Asian J. Exp. Sci., Vol. 23, No. 1, 2009; 329-340

Design and Analysis of Quarter Sweep ADEI Algorithm for Linear and Nonlinear Two Point Boundary Value Problems Containing Singularity: Application to Burger's Equation

Navnit Jha[1]*, Bhupendra K. Sharma[1] and R.C. Chaudhary[2]

[1] Mathematics Group, Birla Institute of Technology & Science, Pilani-333031 (Raj.); India.

[2] Department of Mathematics, University of Rajasthan, Jaipur-302004 (Raj.); India.

Abstract : The formulation of quarter sweep alternating decomposition explicit iterative (QSADEI) method for the numerical solution of the non linear singular equations is discussed in this paper. The concept of QSADEI method was inspired via combination between the quarter sweep iterative and alternating decomposition explicit iterative (ADEI) methods. The proposed QSADEI method shows the superiority over the corresponding Gauss seidal iterative method. The method is applicable to problems both in Cartesian and polar coordinates. The convergence analysis is briefly discussed. Numerical results are provided to illustrate the proposed method.

Key words : Full-, Half-, Quarter-sweep methods, Alternating Decomposition explicit method, Fourth order method, Singular equations, Burger's equations, RMS errors.

Introduction

Consider the non linear two point boundary value problem

$$-u'' + g(t,u,u') = 0, \quad a < t < b \qquad \text{....... (1)}$$

subject to the natural boundary conditions $u(a) = A$, $u(b) = B$

The two point boundary value problem mentioned above has a unique solution (Keller 1968) and is of common occurrence in many fields of science and engineering, e.g., geophysics, quantum mechanics, fluid dynamics, aerodynamics, etc. There are many methods available for the above problems. Quarter sweep alternating decomposition explicit iterative (QSADEI) method is one of them. The quarter sweep iterative method was initiated by Othman and Abdullah, 2000. The concept of this method is extension of half sweep iterative method through explicit decoupled group iterative method (Yousif and Evans, 1995). The QSADEI method explained in this paper is combination of quarter sweep iterative method and alternating decomposition explicit iterative (ADEI) method, earlier discussed by Sahimi *et. al.,* 1993; Sulaiman *et. al.,* 2004 has considered QSADEI method for solving one dimensional diffusion equations, where the theoretical convergence of the method is not discussed and their methods are only applicable to non singular problems. Difficulties were experienced in the past for the numerical solution of singular equations in polar coordinate. The solution usually deteriorates in the vicinity of singularity. In this paper, the method is refined in such a manner that solutions retain their order and accuracy everywhere in the solution region even in the vicinity of the singularity. We also discuss QSADEI and Newton-

* **Corresponding author :** Navnit Jha, Mathematics Group, Birla Institute of Technology & Science, Pilani-333031 (Raj.); India; E-mail : *navnitjha@hotmail.com*

QSADEI method for linear and non linear two point boundary value problem (1) with fourth order accuracy. The theoretical convergence of the method is discussed.

The finite difference approximation to equation (1) in case of the full-, half- and quarter sweep iterative assume that solution domain can be uniformly partitioned into $N+s$ subintervals with interval of $h=(b-a)/(N+s)$, where s = 1, 2, 4 respectively denotes full-, half- and quarter sweep. Let us define $\bar{u}'_k=(u_{k+s}-u_{k-s})/(2sh)$, $\bar{u}'_{k\pm s}=(\pm 3u_{k\pm s}\mp 4u_k \pm u_{k\mp s})/(2sh)$, $\bar{G}_{k\pm s}=g\left(t_{k\pm s},u_{k\pm s},\bar{u}'_{k\pm s}\right)$, $\bar{\bar{u}}'_k=\bar{u}'_k-\frac{sh}{20}\left(\bar{G}_{k+s}-\bar{G}_{k-s}\right)$, $\bar{\bar{G}}_k=g\left(t_k,u_k,\bar{\bar{u}}'_k\right)$

Then the fourth order finite difference replacement of (1) is

$$-\left(u_{k-s}-2u_k+u_{k+s}\right)+\frac{(sh)^2}{12}\left[\bar{G}_{k+s}+\bar{G}_{k-s}+10\bar{\bar{G}}_k\right]+O\left(h^6\right)=0,\quad k = s\ (s)\ N \qquad ....(2)$$

where u_k denote the function satisfying the difference equation at the grids $t_k=a+kh,\ k=0(1)N+s,\ s=1,2,4$ of the discrete problem and $u_0=A$ and $u_{N+s}=B$.

Mohanty *et al.*, 2004a has discussed the fourth order method (2) for s = 1, where whole inner grids are involved but implementation of quarter sweep (s = 4) involves nearly one quarter of whole inner grids, and hence superior performance. To discuss the applications of finite difference formula (2), let us consider the linear singular equation.

$$u''=d(t)u'+e(t)u+g(t),\ 0 < t< 1 \qquad ....(3)$$

where $d(t)=\frac{-\alpha}{t}$ and $e(t)=\frac{\alpha}{t^2}$.

For α = 0, 1 and 2, the above equation represents cartesian, cylindrical and spherical problem, respectively. Applying the method (2) to the equation (3), we obtain

$$u_{k-s}-2u_k+u_{k+s}=\frac{(sh)^2}{12}\left[10\,d_k\bar{u}'_k+\left(1-\frac{shd_k}{2}\right)d_{k+s}\bar{u}'_{k+s}+\left(1+\frac{shd_k}{2}\right)d_{k-s}\bar{u}'_{k-s}\right]$$

$$+\frac{(sh)^2}{12}\left[10\,e_ku_k+\left(1-\frac{shd_k}{2}\right)e_{k+s}u_{k+s}+\left(1+\frac{shd_k}{2}\right)e_{k-s}u_{k-s}\right]$$

$$+\frac{(sh)^2}{12}\left[10\,g_k+\left(1-\frac{shd_k}{2}\right)g_{k+s}+\left(1+\frac{shd_k}{2}\right)g_{k-s}\right]=0,\quad k=s(s)N \qquad ....(4)$$

where $d_k=d(t_k)$, $e_k=e(t_k)$ etc.

The difference scheme (4) is of fourth order accurate for the numerical solution of (1). However, the scheme (4) fails when the solution is to be determined at k = s, the vicinity of the singularity t = 0. We overcome this difficulty by modifying the method (4) in such a manner that the solution retains its order and accuracy everywhere including the region of vicinity of the singularity t = 0.

We need the following approximations:

$$d_{k\pm s} = d_k \pm (sh)\, d_k^{'} + \frac{(sh)^2}{2} d_k^{''} \pm O\left(h^3\right) \qquad \text{........(5a)}$$

$$e_{k\pm s} = e_k \pm (sh) e_k^{'} + \frac{(sh)^2}{2} e_k^{''} \pm O\left(h^3\right) \qquad \text{........(5b)}$$

$$g_{k\pm s} = g_k \pm (sh) g_k^{'} + \frac{(sh)^2}{2} g_k^{''} \pm O\left(h^3\right) \qquad \text{........(5c)}$$

where $d_k = \frac{-\alpha}{t_k},\ d_k^{'} = \frac{\alpha}{t_k^2},\ d_k^{''} = \frac{-2\alpha}{t_k^3}$ etc.

Substituting the approximations (5) into (4) and neglecting $O\left(h^6\right)$ terms, we get a linear difference equation of the form

$$a_k u_{k-s} + b_k u_k + c_k u_{k+s} = RH_k\,,\ \ k = s(s)N \qquad \text{.......(6)}$$

where

$$a_k = -1 - \frac{sh}{24}\left[12 d_k - 2sh\left(2d_k^{'} - d_k^2\right) + (sh)^2\left(d_k^{''} - d_k\, d_k^{'}\right)\right] + \frac{(sh)^2}{12}\left[e_k - \frac{sh}{2}\left(2e_k^{'} - d_k e_k\right)\right]$$

$$b_k = 2 - \frac{(sh)^2}{6}\left(2d_k^{'} - d_k^2 - 5e_k\right) + \frac{(sh)^4}{12}\left(e_k^{''} - d_k e_k^{'}\right)$$

$$c_k = -1 + \frac{sh}{24}\left[12 d_k + 2sh\,(2d_k^{'} - d_k^2) + (sh)^2 (d_k^{''} - d_k d_k^{'})\right], + \frac{(sh)^2}{12}\left[e_k + \frac{sh}{2}(2e_k^{'} - d_k e_k)\right]$$

$$RH_k = \frac{-(sh)^2}{12}\left[12 g_k + (sh)^2\left(g_k^{''} - d_k g_k^{'}\right)\right]$$

Now we consider the non-linear singular equation of the form

$$\eta u^{''} = b(t) u^{'} + uu^{'} + c(t) u + g(t), \quad 0 < t < 1 \qquad \text{...... (7)}$$

where $b(t) = -\frac{\alpha\eta}{t}$, $c(t) = \frac{\alpha\eta}{t^2}$ and $\frac{1}{\eta}$ represents a Reynolds number.

For $\alpha = 1$ and 2, the equation above represents the steady-state Burgers' equation in cylindrical and spherical coordinates respectively. Now applying the formula (2) to the non linear equation (7) and using the same technique discussed for linear singular equation, a difference scheme of $O(h^4)$ for the equation (7) may be expressed as

$$\begin{aligned}
\phi_k(u_{k-s}, u_k, u_{k+s}) \equiv & -\eta\left[u_{k+s} - 2u_k + u_{k-s}\right] \\
& + \frac{(sh)^2}{12}\left[\begin{array}{l}(12c_k + (sh)^2 c_k'' - 2shHb_k c_k')u_k + c_k(u_{k+s} - 2u_k + u_{k-s}) \\ + \left(shc_k' - Hc_k(b_k + u_k)\right)(u_{k+s} - u_{k-s}) - 2shHc_k' u_k^2\end{array}\right] \\
& + \frac{sh}{24}\left[\begin{array}{l}\left(12b_k + (sh)^2 b_k'' - 2hHb_k'(b_k + u_k)\right)(u_{k+s} - u_{k-s}) \\ + \left(4shb_k' - 4Hb_k(b_k + u_k)\right)(u_{k+s} - 2u_k + u_{k-s})\end{array}\right] \\
& + \frac{sh}{24}\left[\begin{array}{l}3(u_{k+s} + 2u_k + u_{k-s})(u_{k+s} - u_{k-s}) - H(b_k + u_k)(u_{k+s} - u_{k-s})^2 \\ -4H(b_k + u_k)u_k(u_{k+s} - 2u_k + u_{k-s})\end{array}\right] \\
& + \frac{(sh)^2}{12}\left[12g_k + (sh)^2 g_k'' - 2shHg_k'(b_k + u_k)\right] = 0, \quad k = s(s)N
\end{aligned} \tag{9}$$

where $H = \frac{sh}{2\eta}$.

Note that, schemes (6) and (9) have local truncation errors of $O(h^4)$ and free from the terms $1/(k \pm s)$, thus easily solved for k = s (s) N in the region 0 < t < 1.

3. The Quarter-sweep Adei Methods

The linear difference equations (6) may be written in tri-diagonal linear system

$\boldsymbol{Au} = \boldsymbol{R}$ .... (10)

where

$$\boldsymbol{A} = \begin{bmatrix} b_s & c_s & & & \\ a_{2s} & b_{2s} & c_{2s} & & \\ & \ddots & \ddots & \ddots & \\ & & a_{N-s} & b_{N-s} & c_{N-s} \\ & & & a_N & b_N \end{bmatrix}_{(N-s)/s}, \quad \boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-s} \\ u_N \end{bmatrix}, \quad \boldsymbol{R} = \begin{bmatrix} RH_s - A \\ RH_{2s} \\ \vdots \\ RH_{N-s} \\ RH_N - \end{bmatrix}$$

In case of $s = 1, 2, 4$ for the respective formulation of full-, half- and quarter sweep ADEI method (Sulaiman et. al. 1993), we split the matrix A as follows

$$\boldsymbol{A} = \boldsymbol{G}_1 + \boldsymbol{G}_2 - \frac{1}{6}\boldsymbol{G}_1\boldsymbol{G}_2$$

where

$$\boldsymbol{G}_1 = \begin{bmatrix} 1 & 0 & 0 & & \\ p_s & 1 & 0 & & \\ 0 & p_{2s} & 1 & & \\ & \ddots & \ddots & & \\ & & p_{N-s} & & 1 \end{bmatrix}, \quad \boldsymbol{G}_2 = \begin{bmatrix} q_s & r_s & 0 & & \\ 0 & q_{2s} & r_{2s} & & \\ & \ddots & \ddots & & \\ & & q_{N-s} & r_{N-s} \\ & & 0 & q_N \end{bmatrix},$$

and

$$q_s = \frac{6}{5}(b_s - 1), \; r_i = \frac{6}{5}c_i, \; p_i = \frac{6a_{i+s}}{6 - q_i}, \; q_i \neq 6, \; q_{i+s} = \frac{6}{5}\left(b_{i+s} - \frac{1}{6}p_i r_i - 1\right), \; i = s(s)N - s$$

Then quarter sweep alternating direction explicit iterative (QSADEI) method is given by

$$(\omega\boldsymbol{I} + \boldsymbol{G}_1)\boldsymbol{u}^{(j+1/2)} = (\omega\boldsymbol{I} - g\boldsymbol{G}_2)\boldsymbol{u}^{(j)} + \boldsymbol{R} \qquad \text{...... (11a)}$$

$$(\omega\boldsymbol{I} + \boldsymbol{G}_2)\boldsymbol{u}^{(j+1)} = (\omega\boldsymbol{I} - g\boldsymbol{G}_1)\boldsymbol{u}^{(j+1/2)} + g\boldsymbol{R} \qquad \text{...... (11b)}$$

where $\omega > 0$ is the acceleration parameter, $\boldsymbol{u}^{(j+1/2)}$ is an intermediate vector and $g = 1 + \omega/6$.

To discuss the algorithms, the QSADEI method (11) takes the matrix form

$$\begin{bmatrix} \omega+1 & 0 & 0 & & \\ p_s & \omega+1 & 0 & & \\ 0 & p_{2s} & \omega+1 & & \\ & \ddots & \ddots & & \\ & & p_{N-s} & & \omega+1 \end{bmatrix} \begin{bmatrix} u_s \\ u_{2s} \\ u_{3s} \\ \vdots \\ u_N \end{bmatrix}^{(j+1/2)} = \begin{bmatrix} \omega - gq_s & -gr_s & 0 & & \\ 0 & \omega - gq_{2s} & -gr_{2s} & & \\ & \ddots & \ddots & & \\ & & \omega - gq_{N-s} & -g\,r_{N-s} \\ & & 0 & \omega - gq_N \end{bmatrix} \begin{bmatrix} u_s \\ u_{2s} \\ u_{3s} \\ \vdots \\ u_N \end{bmatrix}^{(j)} + \begin{bmatrix} R_s \\ R_{2s} \\ R_{3s} \\ \vdots \\ R_N \end{bmatrix}$$

...... (12a)

$$\begin{bmatrix} \omega+q_s & r_s & 0 & & \\ 0 & \omega+q_{2s} & r_{2s} & & \\ & \ddots & \ddots & & \\ & & \omega+q_{N-s} & r_{N-s} & \\ & & & \omega+q_N & \end{bmatrix}\begin{bmatrix} u_s \\ u_{2s} \\ u_{3s} \\ \vdots \\ u_N \end{bmatrix}^{(j+1)} = \begin{bmatrix} \omega-g & 0 & & & \\ -gp_s & \omega-g & & & \\ & \ddots & \ddots & & \\ & & -gp_{N-2s} & \omega-g & 0 \\ & & & -gp_{N-s} & \omega-g \end{bmatrix}\begin{bmatrix} u_s \\ u_{2s} \\ u_{3s} \\ \vdots \\ u_N \end{bmatrix}^{(j+1/2)} + \begin{bmatrix} gR_s \\ gR_{2s} \\ gR_{3s} \\ \vdots \\ gR_N \end{bmatrix}$$

...... (12b)

Simplifying equations (12), we obtain following algorithms

At level (j + 1/2)

$$u_s^{(j+1/2)} = \frac{1}{1+\omega}\left((\omega-gq_s)u_s^{(j)} - gr_s u_{2s}^{(j)} + R_s\right)$$

$$u_i^{(j+1/2)} = \frac{1}{1+\omega}\left((\omega-gq_i)u_i^{(j)} - gr_i u_{i+s}^{(j)} - p_{i-s}u_{i-s}^{(j+1/2)} + R_i\right),\ i = 2s(s)N-s$$

$$u_N^{(j+1/2)} = \frac{1}{1+\omega}\left((\omega-gq_N)u_N^{(j)} - p_{N-s}u_{N-s}^{(j+1/2)} + R_N\right)$$

At level (j + 1)

$$u_N^{(j+1)} = \frac{1}{q_N+\omega}\left(-gp_{N-s}u_{N-s}^{(j+1/2)} + (\omega-g)u_N^{(j+1/2)} + gR_N\right)$$

$$u_i^{(j+1)} = \frac{1}{q_i+\omega}\left(-gp_{i-s}u_{i-s}^{(j+1/2)} + (\omega-g)u_i^{(j+1/2)} - r_i u_{i+s}^{(j+1)} + gR_i\right),\ i = N-s(-s)2s$$

$$u_s^{(j+1)} = \frac{1}{q_s+\omega}\left((\omega-g)u_s^{(j+1/2)} - r_s u_{2s}^{(j+1)} + gR_s\right)$$

4. The Newton-quarter-sweep Adei Methods

For non linear difference scheme (9), we define

$$\boldsymbol{u} = \begin{bmatrix} u_s \\ u_{2s} \\ \vdots \\ u_N \end{bmatrix},\quad \phi(\boldsymbol{u}) = \begin{bmatrix} \phi_s(u) \\ \phi_{2s}(u) \\ \vdots \\ \phi_N(u) \end{bmatrix}$$

and

$$a_k(u) = \frac{\partial \phi_k}{\partial u_{k-s}}, \quad k = 2s(s)N \; ; \qquad ;$$

Then the Jacobian of may be written as

$$\boldsymbol{T} = \begin{bmatrix} b_s(u) & c_s(u) & & & \\ a_{2s}(u) & b_{2s}(u) & c_{2s}(u) & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & a_N(u) & b_N(u) \end{bmatrix}$$

with any initial guess $\boldsymbol{u}^{(0)}$, we define

$$\boldsymbol{u}^{(j+1)} = \boldsymbol{u}^{(j)} + \Delta \boldsymbol{u}^{(j)}, \quad j = 0, 1, 2, \cdots \cdots \quad \ldots\ldots \; (13)$$

where $\Delta \boldsymbol{u}^{(j)}$ satisfies the system

$$\boldsymbol{T} \Delta \boldsymbol{u}^{(j)} = -\phi\left(u^{(j)}\right), \quad j = 0, 1, 2, \cdots \cdots \quad \ldots\ldots \; (14)$$

For Newton-QSADE method, we split the matrix T as

$$\boldsymbol{T} = \boldsymbol{T}_1 + \boldsymbol{T}_2 + \frac{1}{6} \boldsymbol{T}_1 \, \boldsymbol{T}_2,$$

$$\phi(\boldsymbol{u}) \qquad b_k(u) = \frac{\partial \phi_k}{\partial u_k}, \quad k = s(s)N - s$$

where

$$\boldsymbol{T}_1 = \begin{bmatrix} 1 & 0 & 0 & & \\ p_s & 1 & 0 & & \\ 0 & p_{2s} & 1 & & \\ & \ddots & \ddots & & \\ & & p_{N-s} & & 1 \end{bmatrix}, \quad \boldsymbol{T}_2 = \begin{bmatrix} q_s & r_s & 0 & & \\ 0 & q_{2s} & r_{2s} & & \\ & \ddots & \ddots & & \\ & & q_{N-s} & r_{N-s} & \\ & & 0 & q_N & \end{bmatrix}$$

Then Newton-QSADEI method may be written as (see Mohanty et. al. 2004)

$$(\omega \boldsymbol{I} + \boldsymbol{T}_1) \Delta \boldsymbol{u}^{(j+1/2)} = -\phi\left(\boldsymbol{u}^{(j)}\right) + (\omega \boldsymbol{I} + g \boldsymbol{T}_2) \Delta \boldsymbol{u}^{(j)} \qquad \ldots\ldots \; (15a)$$

$$(\omega \boldsymbol{I} + \boldsymbol{T}_2) \Delta \boldsymbol{u}^{(j+1)} = -\boldsymbol{\varphi}\left(\boldsymbol{u}^{(j)}\right) + (\omega \boldsymbol{I} - g \boldsymbol{T}_1) \Delta \boldsymbol{u}^{(j+1/2)}, \; j = 0, 1, 2, \ldots. \qquad \ldots\ldots \; (15b)$$

5. Convergence Analysis

Equations (11a) and (11b) can be written as

$\boldsymbol{u}^{(j+1)} = \boldsymbol{G}\boldsymbol{u}^{(j)} + \boldsymbol{C}$ ...... (16)

where

$$\boldsymbol{G} = (\omega\boldsymbol{I} + \boldsymbol{G}_2)^{-1}(\omega\boldsymbol{I} - g\boldsymbol{G}_1)(\omega\boldsymbol{I} + \boldsymbol{G}_1)^{-1}(\omega\boldsymbol{I} - g\boldsymbol{G}_2)$$

$$\boldsymbol{C} = (\omega\boldsymbol{I} + \boldsymbol{G}_2)^{-1}\left((\omega\boldsymbol{I} - g\boldsymbol{G}_1)(\omega\boldsymbol{I} + \boldsymbol{G}_1)^{-1} + g\right)\boldsymbol{R}$$

Assume that u is the exact solution of (10), then $(\boldsymbol{G}_1 + \boldsymbol{G}_2)\boldsymbol{u} = \boldsymbol{R}$

and

$$(\omega\boldsymbol{I} + \boldsymbol{G}_1)\boldsymbol{u} = (\omega\boldsymbol{I} - g\boldsymbol{G}_2)\boldsymbol{u} + \boldsymbol{R} \qquad \text{...... (17a)}$$

$$(\omega\boldsymbol{I} + \boldsymbol{G}_2)\boldsymbol{u} = (\omega\boldsymbol{I} - g\boldsymbol{G}_1)\boldsymbol{u} + g\boldsymbol{R} \qquad \text{...... (17b)}$$

Let $\boldsymbol{\varepsilon}^{(j)} = \boldsymbol{u}^{(j)} - \boldsymbol{u}$ is the error vector, therefore from equations (11) and (17), we get

$$(\omega\boldsymbol{I} + \boldsymbol{G}_1)\boldsymbol{\varepsilon}^{(j+1/2)} = (\omega\boldsymbol{I} - g\boldsymbol{G}_2)\boldsymbol{\varepsilon}^{(j)} \qquad \text{...... (18a)}$$

$$(\omega\boldsymbol{I} + \boldsymbol{G}_2)\boldsymbol{\varepsilon}^{(j+1)} = (\omega\boldsymbol{I} - g\boldsymbol{G}_1)\boldsymbol{\varepsilon}^{(j+1/2)} \qquad \text{...... (18b)}$$

and hence $\boldsymbol{\varepsilon}^{(j+1)} = \boldsymbol{G}\boldsymbol{\varepsilon}^{(j)}$

For the convergence, it is required to prove $\rho(\boldsymbol{G}) < 1$ for any $\omega > 0$.

Define $\boldsymbol{G}^* = (\omega\boldsymbol{I} + \boldsymbol{G}_2)\boldsymbol{G}(\omega\boldsymbol{I} + \boldsymbol{G}_2)^{-1}$

$$= (\omega\boldsymbol{I} - g\boldsymbol{G}_1)(\omega\boldsymbol{I} + \boldsymbol{G}_1)^{-1}(\omega\boldsymbol{I} - g\boldsymbol{G}_2)(\omega\boldsymbol{I} + \boldsymbol{G}_2)^{-1}$$

then $\boldsymbol{G}^*$ is similar to $\boldsymbol{G}$, and hence $\rho(\boldsymbol{G}) = \rho(\boldsymbol{G}^*)$

Let λ_i and μ_i be the eigen values of $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ respectively, then

$$\lambda_i = 1, \quad \mu_i = \frac{6}{5}(2b_s - 1), \quad i = s(s)N$$

Since $g = 1 + \frac{\omega}{6} > 0$, we have

$$\left\|(\omega\boldsymbol{I} - g\boldsymbol{G}_1)(\omega\boldsymbol{I} + \boldsymbol{G}_1)^{-1}\right\| \le \left\|(\omega\boldsymbol{I} - g\boldsymbol{G}_1)\right\| . \left\|(\omega\boldsymbol{I} + \boldsymbol{G}_1)\right\|^{-1} \le \underset{i}{Max}\left|\frac{\omega - g\lambda_i}{\omega + \lambda_i}\right| < 1$$

$$\left\|(\omega\boldsymbol{I} - g\boldsymbol{G}_2)(\omega\boldsymbol{I} + \boldsymbol{G}_2)^{-1}\right\| \le \left\|(\omega\boldsymbol{I} - g\boldsymbol{G}_2)\right\| . \left\|(\omega\boldsymbol{I} + \boldsymbol{G}_2)\right\|^{-1} \le \underset{i}{Max}\left|\frac{\omega - g\mu_i}{\omega + \mu_i}\right| < 1$$

Thus we obtain

$$\rho(\boldsymbol{G})=\rho(\boldsymbol{G}^{*})\leq\left\|\boldsymbol{G}^{*}\right\|\leq\left\|(\omega\boldsymbol{I}-g\boldsymbol{G}_1)(\omega\boldsymbol{I}+\boldsymbol{G}_1)^{-1}\right\|\left\|(\omega\boldsymbol{I}-g\boldsymbol{G}_2)(\omega\boldsymbol{I}+\boldsymbol{G}_2)^{-1}\right\|<1$$

Hence the convergence follows.

6. Numerical Experiments and Comparative Results

To prove the efficiency of the implementation of the QSADEI method, two singular problems are considered for computation whose exact solutions are known. The right hand side function and boundary conditions may by obtained using the exact solution as a test procedure. All results of numerical experiments, which were gained from implementation of the Gauss seidal, full (s = 1), half (s = 2) and quarter (s = 4) sweep ADEI algorithms has been recorder in the tables. While solving non-linear equation, 5 inner iterations were considered. In all cases, we have taken $\boldsymbol{u}^{(0)}=\boldsymbol{0}$ and the iterations were stopped when $\left|\boldsymbol{u}^{(j+1)}-\boldsymbol{u}^{(j)}\right|\leq 10^{-14}$ was achieved.

Linear Singular Problem

$$u''+\frac{\alpha}{t}u'-\frac{\alpha}{t^2}u=g(t),\quad 0<t<1$$

The exact solution is . The root mean square (RMS) errors and number of iterations for GS, full-, half- and quarter sweep ADEI methods are tabulated in Table-1.

Non-linear Singular Problem (Burger's Equation)

$u(t)=e^{t^4}$

$$\eta\left(u''+\frac{\alpha}{t}u'-\frac{\alpha}{t^2}u\right)=uu'+g(t),\quad 0<t<1$$

The exact solution is given by $u(t)=t^2\cosh(t)$. The RMS errors and number of iterations both for Newton-ADEI ($\omega=1.2$) and Newton-GS are tabulated in Table-2.

Table 1 : Comparison of the number of iterations and root mean square errors

	Full-sweeps s =1				*Half-sweeps s*		
N	GS	ADEI	ω_{opt}	RMS	GS	ADEI	ω_{opt}
							α=1
16	588	102	0.590	0.113(-03)	172	57	0.790
32	2112	192	0.477	0.805(-05)	588	102	0.590
64	7793	366	0.414	0.534(-06)	2112	192	0.477
128	29124	726	0.378	0.344(-07)	7793	366	0.414
256	109287	1406	0.359	0.215(-08)	29124	726	0.378
512	410647	2882	0.349	0.577(-10)	109287	1406	0.359
							α=2
16	436	88	0.644	0.128(-03)	128	48	0.910
32	1564	164	0.503	0.912(-05)	436	88	0.644
64	5768	323	0.426	0.605(-06)	1564	164	0.503
128	21558	608	0.384	0.389(-07)	5768	323	0.426
256	80986	1263	0.362	0.246(-08)	21558	608	0.384
512	304474	3094	0.352	0.852(-10)	80986	1263	0.362

Table 2 : Comparison of the number of iterations and root mean square errors

N	*Full-sweeps* $s = 1$			*Half-sweeps* $s = 2$			*Quarter-sweeps* $s = 4$		
	GS	ADEI	RMS	GS	ADEI	RMS	GS	ADEI	RMS
				$\alpha = 1, \eta = 0.10$					
64	101	1349	0.184(-07)	275	89	0.274(-06)	77	34	0.384(-05)
128	3776	1326	0.120(-08)	1011	349	0.184(-07)	275	89	0.274(-06)
256	14191	5020	0.747(-10)	3776	326	0.120(-08)	1011	349	0.184(-07)
512	53404	18965	0.153(-10)	4191	20	0.747(-10)	3776	1326	0.120(-08)
				$\alpha = 1, \eta = 0.02$					
64	401	133	0.257(-06)	110	40	0.391(-05)	32	32	0.553(-04)
128	1497	514	0.164(-07)	401	133	0.257(-06)	110	40	0.391(-05)
256	5623	1955	0.104(-08)	1497	514	0.164(-07)	401	133	0.257(-06)
512	21162	7396	0.629(-10)	5623	1955	0.104(-08)	497	514	0.164(-07)
				$\alpha = 2, \eta = 0.10$					
64	873	300	0.137(-07)	237	76	0.207(-06)	67	37	0.296(-05)
128	3261	1144	0.885(-09)	873	300	0.137(-07)	237	76	0.207(-06)
256	12264	4333	0.563(-10)	3261	1144	0.885(-09)	873	300	0.137(-07)
512	46184	16389	0.149(-10)	2264	4333	0.563(-10)	3261	1144	0.885(-09)
				$\alpha = 2, \eta = 0.02$					
128	1341	460	0.161(-07)	360	118	0.252(-06)	99	39	0.384(-05)
256	5038	1749	0.102(-08)	1341	460	0.161(-07)	360	118	0.252(-06)
512	18972	6625	0.619(-10)	5038	1749	0.102(0-8)	1341	460	0.161(-07)

References

Mohanty R.K., Sachdev P.L. and Navnit J. (2004): An $O(h^4)$ accurate cubic spline TAGE method for nonlinear singular two point boundary value problems. *Applied Mathematics and Computations*, **158**, 853-868.

Mohanty R.K., Sachdev P.L. and Navnit J. (2004): TAGE method for nonlinear singular two point boundary value problem using a fourth order difference scheme. *Neural, parallel and & Scientific Computations*, **11**, 281-296.

Othman M. and Abdullah A.R. (2000): An efficient four point modified explicit group poisson solver. *Int. J. Comput. Math*., **76**, 203-217.

Abdullah A.R. (1991): The four point explicit decoupled group (EDG) method: a fast poisson solver. *Int. J. Comput. Math*., **38**, 61-70.

Sahimi M.S., Ahmad A. and Bakar A.A. (1993): The iterative alternating decomposition explicit (IADE) method to solve the heat conduction equation. *Int. J. Comput. Math*., **47**, 219-229.

Sulaiman J., Othman M. and Hasan M.K. (2004): Quarter-sweep iterative alternating decomposition explicit algorithm applied to diffusion equations. *Int. J. Comput. Math*., **81(12)**, 1559-1565.

Yousif W.S. and Evans D.J. (1995): Explicit de-coupled group iterative methods and their implementations. *Parallel Algorithms and Applications*, **7**, 53-71.

Keller H.B. (1968): Numerical methods for two point boundary value problem, Blaisdell Pub. Co.