Applied Mathematical Sciences, Vol. 6, 2012, no. 34, 1667 – 1676

Solving Nonlinear Least Squares Problems with B-Spline Functions

J. Izadian [*] and N. Farahbakhsh

Department of Mathematics, Faculty of sciences, Mashad Branch,
Islamic Azad University, Mashad, Iran.
Jalal_Izadian@yahoo.com

M. Jalili

Department of Mathematics, Neyshabur Branch,
Islamic Azad University, Neyshabur, Iran.
Jalili.maryam@yahoo.com

Abstract

In this paper a new method for solving nonlinear least squares problems is presented. Ordinary least squares methods for solving these problems with linear methods, like polynomials and trigonometric least squares functions are not suitable to follow nonlinear behavior of the desired solution. Nonlinear least squares methods are dependent on initial estimation. B-spline functions are not nonlinear related to unknown parameters, but can behave like nonlinear functions and require no initial estimation. Numerical experiments show that B-spline functions, due to piecewise polynomial character and smoothness, are useful tool for solving nonlinear least squares problems.

Keywords: Least squares method; Spline functions; B-spline functions

1. Introduction

Linear least squares method was introduced by Gauss in 1877, nowadays this method is a basic tool for fitting curves and approximating functions. Linear least squares methods are suitable for linear least squares problems, in which the solution is a linear function of unknown parameters, and can be a polynomial function or polynomial trigonometric function. These functions would be good approximations for the linear least squares problems, but when the exact solution is a nonlinear function, for example an exponential function, approximation function must be the same. This case is called nonlinear least squares method.

As one knows a least squares problem is finding a function $p : R^{n+1} \to R$ which minimizes a certain function $F : R^n \to R$, on a domain $D \subset R^n$.To define F, consider a set of m data points in R^2, (x_i, y_i), $i = 1,2,3,..., m$, where x_i, $i = 1,2,3,..., m$ are distinct points in R, then p is a function of real variable x, and real parameters $a_1, a_2, a_3,..., a_n$. let $\tilde{p}_i$ is defined as follows:

$$\tilde{p}_i = p(x_i, a_1, a_2,..., a_n), \qquad i = 1,2,3,..., m,$$

Now ,suppose $f_i : R^n \to R$ are defined as follows:

$$f_i(a) = y_i - \tilde{p}_i = y_i - p(x_i, a), \qquad i = 1, 2,..., m \tag{1.1}$$

where $a = (a_1, a_2,..., a_n)$, and $m > n$. The least squares problem is minimizing Euclidean norm of vector $(f_1(a),..., f_m(a))$, or solving following problem:

$$\min_{a \in R^n} F(a) = \| f(a) \|^2$$

with

$$f(a) = (f_1(a), f_2(a),..., f_m(a)).$$

If F is differentiable, then the following equation must be solved

$$\nabla f(a) = 0 \tag{1.2}$$

in which Df is Jacobian matrix of f. If p is a linear function of parameters a_i for $i = 1,2,..., n$, the least squares problem is linear, otherwise nonlinear. Solving nonlinear problems is not simple. In particular, there isn't any exact method to determine the solution. But one can use certain method, such as Newton's method, to solve numerically the equation (1.2). Another known method for solving nonlinear least squares problems is Gauss-Newton's method. This method uses the Taylor expansion near an initial point $a^{(0)}$, for finding a new point the following equation is applied [5]:

$$((Df(a^{(k)}))^t Df(a^{(k)}))h_{gn} = -(Df(a^{(k)}))^t f(a^{(k)}) \qquad k = 0,1,2,...$$

where h_{gn}, step length, is defined by $a^{(k+1)} = a^{(k)} + h_{gn}$.

There is also the Levenberg-Marquart method, that is introduced first by Levenberg [6] in 1944, and improved by Marquart in 1963, [8]. This method is based on Guass-Newton's method, that is given by following equation:

$$(Df(a^{(k)})^t Df(a^{(k)}) + \mu I)h_{lm} = -Df(a^{(k)}) f(a^{(k)}) \qquad \mu \ge 0 \ , \quad k = 0,1,2,...$$

Another efficient method is Powell's Dog Leg method that is a combination the Guass-Newton and steepest decent method, and uses trust region to find the minimum point [9]. Hybrid method of Madsen (1988), [7] can also be used for solving the nonlinear least squares problem. Finally one can mention Levenberg-

Marquart-Feletcher method, that is introduced by Feletcher who improved Levenberg-Marquart method as follows [4]:

$$[(Df(a^{(k)})^t Df(a^{(k)})) + \lambda^{(k)} D_k]h_{lmf} = -(Df(a^{(k)})^t Df(a^{(k)})),$$

$$k=0,1,2,\ldots,$$

where h_{lmf}, step length, is defined by $a^{(k+1)} = a^{(k)} + h_{lmf}$., and $\lambda^{(k)}$ are certain real parameter and D_k a given diagonal matrix.

All these methods are suitable only when the initial point is near the exact solution for which it is necessary to have a certain information of behavior of exact solution, that is not always the case. For solving this inconvenience, we introduce a linear least square method which can behave as nonlinear function, by using B-spline functions.

This paper is organized in 4 sections. After introduction, in section 2, the least squares B-spline method for solving nonlinear least squares problems is introduced . In section 3, the numerical results are presented, and in section 4, we give the conclusion .

2. B-spline least squares method.

The third degree B-spline, as an interpolation function on an interval, that consists five knots $x_0 < x_1 < x_2 < x_3 < x_4$, is considered, for which $f_i = f(x_i)$, $i = 0,1,2,3,4,$ are given known real values. A third degree piecewise polynomial function S is defined by polynomials S_j on intervals $[x_j, x_{j+1}]$, $j = 0,1,2,3$. The following conditions are imposed:

1. $S(x_j) = f(x_j)$, $j = 0,1,2,3,4$,
2. $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$, $j = 0,1,2$,
3. $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$, $j = 0,1,2$,
4. $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$, $j = 0,1,2$,
5. Only one of the conditions (i) or (ii) is satisfied:
   i) $S''(x_0) = S''(x_4) = 0$,
   ii) $S'(x_4) = f'(x_4)$, and $S'(x_0) = f'(x_0)$.

For determining S with 16 unknown coefficients, 16 conditions are needed. 14 conditions from (1-4) are chosen and only one pair of conditions (5) can be selected. Since the cubic spline functions are twice differentiable in $[x_0, x_4]$, one of two conditions of interpolation can be eliminated, Due to this fact, the following conditions, instead of four of some previous conditions, are considered:

1. $S(x_j) = f(x_j)$, $j = 0,2,4,$
2. $S(x_4) = 0$, $S(x_2) = 1$, and $S(x_0) = 0$,
3. $S''(x_0) = S''(x_4) = 0$,

4. $S'(x_0) = S'(x_4) = 0$.

now one has 16 equations and 16 unknowns. So a B-spline interpolation function of the third degree can be determined by solving these simultaneous linear equations that consist 16 equations and 16 unknowns. By repeating the same procedure, a basis B-spline function in whole interval $(-\infty, \infty)$ with $S \in C^2(-\infty, \infty)$, is determined.

Now suppose (x_i, y_i), $i = 1,2,...,m$ are known data of a curve (or function), which

$$x_0 < x_1 < x_2 < ... < x_m .$$

The problem is to approximate these data a least squares function, say, p with $p : R^{n+1} \to R$, and $\tilde{p} = R \to R$ defined by

$$y = \tilde{p}(t) = p(t, a_1, a_2, ..., a_n)$$

which minimizes the following function.

$$F(a_1, ..., a_n) = \sum_{i=1}^{m} (y_i - \tilde{p}(x_i))^2$$

where unknown parameters $a_1, a_2, ..., a_n$ must be determined. If $\tilde{p}(t)$ is linear with respect to the parameters, the problem is a linear least squares problem, otherwise it is a nonlinear least squares problem. Now, $\tilde{p}(t)$ is defined as follows:

$$\tilde{p}(t) = \sum_{i=1}^{n} a_i Q_i(t) , \tag{2.1}$$

where $Q_i(t)$, for $i = 1,2,...,n$ are basis B-spline functions . Evidently, $\tilde{p}(t)$ is linear with respect to the parameters a_i, $i = 1,2,...,n$. For a uniform partition of $t_1 < t_2 < ... < t_N$ with step length h, $Q_i(t)$ can be given as follows [3] :

$$Q_i(t) = \frac{1}{4h^3} \begin{cases} (t - t_i)^3 & t \in [t_i, t_{i+1}] \\ h^3 + 3h^2(t - t_{i+1}) + 3h(t - t_{i+1})^2 - 3(t - t_{i+1})^3 & t \in [t_{i+1}, t_{i+2}] \\ h^3 + 3h^2(t_{i+3} - t) + 3h(t_{i+3} - t)^2 - 3(t_{i+3} - t)^3 & t \in [t_{i+2}, t_{i+3}] \\ (t_{i+4} - t)^3 & t \in [t_{i+3}, t_{i+4}] \\ 0 & t \in R - [t_i, t_{i+4}] \end{cases}$$

where $h = t_{i+1} - t_i$, $i = 1,2,..., N-1$. and $n = N - 4$.

These B- spline are used in certain collocation methods for solving ODE's and PDE's [3]. By defining p with (2.1), $\tilde{p}(t)$ is written as follows:

$$
\begin{aligned}
\tilde{p}(t) &= a_i \frac{1}{4h^3}\left[(t_{i+4}-t)^3\right] \\
&+ a_{i+1}\frac{1}{4h^3}\left[h^3+3h^2(t_{i+4}-t)+3h(t_{i+4}-t)^2-3(t_{i+4}-t)^3\right] \\
&+ a_{i+2}\frac{1}{4h^3}\left[h^3+3h^2(t-t_{i+3})+3h(t-t_{i+3})^2-3(t-t_{i+3})^3\right] \\
&+ a_{i+3}\frac{1}{4h^3}\left[(t-t_{i+3})^3\right] \quad , \qquad t\in\left[t_{i+3},t_{i+4}\right], \qquad i=1,2,\ldots,n.
\end{aligned}
$$

Now, for determining $\tilde{p}$ as a least squares solution, the parameters a_i for $i=1,2,\ldots,n$ must be computed. Then the function F must be minimized with respect, $a_1,a_2,\ldots,a_n$. But (2.1) deduces:

$$F(a_1,\ldots,a_n)=\sum_{i=1}^{m}\left(y_i-\sum_{j=1}^{n}a_jQ_j(x_i)\right)^2 \tag{2.2}$$

Accepting the notations

$$y=(y_1,\ldots,y_m), \qquad q=(\tilde{p}(x_1),\ldots,\tilde{p}(x_m)),$$

the relation (2.2) can be written as:

$$F(a_1,\ldots,a_n)=\sum_{i=1}^{m}(y_i-\tilde{p}(x_i))^2=(y-q)^t(y-q)=\|y-q\|^2=\|f\|^2$$

If F has a minimum point as $\hat{a}=(\hat{a}_1,\hat{a}_2,\ldots,\hat{a}_n)$, by considering to the differentiable F, the following equation must be solved

$$\nabla F(\hat{a}_1,\ldots,\hat{a}_n)=0. \tag{2.3}$$

Then, $\hat{a}$ can be computed by solving (2.3). But $\nabla F(a)$ can be expressed as:

$$\nabla F(a_1,\ldots,a_n)=\nabla[(y-q)^t(y-q)]. \tag{2.4}$$

From (2.3) and (2.4), it follows:

$$Dq^t q = Dq^t y,$$

or

$$\begin{bmatrix} Q_1(x_1) & \cdots & Q_1(x_m) \\ \vdots & & \vdots \\ Q_n(x_1) & \cdots & Q_n(x_m) \end{bmatrix}\begin{bmatrix} \sum_{j=1}^{n}a_jQ_j(x_1) \\ \vdots \\ \sum_{j=1}^{n}a_jQ_j(x_m) \end{bmatrix}=\begin{bmatrix} \sum_{i=1}^{m}Q_1(x_i)y_i \\ \vdots \\ \sum_{i=1}^{m}Q_n(x_i)y_i \end{bmatrix},$$

which can be transformed to the following linear matrix equation:

$$\begin{bmatrix} \sum_{i=1}^{m}(Q_1(x_i))^2 & \sum_{i=1}^{m}Q_1(x_i)Q_2(x_i) & \cdots & \sum_{i=1}^{m}Q_1(x_i)Q_n(x_i) \\ \vdots & & & \\ \sum_{i=1}^{m}Q_n(x_i)Q_1(x_i) & \sum_{i=1}^{m}Q_n(x_i)Q_2(x_i) & \cdots & \sum_{i=1}^{m}(Q_n(x_i))^2 \end{bmatrix}\begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}=\begin{bmatrix} \sum_{i=1}^{m}Q_1(x_i)y_i \\ \vdots \\ \sum_{i=1}^{m}Q_n(x_i)y_i \end{bmatrix} \tag{2.5}$$

The above linear system has exactly one solution, since the columns of Dq are linearly independent, and thus Dq^tDq the coefficient matrix of system is nonsingular. By solving (2.5) we obtain $\hat{a}$, so $\hat{p}(t)$ will be determined by equation:

$$\hat{p}(t) = \sum_{i=1}^{m} \hat{a}_i Q_i(t) \quad , \quad \hat{a} = (\hat{a}_1, \ldots, \hat{a}_m) \ .$$

3. Numerical results

In this part of paper, two examples with different data to find the B-spline least squares for a polynomial trigonometric function and a exponential function are presented. B-spline method with Quasi-Newton (QN), Levenberg-Marquart (LM), Dog Leg (DL), Levenberg-Marquart-Feletcher (LMF) and hybrid method (HM) [5] are also compared.

3.1. Example

In this example, $m = 20, n = 8$, and

$x = (0, 0.05, 0.1, 0.17, 0.2, 0.22, 0.3, 0.35, 0.4, 0.48,$
$0.5, 0.54, 0.6, 0.69, 0.7, 0.71, 0.8, 0.84, 0.9, 1).$

$y = (0, 0.055, 0.089, 0.135, 0.185, 0.225, 0.27, 0.3, 0.33, 0.35,$
$0.362, 0.355, 0.365, 0.36, 0.34, 0.321, 0.292, 0.228, 0.185, 0.12),$

$t = (-0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6),$

The optimal value of a is

$\hat{a} = (-0.0561, -0.0144, 0.01315, 0.2333, 0.2494, 0.2142, 0.0160, 0.2042)$,

optimal value of $\|f\|$ is 0.0452, and CPU-Time is 2.148313 seconds. The data, the B-spline and the error function graphs, are presented in Fig.1.

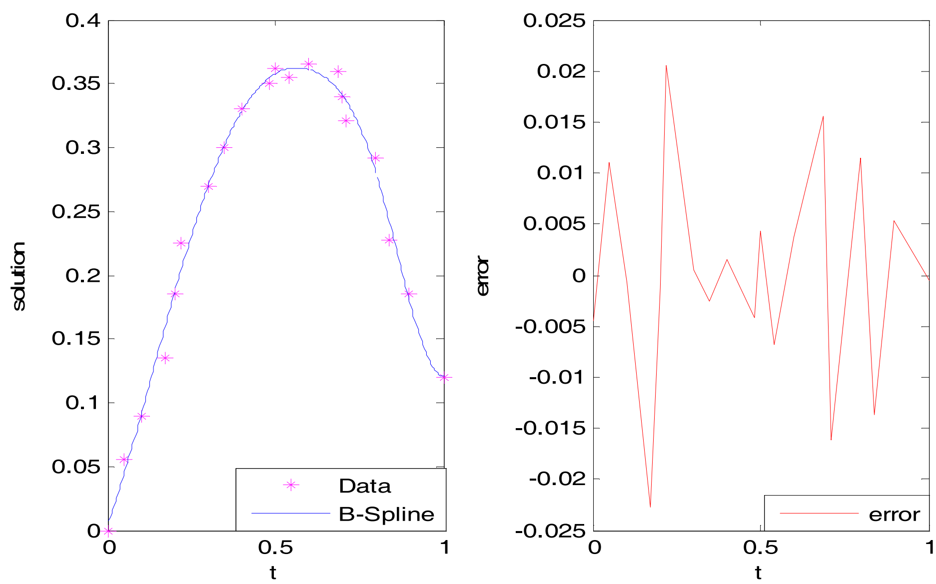

Fig.1. rigth, graph of F (error), left, data points and B-spline function graph.

3.2. Example

In this example, $m = 20, n = 8$, and

$$x = (0,\ 0.05,0.1,0.17,0.2,0.22,0.3,0.35,0.4,0.48,\\ 0.5,0.54,0.6,0.69,0.7,0.71,0.8,0.84,0.9,1).$$

$$y = (0.09,0.18,0.35,0.44,0.52,0.58,0.79,0.82,0.99,0.95,1.02,\\ 1.05,1.11,1.19,1.2,1.09,1.04,1.15,1.13,1.08),$$

$$t = (-0.6,-0.4,-0.2,0,0.2,0.4,0.6,0.8,1,1.2,1.4,1.6).$$

The optimal value of a is

$$\hat{a} = (-0.1768, 0.0400, 0.3703, 0.6383, 0.7542, 0.7610, 0.7245, 0.6708),$$

optimal value of $\|f\|$ is 0.1863, and CPU-Time is 1.846861 seconds. The data, the graphs of B-spline solution, and error $\|f\|$ are demonstrated in Fig.2.

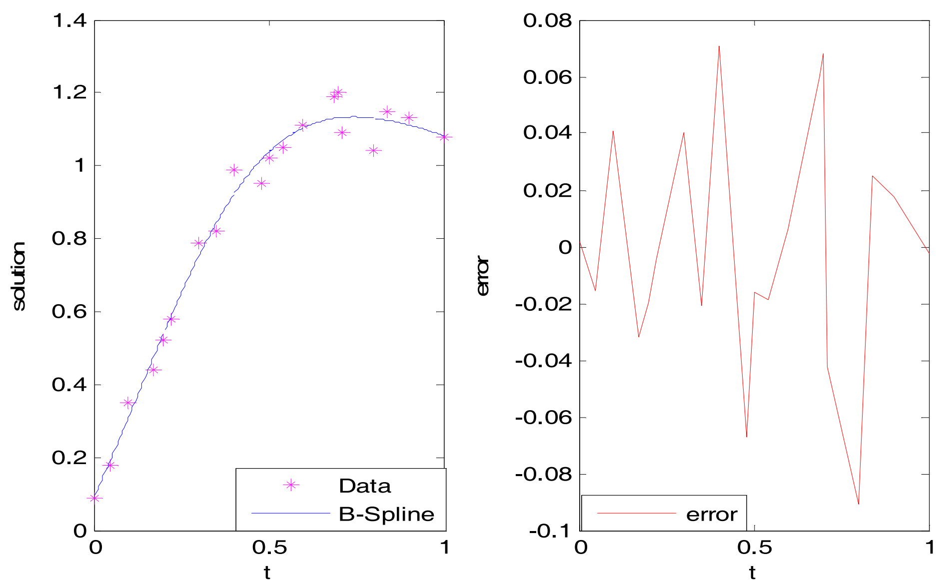

Fig.2. rigth, the graph of F (error), left, data points and B-spline function graph.

3.3. Example

In this example we choose $m = 20, n = 8$, and

$$x = (0,0.05,0.1,0.17,0.2,0.22,0.3,0.35,0.4,0.48,\\ 0.5,0.54,0.6,0.69,0.7,0.71,0.8,0.84,0.9,1),$$

$$y = (0,0.055,0.089,0.135,0.185,0.225,0.27,0.3,0.33,0.35,\\ 0.362,0.355,0.365,0.36,0.34,0.321,0.292,0.228,0.185,0.12).$$

Exact solution is a least squares polynomial function, (with $n = 5$) as follows:

$$p(t) = a_1 + a_2 t + a_3 t^2 + a_4 t^3 + a_5 t^4$$

and B- spline least squares solution is given by

$$\tilde{p}(t) = \sum_{i=1}^{n} a_i Q_i(t)$$

The used partition for B-spline is as follows

$t = (-0.6\ ,-0.4\ ,-0.2\ ,0\ ,0.2\ ,0.4\ ,0.6\ ,0.8\ ,1\ ,1.2\ ,1.4\ ,1.6)$.

The numerical results of 5 Nonlinear least squares methods are compared with B-spline least squares method in table.1. The results show efficiency and precision of B-spline method.

Table.1. comparison of nonlinear least squares method with B-spline.

Method	least squares method	Iteration number	CPU-Time	$\lVert\nabla F\rVert$	$\lVert f\rVert$
GN	Polynomial	1	0.386037	2.4326×10^{-18}	0.0518
LM	Polynomial	509	90.226827	9.9511×10^{-7}	0.0645
DL	Polynomial	320	78.187504	7.3228×10^{-29}	0.0518
LMF	Polynomial	11	2.75137	7.0919×10^{-7}	0.06163
HM	Polynomial	508	141.271116	9.9511×10^{-7}	0.0645
BS	B-spline	1	2.148313	0	0.0452

3.4. Example

In this example we choose $m = 20, n = 8$, and

$x = (0, 0.05\ ,0.1\ ,0.17, 0.2\ ,0.22\ ,0.3, 0.35\ ,0.4\ ,0.48\ ,$
$0.5\ ,0.54\ ,0.6, 0.69\ ,0.7, 0.71\ ,0.8\ ,0.84\ ,0.9, 1),$

$y = (0.09\ ,0.18\ ,0.35\ ,0.44\ ,0.52\ ,0.58\ ,0.79\ ,0.82\ ,0.99\ ,0.95\ ,$
$1.02\ ,1.05\ ,1.11\ ,1.19\ ,1.2\ ,1.09\ ,1.04\ ,1.15\ ,1.13\ ,1.08)$

The expression of nonlinear solution is as follows $n = 4$:

$$p(t) = a_1 \sin a_2 t + e^{a_3} a_4 t$$

and B- spline least squares solution is given by

$$\tilde{p}(t) = \sum_{i=1}^{n} a_i Q_i(t) \cdot$$

The partition is as follows

$t = (-0.6\ ,-0.4\ ,-0.2\ ,0\ ,0.2\ ,0.4\ ,0.6\ ,0.8\ ,1\ ,1.2\ ,1.4\ ,1.6)$.

The numerical results of 5 Nonlinear least squares methods are compared with B-spline least squares method in table.2. The results show efficiency and precision of B-spline method

Table.2. comparison of nonlinear least squares method with B-spline.

Method	least squares method	Iteration number	CPU-Time	$\lVert \nabla F \rVert$	$\lVert f \rVert$
GN	nonlinear	12	5.426	3.8504×10^{-9}	0.2111
LM	nonlinear	280	161.6549	9.9515×10^{-7}	0.214
DL	nonlinear	34	20.095	1.6593×10^{-11}	0.2105
LMF	nonlinear	24	14.4966	4.76403×10^{-7}	0.2125
HM	nonlinear	279	283.535466	9.91509×10^{-7}	0.2140
BS	B-spline	1	1.846861	0	0.1863

4. Conclusion

In this paper a linear least squares method, using B-spline functions is presented. The results of numerical experiments are compared with the results of well-known nonlinear methods. In a great number of examples, the obtained residues for B-spline are less than the non-linear examples. The great advantage of B-spline least squares method is the independence of solution to initial guess point, and CPU Time is very low in B-spline least squares method relative to nonlinear least squares methods.

References

[1] Balda,. M., *"Levenberg-Marquardt-Fletcher Algorithm for Nonlinear Least Squares problems", Math Works, File Exchange, ID 16063. (2007).*

[2] Burden, R.L., and Faires, J. D. *"Numerical Analysis", Sixth Edition. Brooks/Cole Publishing Company (1997).*

[3] Dag I., Irk D., and Sahin A., *" B-Spline Collocation Methods for Numerical Solutions of The Burgers' Equation", copyright Hindawi Publishing Corporation, Mathematical Problems in Engineering, (2005).*

[4] Fletcher, R., *"A Modified Marquart Subroutine for Nonlinear Least Squares". Rpt. AERE -R 6799, Harwell. (1971).*

[5] Frandsen, *P.E.,Jonasson,K.Nielsen,H.B. and Tingleff,O.,"Unconstrained Optimization",* 3rd *Edition, IMM, DTU. (2004).*

[6] Levenberg, K., *"A Method for the Solution of Certain Problems in Least Squares". Quart. Appl. Math.2, pp 164-168. (1944).*

[7] Madsen, K., "*A Combined Gauss-Newton and Quasi-Newton Method for Nonlinear Least Squares". Institute for Numerical Analysis (new part of IMM), DTU. Report NI -88-10. (1988).*

[8] Marquart.,*"An Algorithm for Least Squares Estimation on Nonlinear Parameters". SIAM J.Appl. Math.11, pp 431-441, (1963).*

[9] Powell, M. J. D., *"A Hybrid Method for Nonlinear Equations."In P.Rabinowitz (ed): Numerical Methods for Nonlinear Algebraic Equations, Gordom and Breach.pp 87 ff, (1970).*

[10] Stoer, J., and Bulirsch, R., *"Introduction to Numerical Analysis" translated by R, Bartels. W .Gautschi and C. Witzgall, new York, Heidelbery Berlin (1980).*

Received: October, 2011