x_4x_5; & x_4(0) &= 0.0899\\
\frac{dx_5}{dt} &= -qx_5 + 3k_2x_2x_3 - k_3x_4x_5; & x_5(0) &= 0.1804\\
\frac{dx_6}{dt} &= -qx_6 + 2k_3x_4x_5 - k_4x_1x_6\sqrt{0.9}; & x_6(0) &= 0.1394\\
\frac{dx_7}{dt} &= -qx_7 + 2k_4x_1x_6\sqrt{0.9}; & x_7(0) &= 0.1406\\
q &= 8.75 + k_5
\end{aligned} \quad (27)$$

TABLE 1. COMPARISON OF COMPUTATIONAL EFFORT AND REGION OF CONVERGENCE ALONG THE DIRECTION (1,1,1,1,1) IN THE PARAMETER SPACE FOR EXAMPLE 1*

Data Length Used, τ	Initial Guess $k_i^{(o)}$, $i = 1,2,\ldots 5$	Iterations	cpu Time (s) Gauss-Newton	Simplified Quasilinearization	Standard Quasilinearization
0.20	10	5	2.94	2.94	3.24
	15	7	3.12	3.12	3.54
	16	Failed	—	—	—
0.15	15	7	2.64	2.64	3.06
	16	7	2.70	2.70	3.06
	17	Failed	—	—	—
0.10	16	6	2.16	2.16	2.40
	17	7	2.28	2.28	2.52
	18	15(local)	3.18	3.24	3.72
0.05	18	5	1.74	1.74	1.86
	20	6	1.80	1.86	1.92
	23	7	1.86	1.86	1.98
	24	15(local)	2.34	2.40	2.64

* Computations for all examples were carried out on an IBM/3033 using J. H. Verner's Runge-Kutta formulae of 5th and 6th order (flexible step size) for integration.

The output vector is related to the state variables by

$$y(t) = Cx(t) \tag{28}$$

where

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

which means that we observe only four out of the seven state variables, in particular x_1, x_4, x_5 and x_6. The problem is to determine the five unknown parameters given the measurements

$$\hat{y}(t_i), \quad t_i = 0.01i, \quad i = 1,2,\ldots,20$$

The values of the unknown parameters are $k_1 = 17.6$, $k_2 = 73.0$, $k_3 = 51.3$, $k_4 = 23.0$ and $k_5 = 6.0$.

The data were generated from

$$\hat{y}(t_i) = [I + \sigma R_i]Cx^*(t_i), \quad i = 1,2,\ldots,20 \tag{29}$$

where R_i are 4×4 diagonal matrices of random numbers uniformly distributed from -1 to 1, σ is the maximum noise level and $x^*(t_i)$ the exact solution of Eq. 27, obtained using the true parameter values. A maximum noise level of 0.01 was used here.

The relatively high number of state variables and unknown parameters makes this model a good example for testing the effectiveness of the proposed method to obtain initial guesses. Furthermore, we can obtain a good quantitative idea of the difference in the computational effort required by the standard quasilinearization and Gauss-Newton method, as well as the proposed simplified quasilinearization method.

In Table 1 the required computer time by each method is shown for a set of different runs. As expected, all three methods produced the same new estimates at each iteration. On the average, the Gauss-Newton method required 11% less computer time than the standard quasilinearization, since 42 instead of 49 differential equations were integrated during each iteration. The simplified quasilinearization method required practically the same computer time as the Gauss-Newton method, since the only additional arithmetic operation was the evaluation of $g(t)$ from Eq. 13.

The determination of the size of the region of convergence in the five-dimensional parameter space requires an excessive amount of computational work. However, one can obtain a good idea of the size, looking at specific directions in the parameter space. For the sake of simplicity, in this example we estimated the size of the region of convergence along the direction (1,1,1,1,1) in the parameter space. The results are shown in Table 1 using several data lengths.

Suppose now that the only information we have is that all parameters are positive and less than 200. Using the LJ optimization procedure with starting values $k_i = 100$, $i = 1,2,\ldots,5$ and initial search regions of 100, we obtain after 3 iterations $k = [19., 78., 90., 115., 4.5]^T$, which corresponds to a reduction in the performance index from 4.008 to 0.076. The required computer time for the three iterations is 1.32 sec. A further reduction of the performance index to 0.034 is obtained after 7 iterations requiring only 1.68 s of cpu time. Nevertheless, only one iteration of the LJ optimization

TABLE 2. TYPICAL RESULTS OF THE LJ OPTIMIZATION PROCEDURE FOR EXAMPLE 1*

Run	Initial Guess	Initial Perf. Index	Iteration	Final Perf. Index	cpu Time (s)
1	$k_i = 100$		1	0.130	1.26
	$i = 1,2,\ldots,5$	4.008	3	0.076	1.32
			7	0.034	1.68
2	$k_i = 200$		1	0.080	1.30
	$i = 1,2,\ldots,5$	5.065	3	0.036	1.44
			6	0.013	1.68
3	$k_i = 300$		1	0.126	1.56
	$i = 1,2,\ldots,5$	5.469	3	0.054	1.86
4	$k_i = 400$	5.668	1	0.252	1.60
	$i = 1,2,\ldots,5$		3	0.096	1.86
5	$k_i = 500$	5.782	1	0.430	1.66
	$i = 1,2,\ldots,5$		3	0.180	1.89
			13	0.091	3.06
			16	0.057	3.36
			20	0.047	3.60

* Constraints: $k_i > 0$, $i = 1,2,\ldots,5$.
Data length used: $\tau = 0.10$.

procedure was sufficient to result in parameter values which fall inside the region of convergence of the quasilinearization method.

In Table 2 the computational effort and the achieved reduction in the performance index by the LJ optimization procedure are shown, for several runs with starting parameter values as high as 500. In all cases, at most three iterations were necessary to bring the parameters inside the region of convergence of the quasilinearization method, requiring less than 2 s of computer time. Even after 20 iterations starting with parameter values $k_i = 500$, $i = 1,2,\ldots,5$, the computation time was only 3.60 s.

Comparing the computer time reported in Tables 1 and 2, it is seen that the required computational effort for a few iterations of the LJ optimization procedure compares favourably with that required by the Gauss-Newton method. Therefore, the proposed method to obtain good initial guesses is a practical, effective and computationally attractive approach, even when the number of state variables and unknown parameters is large.

Example 2

The simple one parameter model used by Hwang and Seinfeld (1972) and Wang and Luus (1980) which demonstrates the difficulty of obtaining the global minimum is used here. The system is given by

$$\frac{dx_1}{dt} = x_2; \qquad x_1(0) = 0$$
$$\frac{dx_2}{dt} = -kx_1; \quad x_2(0) = \pi \tag{30}$$

However, in the present work we assume that there is a nonlinear relationship between the observed and state variables given by

$$y(t) = x_1(t)x_2(t) \tag{31}$$

The problem here is to determine the unknown parameter k, given the measurements

$$\hat{y}(t_i), \quad t_i = 0.01i, \quad i = 1,2\ldots,100$$

The true value of the parameter k is $\pi^2 = 9.8696044$. The data were generated from

$$\hat{y}(t_i) = x_1^*(t_i)x_2^*(t_i)[1 + \sigma r_i], \quad i = 1,2\ldots,100 \tag{32}$$

where r_i are random numbers uniformly distributed between -1 and 1, σ is the maximum noise level and $x_j^*(t_i)$, $j = 1,2$ is the exact solution of Eq. 30, obtained using $k = 9.8696044$. A maximum noise level of 0.02 was used.

In this example the output $y(t)$ is related to the state variables $x_1(t)$ and $x_2(t)$ through a nonlinear relationship. There are two approaches that may be used to solve the parameter estimation problem by quasilinearization. We can either linearize the output (output linearization approach) or solve the resulting nonlinear algebraic equation (nonlinear output approach).

First we examine the speed of convergence and the required computational effort by these two procedures. As shown in Table 3, the nonlinear output approach requires fewer number of iterations to converge as anticipated. However, it is computationally slightly slower, since the iterative solution of a nonlinear algebraic equation is required. Therefore, from the computational point of view, the algorithm based on output linearization is more attractive.

TABLE 3. SPEED OF CONVERGENCE AND COMPUTATIONAL EFFORT OF LINEAR AND NONLINEAR APPROACHES FOR EXAMPLE 2

Data Length Used, τ	Initial Guess $k^{(o)}$	Linear Approach		Nonlinear Approach	
		Iterations	cpu Time (s)	Iterations	cpu Time (s)
1.00	25	8	1.20	7	1.26
0.30	25	5	0.90	4	0.96
0.30	35	6	0.90	5	0.96
0.20	65	6	0.84	5	0.96

TABLE 4. REGION OF CONVERGENCE OF LINEAR AND NONLINEAR APPROACHES FOR EXAMPLE 2

Data Length Used, τ	Maximum Value of $k^{(o)}$ Converged	
	Linear Approach	Nonlinear Approach
1.00	25	25
0.50	50	35 (55)
0.30	90	35 (140)
0.25	125	45 (170)
0.23	400	50 (500)
0.20	700	65 (800)

We shall now proceed to examine the size of the region of convergence for these two approaches. As shown in Table 4, when a data-length τ greater than 0.50 is used, both algorithms give practically the same results. However, when τ is decreased to 0.25 or less the output linearization approach has a significantly larger region of convergence. In the other approach the use of $\boldsymbol{k}^{(j)}$ as an initial guess for $\boldsymbol{k}^{(j+1)}$ in the numerical solution of Eq. 22 was unsuccessful when the starting value of the quasilinearization method was far from the optimum. In this case, the stationary condition, a third degree polynomial with respect to $\boldsymbol{k}^{(j+1)}$, has three real roots and Newton's method converged to a root which corresponded to a local maximum. This was easily verified by the sign of the second derivative of the performance index. In Table 4 the numbers shown in parentheses represent the maximum $k^{(o)}$ for which quasilinearization method converged, when the above problem was overcome using different starting values in the numerical solution of Eq. 22. Although this was done rather easily for the given example, in a more complicated problem where there are two or more unknown parameters, it becomes a tedious and time consuming task.

Despite the simplicity of this example, the above comparisons indicate clearly the problems that may arise and the advantages of linearizing the output when there is a nonlinear relationship between the observed and state variables.

Let us now see how effective the proposed use of direct search can be when we have no prior knowledge of the value of the unknown parameter. In Table 5 the results of several runs are shown where we assumed as initial guess values of k as high as 40,000. Using a few iterations of the LJ optimization procedure we were able to arrive at parameter values which are very close to the global minimum. In all cases at most two iterations were required so that the obtained parameter value falls inside the region of convergence of the quasilinearization method, provided that a small section of the data is used initially. The required computer time for two iterations of the LJ optimization procedure was kept below 2.5 s. Comparing the figures shown in Tables 3 and 5, the computation time required by the simplified quasilinearization method to converge is somewhat less. It must be realized, however, that in the

TABLE 5. TYPICAL RESULTS OF THE LJ OPTIMIZATION PROCEDURE FOR EXAMPLE 2*

Starting Value of k	Iteration Number	Obtained k	cpu Time (s)
1,000	1	10.0	0.96
2,000	1	20.0	1.08
	2	4.8	1.20
	7	8.7	1.62
5,000	1	50.0	1.26
	2	12.0	1.32
10,000	1	100.0	1.38
	2	24.0	1.56
	5	6.9	1.80
20,000	1	600.0	1.68
	2	106.0	2.04
	7	8.0	3.00
40,000	1	4880.0	1.86
	2	92.0	2.46
	4	23.4	3.06

* Constraints: $k > 0$.
Data length used: $\tau = 0.20$.

above comparisons the direct search method was applied to parameter values very far from the optimum, whereas quasilinearization was applied to values of k inside the region of convergence. In addition, one may have to use several starting points before quasilinearization method converges to the global optimum.

We conclude that the computational effort required for a few iterations of the LJ optimization procedure is quite small, while providing, in a very simple way, sufficiently good initial guesses for quasilinearization or Gauss-Newton method.

ACKNOWLEDGMENT

This research was supported by Natural Sciences and Engineering Research Council Grant A-3515. The computations were done with the facilities of the University of Toronto Computer Centre.

NOTATION

C = $(m \times n)$ observation matrix
f = n-dimensional vector function
g = n-dimensional vector defined by Eq. 9
G = $(n \times p)$ sensitivity matrix defined by Eq. 10
h = m-dimensional vector function
k = p-dimensional parameter vector
Q = $(m \times m)$ positive-definite, symmetric weighting matrix
r = random number uniformly distributed from -1 to 1
R = (4×4) diagonal matrix of random numbers uniformly distributed from -1 to 1
S = Performance index defined by Eq. 4
t = time
x = n-dimensional state vector
y = m-dimensional output vector

Greek Letters

σ = maximum noise level
τ = data length used for parameter estimation

Superscripts

$\hat{}$ = measured value of the variable
$*$ = value of the variable at the optimum
(j) = value of the variable at the jth iteration
T = transpose

LITERATURE CITED

Bard, Y., "Comparison of Gradient Methods for the Solution of Nonlinear Parameter Estimation Problems," *SIAM J. Numer. Anal.*, **7**, 157 (1970).

Bard, Y., *Nonlinear Parameter Estimation*, Academic Press, New York (1974).

Bellman, R., and R. Kalaba, *Quasilinearization and Nonlinear Boundary Value Problems*, American Elsevier, New York, (1965).

Bergman, R. N., R. E. Kalaba, and K. Spingarn, "Optimizing Inputs for Diagnosis of Diabetes. I. Fitting a Minimal Model to Data," *J. Optimization Theory Applic.*, **20**, 47 (1976).

Donnelly, J. K., and D. Quon, "Identification of Parameters in Systems of Ordinary Differential Equations Using Quasilinearization and Data Perturbation," *Can. J. Chem. Eng.*, **48**, 114 (1970).

Emig, G., and M. Köppner, "Die Methode der Quasilinearisierung zur Ermittlung Kinetischer Parameter," *Chem. Eng. Sci.*, **29**, 2339 (1974).

Hwang, M., and J. H. Seinfeld, "A New Algorithm for the Estimation of Parameters in Ordinary Differential Equations," *AIChE J.*, **18**, 90 (1972).

Jaakola, T. H. I., and R. Luus, "A Note on the Nonlinear Programming of Chemical Process Optimization," *Operations Research*, **122**, 415 (1974).

Lapidus, L., and R. Luus, *Optimal Control of Engineering Processes*, Blaisdell, MA (1967).

Lee, E. S., *Quasilinearization and Invariant Imbedding*, Academicc Press, New York (1968).

Luus, R., and T. H. I. Jaakola, "Optimization by Direct Search and Systematic Reduction of the Size of Search Region," *AIChE J.*, **19**, 760 (1973).

Marquardt, D. W., "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *J. Soc. Indust. Appl. Math.*, **11**, 431 (1963).

Nieman, R. E., and D. G. Fisher, "Parameter Estimation using Linear Programming and Quasilinearization," *Can. J. Chem. Eng.*, **50**, 802 (1972).

Ramaker, B. L., C. L. Smith, and P. W. Murrill, "Determination of Dynamic Model Parameters Using Quasilinearization," *Ind. Eng. Chem. Fund.*, **9**, 28 (1970).

Rao, S. N., and R. Luus, "Evaluation and Improvement of Control Vector Iteration Procedures for Optimal Control," *Can. J. Chem. Eng.*, **50**, 777 (1972).

Seinfeld, J. H., and G. R. Gavalas, "Analysis of Kinetic Parameters from Batch and Integral Reactor Experiments," *AIChE J.*, **16**, 644 (1970).

Seinfeld, J. H. and L. Lapidus, *Mathematical Methods in Chemical Engineering*, 3, Prentice Hall, Englewood Cliffs, NJ (1974).

Swartz, J., and H. Bremerman, "Discussion of Parameter Estimation in Biological Modelling: Algorithms for Estimation and Evaluation of the Estimates," *J. Math. Biology*, **1**, 241 (1975).

Wang, B. C., and R. Luus, "Optimization of Non-Unimodal Systems," *Int. J. Numer. Meth. Eng.*, **11**, 1235 (1977).

Wang, B. C., and R. Luus, "Reliability of Optimization Procedures for Obtaining Global Optimum," *AIChE J.*, **24**, 619 (1978).

Wang, B. C., and R. Luus, "Increasing the Size of Region of Convergence for Parameter Estimation Through the Use of Shorter Data-Length," *Int. J. Control*, **31**, 947 (1980).

Xuyen, V. T., and W. Y. Svrcek, "On Equivalence of the Gauss-Newton Technique, the Parameter Influence Coefficient Technique, and the Quasilinearization Technique in Dynamic System Identification by Least Squares," *J. Optimization Theory Applic.*, **22**, 117 (1977).

Manuscript received December 4, 1981; revision received April 26, and accepted September 3, 1982.