High accuracy extrapolation cascadic Newton multigrid computation for two-dimensional nonlinear Poisson equations

Ming Li *, Chenglin Li

Department of Mathematics and Statistics, Honghe University, Mengzi, 661199, Yunnan, China

Email: mathlm@126.com (M. Li)

ABSTRACT: An extrapolation cascadic Newton multigrid (ECNMG) method is proposed for high accuracy numerical solutions of two-dimensional nonlinear Poisson equations, by incorporating the fourth-order compact difference schemes, the extrapolation techniques and the existing Newton multigrid method. A series of grid level dependent computational tolerances are discussed to distribute computational cost on different grids, and an extrapolation interpolation strategy and a bi-quartic polynomial interpolation are used for two fourth-order approximations from current and previous grids to provide an extremely accurate initial guess on the next finer grid, which can greatly reduce the iterations of the Newton multigrid computation for computing an approximation with discretization-level accuracy. Additionally, a completed Richardson extrapolation technique is adopted for the fourth-order computed solution to generate a sixth-order extrapolated solution cheaply. Numerical results of two-dimensional nonlinear Poisson-Boltzmann equations with five different fourth-order compact difference schemes are conducted to demonstrate the new ECNMG algorithm achieve sixth-order accuracy and keep less cost simultaneously, more efficient than the existing Newton-MG method.

KEYWORDS: Extrapolation, multigrid algorithm, sixth-order solution, fourth-order compact difference scheme, nonlinear Poisson equation

MSC CLASSIFICATION: 65N12; 65N30; 65N55;

1 INTRODUCTION

Larger scale, high efficiency, high accuracy and so on are the main difficulties in modern science and engineering computations fields. Multigrid method is one of the most efficient approaches for the larger scale discrete systems of partial differential equations. Extrapolation strategy is a well-known numerical tool to improve the approximation quality for partial differential equations. In recent years, some efficient multigrid methods combined with extrapolation techniques [4–6, 10, 12–19, 23–26, 33] have established for solving some partial differential equations. An extrapolation cascadic multigrid (EXCMG) method is first developed for the second-order linear elliptic problems by Chen and his collaborators [4] in 2008. Thereafter, the EXCMG method has been successfully extended for many other cases [5, 12–14, 24–26], such as non-smooth elliptic problems [12], linear parabolic problems [13], fractional diffusion equations [26], and some other related linear problems [14, 25]. Moreover, Pan [23] and Li [15, 16] studied some EXCMG methods combined with the high order (fourth-order and sixth-order) compact difference schemes to solve linear Poisson equations. During the same period, Wang and Zhang first design a multiscale multigrid (MSMG) method for linear Poisson equation [32]. And then, Dai, Lin and Zhang further discussed the MSMG method with a completed Richardson extrapolation technique for anisotropic linear Poisson equation [6]. Besides, Dai and his collaborators applied the EXCMG to provide an initial guess for the originally MSMG method, and constructed an EXCMG accelerated multiscale multigrid (EXCMG-MSMG) method [7] for accelerating the whole computational process. A significant amount of numerical results demonstrate these extrapolation multigrid methods (including EXCMG, MSMG and EXCMG-MSMG) are cost-effective approaches, which achieve the high accuracy and high efficiency simultaneously for linear discrete equations. However, compared with linear problems, these efficient extrapolation multigrid methods are seldom discussed for nonlinear problems. Therefore it will be interesting to further extend these idea for nonlinear cases.

Numerical solutions of the Poisson equations play an important role in fields of mechanical engineering and electrostatics. In this paper, we try to design an efficient numerical algorithm, i.e., an extrapolation cascadic Newton multigrid (ECNMG) method combined with the fourth-order compact difference schemes, for computing a high accuracy numerical solution of the Dirichlet boundary value problems of the two-dimensional (2D) nonlinear Poisson equations. In our computation, the problem domain is discretized by regular grids, and five nine-point fourth-order compact difference schemes are employed to discretized by the 2D nonlinear Poisson equation. By applying the extrapolation interpolation technique and a bi-quartic Lagrange interpolation for the approximations from two-level of grids (current and previous grids), we are able to obtain a much better initial guess for accelerating the Newton multigrid computation on an approximation solution. Additionally, a series of grid level dependent computational tolerances are taken in the Newton multigrid on different grids to generate conveniently the numerical solutions with discretization-level accuracy for extrapolations. Moreover, when the analytic solution is sufficiently smooth, a simple completed extrapolation strategy [6,23] is applied for two fourth-order approximations on two different scale grids (current and previous grids), and constructed a sixth-order accurate solution on the entire current grid cheaply and directly. Numerical experiments are conducted to demonstrate the superiority of our ECNMG algorithm, which can achieve sixth-order accuracy and keep less cost simultaneously.

The rest of the paper is organized as follows. Section 2 introduces five nine-point fourth-order compact difference schemes. Section 3 describes the proposed ECNMG method. Supporting numerical results are reported in Section 4. Concluding remarks are given in Section 5.

2 MODEL PROBLEM AND DISCRETIZATION

In this paper, we consider the two dimensional (2D) nonlinear Poisson equation in the form of

$$\Delta u(x, y) = f(x, y, u), \quad (x, y) \in \Omega, \tag{2.1}$$

with suitable Dirichlet boundary conditions on $\partial\Omega$. Here, the specified forcing function $f(x, y, u)$ as well as the unknown analytic solution $u(x, y)$ are assumed to be continuously differentiable and have the required partial derivatives on Ω.

To keep matters simple, assuming Ω is a rectangular $[D_a, D_b]\times[D_c, D_d]$, and subdividing it into a uniform grid Ω_h with uniform mesh-sizes $h_x = \frac{D_b-D_a}{N_x}$ and $h_y = \frac{D_d-D_c}{N_y}$. Here N_x and N_y are the number of uniform intervals in the x and the y coordinate directions, respectively. We take $U_{i,j}$ to represent approximations of the exact solution $u(x, y)$ at the mesh node (x_i, y_j) with $x_i = D_a + ih_x$ and $y_j = D_c + jh_y$, $(0 \le i \le N_x, 0 \le j \le N_y)$. Set $f_{i,j} = f(x_i, y_j, U_{i,j})$ and $\gamma = h_x/h_y$. Next, we introduce the following nine-point fourth-order compact schemes for the model problem (2.1):

HOC4a (cf. Zhai et al. [34])

$$
\begin{aligned}
&(10-2\gamma^2)(U_{i+1,j}+U_{i-1,j})+(10\gamma^2-2)(U_{i,j+1}+U_{i,j-1})\\
&-(20+20\gamma^2)U_{i,j}+(1+\gamma^2)(U_{i+1,j+1}+U_{i+1,j-1}+U_{i-1,j+1}+U_{i-1,j-1})\\
=&h_x^2[\frac{1}{12}(f_{i+1,j+1}+f_{i-1,j+1}+f_{i+1,j-1}+f_{i-1,j-1})+\frac{5}{6}(f_{i+1,j}+f_{i-1,j}+f_{i,j+1}+f_{i,j-1})+\frac{25}{3}f_{i,j}].
\end{aligned}
\tag{2.2}
$$

HOC4b (cf. Refs. [21,30,33,35])

$$
\begin{aligned}
&(10-2\gamma^2)(U_{i+1,j}+U_{i-1,j})+(10\gamma^2-2)(U_{i,j+1}+U_{i,j-1})-(20+20\gamma^2)U_{i,j}+\\
&(1+\gamma^2)(U_{i+1,j+1}+U_{i+1,j-1}+U_{i-1,j+1}+U_{i-1,j-1})=h_x^2(f_{i+1,j}+f_{i-1,j}+f_{i,j+1}+f_{i,j-1}+8f_{i,j}).
\end{aligned}
\tag{2.3}
$$

HOC4c (cf. Refs. [21,31])

$$
\begin{aligned}
&(20-4\gamma^2)(U_{i+1,j}+U_{i-1,j})+(20\gamma^2-4)(U_{i,j+1}+U_{i,j-1})-(40+40\gamma^2)U_{i,j}+\\
&(2+2\gamma^2)(U_{i+1,j+1}+U_{i+1,j-1}+U_{i-1,j+1}+U_{i-1,j-1})\\
=&h_x^2[f_{i+1,j+1}+f_{i-1,j+1}+f_{i+1,j-1}+f_{i-1,j-1}+20f_{i,j}].
\end{aligned}
\tag{2.4}
$$

HOC4d (cf. Zhai et al. [34])

$$
\begin{aligned}
&(30-6\gamma^2)(U_{i+1,j}+U_{i-1,j})+(30\gamma^2-6)(U_{i,j+1}+U_{i,j-1})-(60+60\gamma^2)U_{i,j}+\\
&(3+3\gamma^2)(U_{i+1,j+1}+U_{i+1,j-1}+U_{i-1,j+1}+U_{i-1,j-1})\\
=&h_x^2(f_{i+1,j+1}+f_{i-1,j+1}+f_{i+1,j-1}+f_{i-1,j-1}+f_{i+1,j}+f_{i-1,j}+f_{i,j+1}+f_{i,j-1}+28f_{i,j}).
\end{aligned}
\tag{2.5}
$$

HOC4e (cf. Zhai et al. [34])

$$
\begin{aligned}
&(240-48\gamma^2)(U_{i+1,j}+U_{i-1,j})+(240\gamma^2-48)(U_{i,j+1}+U_{i,j-1})-(480+480\gamma^2)U_{i,j}+\\
&(24+24\gamma^2)(U_{i+1,j+1}+U_{i+1,j-1}+U_{i-1,j+1}+U_{i-1,j-1})\\
=&h_x^2[f_{i+1,j+1}+f_{i-1,j+1}+f_{i+1,j-1}+f_{i-1,j-1}+22(f_{i+1,j}+f_{i-1,j}+f_{i,j+1}+f_{i,j-1})+196f_{i,j}].
\end{aligned}
\tag{2.6}
$$

Taking one of the above fourth-order compact schemes (HOC4a, HOC4b, HOC4c, HOC4d, HOC4e) with mesh-size $h = max\{h_x, h_y\}$ for the model problem (2.1) on grid Ω_h, we obtain the corresponding nonlinear system

$$F(u_h) = 0, \tag{2.7}$$

where the notation $F(u_h)$ signifies that the operator is nonlinear.

Furthermore, the domain Ω also can be further subdivided into a sequence of unform grids Ω_j ($j = 1, 2, \ldots, L$) with mesh-size $h_j = 2^{1-j}h$. We can obtain the corresponding nonlinear equations

$$F(u_j) = 0, \quad j = 1, 2, \ldots, L. \tag{2.8}$$

Algorithm 1 Newton-MG method for solving $F(x) = 0$

input ϵ, k_{max} and x_0, set $k = 0$
for $k = 0, 1, \ldots, k_{max}$ **do**
  if $\|F(x_k)\| \le \epsilon$ **then** stop
  else
    construct Jacobian matrix $A \Leftarrow F'(x_0)$
    compute Jacobian system $As_k = -F(x_k)$ by MG V-cycles, get $\tilde{s}_k$
    if $\|\tilde{s}_k\| \ge \epsilon$ **then**
      update $x_{k+1} \Leftarrow x_k + \tilde{s}_k,\ k \Leftarrow k + 1$
    else stop
    end if
  end if
end for

3 EXTRAPOLATION CASCADIC NEWTON MULTIGRID COMPUTATION

The usual nonlinear numerical algorithms (such as Newton, Newton-like, nonlinear Gauss-Seidel relaxation) are widely used for the results nonlinear system from the fourth-order compact schemes, but the accuracy of computed approximation reach fourth-order at most. In order to obtain a sixth-order (rather than fourth-order) accurate numerical solution from the nonlinear system, we aim to construct an efficient high precision extrapolation cascadic Newton multigrid computation, which based on the so-called Newton multigrid method (Newton-MG) [2] and extrapolation techniques. Here we first introduce the Newton-MG method in the follow subsection.

3.1 CLASSICAL NEWTON-MG METHOD

It is well known that Newton's method converges quadratically. Multigrid method is one of the fastest and most efficient iterative methods for linear discrete system. In 1982, Dembo et al. applied classical V-cycle multigrid to solve the Jacobian system approximately in Newton iterations, and originally proposed the Newton-MG method [2]. This approach is also called an efficient inexact Newton's method, which takes advantages of Newton and multigrid methods to accelerate the nonlinear system solution. Numerical experiments of the nonlinear problems in Ref. [2] shows that the Newton-MG method is much more efficient than classical Newton's method and full approximation scheme (FAS).

To be precise, Newton-MG method with a given initial solution x_0 generates a sequence $\{x_k\}$ of approximations to the exact solution of nonlinear system $F(x) = 0$ as described in Algorithm 1.

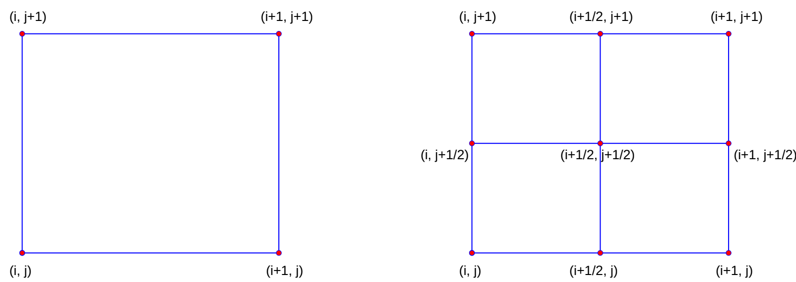
3.2 EXTRAPOLATION FOR INITIAL GUESS

As well as classical iterative algorithms, the k-th iterative solution x_k of Newton-MG method also satisfies the following inequation

$$\|x_k - x_*\| \le \theta^k \|x_0 - x_*\|. \tag{3.9}$$

Here, $\theta \in (0, 1)$ is convergence factor of the Newton-MG method, x_0 is an arbitrary initial value, and x_* is the exact solution of nonlinear system $F(x) = 0$.

From Eq. (3.9), we can imagine that if the initial error $\|x_0 - x_*\|$ is small enough, which will effectively reduce the number of iterations and accelerate the convergence of Newton-MG method. Therefore, we aim



(a) Coarse grid Ω_h (b) Fine grid $\Omega_{h/2}$

Figure 1: Illustration of cells on the coarse and fine grids.

to construct a better initial guess for accelerating the Newton-MG method, by employing some interpolation techniques. Classical Richardson extrapolation is a well-known numerical strategy to eliminate certain error terms and thus to improve the approximation quality. In the last decade, some extrapolation interpolation techniques have been developed to provide better initial guesses for accelerating multigrid methods [4, 14, 16, 23]. Here, we demonstrate how to construct an initial guess from an extrapolation interpolation and a bi-quartic polynomial interpolation operators [23] with fourth-order compact schemes. To recapitulate briefly, the extrapolation interpolation formulas [23] are employed to construct high accuracy extrapolation solution on the entire fine grid (mesh-size $h/2$), by combining two fourth-order accurate solutions of the fine (mesh-size $h/2$) and the coarse (mesh-size h) grids.

To better illustrate the extrapolation interpolation formulas, we take a coarse cell $I_{i,j} = [x_i, x_{i+1}]\times[y_j, y_{j+1}]$ of grid Ω_h (Figure 1(a)), which include nine fine grid nodes (Figure 1(b))

$$(x_s, y_t), (x_{i+1/2}, y_t), (x_s, y_{j+1/2}), (x_{i+1/2}, y_{j+1/2}), \quad s = i, i+1, t = j, j+1.$$

Assume these nodal solutions

$$\begin{aligned}
&\left\{u^h_{s,t}, \quad s = i, i+1, \ \ t = j, j+1\right\} \quad \text{on } \Omega_h,\\
&\left\{u^{h/2}_{s,t}, \ \ u^{h/2}_{i+1/2,t}, \ \ u^{h/2}_{s,j+1/2}, \quad s = i, i+1, \ \ t = j, j+1\right\} \quad \text{on } \Omega_{h/2}.
\end{aligned}$$

have been provided, the extrapolation approximations

$$\{Nu^{h/2}_{s,t}, Nu^{h/2}_{i+1/2,t}, Nu^{h/2}_{s,j+1/2}, Nu^{h/2}_{i+1/2,j+1/2}\} \tag{3.10}$$

can be constructed at these nodes (x_s, y_t), $(x_{i+1/2}, y_t)$, $(x_s, y_{j+1/2})$, $(x_{i+1/2}, y_{j+1/2})$ of the fine grid $\Omega_{h/2}$ with mesh-size $h/2$, through applying the extrapolation interpolation formulas [23], which can be described as follow

$$Nu^{h/2}_{s,t} = \frac{17}{16}u^{h/2}_{s,t} - \frac{1}{16}u^h_{s,t}, \tag{3.11}$$

$$Nu^{h/2}_{i+1/2,t} = u^{h/2}_{i+1/2,t} + \frac{1}{32}[(u^{h/2} - u^h)_{i,t} + (u^{h/2} - u^h)_{i+1,t}], \tag{3.12}$$

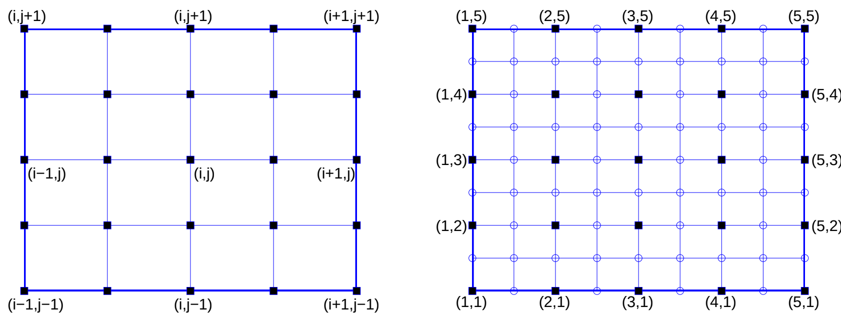
$$Nu^{h/2}_{s,j+1/2} = u^{h/2}_{s,j+1/2} + \frac{1}{32}[(u^{h/2} - u^h)_{s,j} + (u^{h/2} - u^h)_{s,j+1}], \tag{3.13}$$

and

$$
\begin{aligned}
Nu^{h/2}_{i+1/2,j+1/2} =& u^{h/2}_{i+1/2,j+1/2} + \frac{1}{64}[(u^{h/2}-u^h)_{i,j} + (u^{h/2}-u^h)_{i+1,j} \\
&+ (u^{h/2}-u^h)_{i,j+1} + (u^{h/2}-u^h)_{i+1,j+1}].
\end{aligned} \tag{3.14}
$$

Similarly, applying the above extrapolation interpolation formulas for the two solutions u^h, $u^{h/2}$ on the entire coarse and fine grids, we immediately obtain the extrapolation solution $Nu^{h/2}$ on the global fine grid $\Omega_{h/2}$. We denote this process as

$$
Nu^{h/2} = NE(u^h, u^{h/2}). \tag{3.15}
$$



(a) Sixteen neighboring elements on $\Omega_{h/2}$

(b) Refined elements on $\Omega_{h/4}$

Figure 2: One interpolation cell which contains sixteen neighboring elements on $\Omega_{h/2}$.

Next, we discuss how to construct the bi-quartic polynomial interpolation operator [23] for these extrapolation solutions values on the fine grid $\Omega_{h/2}$, and interpolate initial guess on the refined grid $\Omega_{h/4}$. As argued previously, we explain the process of this operator on one interpolation cell which contains 16 neighboring elements of fine grid $\Omega_{h/2}$ or 64 neighboring elements of refined grid $\Omega_{h/4}$ (see Figure 2). To keep matters simple, we relabel the refined grid nodes as (s, t), which represents the label of the grid nodes (ξ_s, η_t) $(s, t = 1, 2, \cdots, 5)$, the big black squares denote the grid points on $\Omega_{h/2}$, and the small circles denote the grid points on $\Omega_{h/4}$ which do not belong to $\Omega_{h/2}$ (see Figure 2 (b)). The bi-quartic Lagrange interpolation function in terms of natural coordinates (xi_s, η_t) is

$$
L(\xi, \eta) = \sum_{m=1}^{25} W_m(\xi, \eta) u_m, \tag{3.16}
$$

where the basis functions $W_m(\xi, \eta)$ can be written as

$$
W_m(\xi, \eta) = l_s(\xi) l_t(\eta). \tag{3.17}
$$

Here, $l_s(\xi)$, $l_t(\eta)$ $(s, t = 1, 2, \cdots, 5)$ are the Lagrange basis polynomials of degree 4, defined as

$$l_s(\xi) = \prod_{k=1,k\neq s}^{5} \frac{\xi - \xi_k}{\xi_s - \xi_k}, \tag{3.18}$$

$$l_t(\eta) = \prod_{k=1,k\neq t}^{5} \frac{\eta - \eta_k}{\eta_t - \eta_k}, \tag{3.19}$$

and (ξ_s, η_t) is the natural coordinate of node $m(1 \leq m \leq 25)$.

Using the bi-quartic interpolation Eq. (3.16) for the computed solution at the 25 big black squares in Figure 2 (b), we can obtain the initial values at other 56 $(9^2 - 5^2)$ small circles on the interpolation cell. With the similar discussion, we can provide an initial guess $u^{h/4,0}$ on the refined grid $\Omega_{h/4}$, by constructing the bi-quartic polynomial interpolation operator $P_{h/2}^{h/4}$ for the known solution $Nu^{h/2}$ on the entire grid $\Omega_{h/2}$. We denote the procedure of this interpolation operator as the form

$$u^{h/4,0} = P_{h/2}^{h/4} N u^{h/2}. \tag{3.20}$$

3.3 EXTRAPOLATION FOR SIXTH-ORDER APPROXIMATION

Richardson extrapolation technique [27,28] is a powerful tool to enhance the accuracy of approximation solution, which takes separate second-order solutions on a fine gird and on the subgrid formed of alternate points, and incorporates them to calculate a fourth-order solution on the subgrid. In 1983, Marchuk and Shaidurov extended this approach to finite difference method [22]. Blum et al. discussed this strategy in conjunction with the finite element method [1, 3, 11, 20]. Roache and Knupp improved the classical Richardson extrapolation strategy, and developed a completed extrapolation technique [29] for second-order difference scheme, which produces a fourth-order accurate solution on all the fine grid points by combining second-order solutions on the fine and the coarse grids. Borrowing the idea, Dai, Pan and his collaborators studied a completed Richardson extrapolation operator [6, 23] combined with fourth-order compact difference scheme to obtain a sixth-order solution on the fine grid.

In this paper, we shall take the completed extrapolation strategy [6, 23] to enhance the accuracy of approximation of nonlinear system. Again, we choose a coarse grid cell $I_{i,j} = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$ of grid Ω_h, to introduce the completed extrapolation operator [6, 23]. Assume the coarse grid nodal values $u_{s,t}^h$ and the fine grid nodal values $u_{s,t}^{h/2}$, $u_{i+1/2,t}^{h/2}$, $u_{s,j+1/2}^{h/2}$ with $s = i, i+1$; $t = j, j+1$ on the cell have been computed. The completed extrapolation formulas [6, 23] can be rewritten explicitly as below

$$Eu_{s,t}^{h/2} = \frac{16}{15}u_{s,t}^{h/2} - \frac{1}{15}u_{s,t}^{h}, \tag{3.21}$$

$$Eu_{i+1/2,t}^{h/2} = u_{i+1/2,t}^{h/2} + \frac{1}{30}[(u^{h/2} - u^h)_{i,t} + (u^{h/2} - u^h)_{i+1,t}], \tag{3.22}$$

$$Eu_{s,j+1/2}^{h/2} = u_{s,j+1/2}^{h/2} + \frac{1}{30}[(u^{h/2} - u^h)_{s,j} + (u^{h/2} - u^h)_{s,j+1}], \tag{3.23}$$

and

$$\begin{aligned} Eu_{i+1/2,j+1/2}^{h/2} =& u_{i+1/2,j+1/2}^{h/2} + \frac{1}{60}[(u^{h/2} - u^h)_{i,j} + (u^{h/2} - u^h)_{i+1,j} \\ &+ (u^{h/2} - u^h)_{i,j+1} + (u^{h/2} - u^h)_{i+1,j+1}]. \end{aligned} \tag{3.24}$$

Employing the above extrapolation formulas for the fourth-order accurate solutions (u^h and $u^{h/2}$) on two

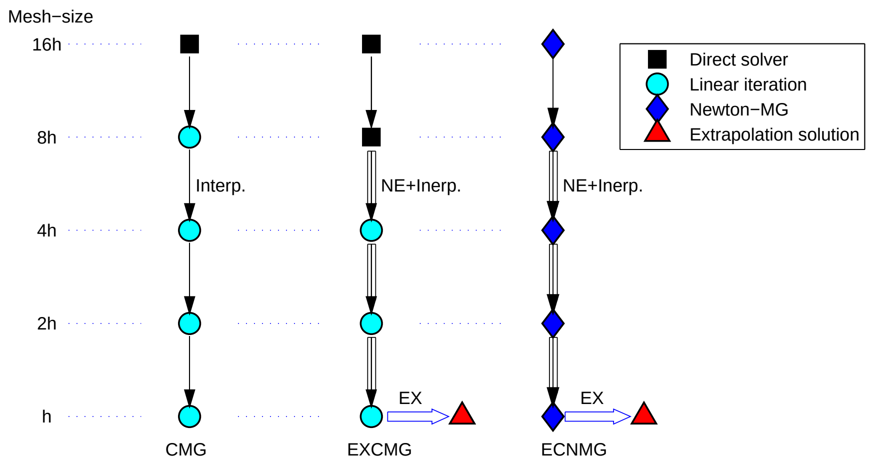


Figure 3: Illustration of the CMG, the EXCMG and the proposed ECNMG methods ($Interp.$ denotes interpolation, NE denotes extrapolation interpolation formulas, EX denotes completed extrapolation)

different scale grids (Ω_h and $\Omega_{h/2}$), we can obtain an extrapolation solution $Eu^{h/2}$ on the global fine grid $\Omega_{h/2}$. Similarly, we denote this procedure from u^h, $u^{h/2}$ to $Eu^{h/2}$ as a completed extrapolation operator, and call it as Ex, i.e.

$$Eu^{h/2} \Leftarrow EX(u^h, u^{h/2}). \tag{3.25}$$

Remark 1 *An observation is in order, the extrapolation interpolation NE (in previous subsection) and the completed extrapolation Ex (in current subsection) are totally different.*

(1) They have significantly different parameters. See the expressions of NE (Eqs. (3.11)-(3.14)) and EX (Eqs. (3.21)-(3.24)) for details.

(2) They play different roles. The extrapolation interpolation NE is combined with a bi-quartic polynomial interpolation for providing a good initial guess, which is superclose to the exact difference solution of nonlinear system (rather than the analytic solution u of model problem). The completed extrapolation EX is adopted to construct an improved solution, which is a sixth-order approximation of the analytic solution u on the entire fine grid of model problem.

3.4 DESCRIPTION OF THE ECNMG COMPUTATION

In this subsection, we try to design an cost-efficient numerical approach, i.e., extrapolation cascadic Newton multigrid (ECNMG) computation, for computing a sixth-order accurate approximation from the discrete system. Roughly speaking, we take the Newton-MG method as a basic nonlinear iterative solver, which embedded the better initial guesses and a series of grid level dependent computational tolerances to obtain the corresponding fourth-order accurate approximations of the nonlinear systems with less computational cost. We also choose the completed extrapolation strategy to enhance the accuracy of approximation of nonlinear system directly and cheaply. The process of this method is described in Algorithm 2, and the structure diagram of the ECNMG method is given in Figure 3. The key ingredients of the proposed algorithm include the following aspects.

Algorithm 2 Extrapolation cascadic Newton multigrid (ECNMG)

run the Newton-MG solver for computing two approximations U_1^*, U_2^* on the grids Ω_1, Ω_2, respectively
for $j = 3, 4, \cdots, L$ **do**
    use the extrapolation interpolation operator NE to update solution, $NU_{j-1} \Leftarrow NE_{j-2}^{j}(U_{j-2}^*, U_{j-1}^*)$
    use the bi-quartic interpolation operator P_{j-1}^{j} to construct an initial guess, $U_j^0 \Leftarrow P_{j-1}^{j} NU_{j-1}$
    run the Newton-MG solver with U_j^0 and tolerances ϵ_j for computing an approximation U_j^* on grid Ω_j
    use the completed extrapolation operator Ex to obtain an extrapolation solution, $Eu_j \Leftarrow Ex(U_{j-1}^*, U_j^*)$
end for

(1) A better initial guess is provided for Newton-MG method to accelerate the computation process of nonlinear system on the next finer grid, through employing the extrapolation interpolation technique and the bi-quartic polynomial interpolation [23] for the known approximations on previous and current grids.

(2) Instead of performing a fixed number of multigrid cycles as used in classical multigrid structure, a series of grid level dependent computational tolerances are discussed for reducing the iterations of nonlinear solver on different scale fourth-order solutions.

(3) A sixth-order (rather than fourth-order) accurate approximation is generated on current grid cheaply and directly, by employing a completed extrapolation operator [6, 23] for the fourth-order accuracy solutions of discrete nonlinear systems on current and previous grids.

The computational tolerances ϵ_j in ECNMG method is chosen as below

$$\epsilon_j = \begin{cases} 10^{j-L}\epsilon_L, & j \geq \bar{j}, \\ 10^{\bar{j}-L}\epsilon_L, & j < \bar{j}, \end{cases} \tag{3.26}$$

where, L is the number of grid levels, ϵ_L is a given computational tolerance on the finest grid, threshold value $\bar{j} = [L/2]$.

4 NUMERICAL RESULTS

In our numerical experiments, we tested the ECNMG method for two nonlinear Poisson-Boltzmann equations on the unit square domain $\Omega = [0, 1] \times [0, 1]$, and compared the results with the Newton-MG method [2]. In these cases, the source terms $f(x, y)$ and Dirichlet boundary conditions on $\partial\Omega$ are presented to satisfy the known analytic solutions.

The classical V-cycle multigrid method embedded with the conjugate gradient relaxation is applied as linear solver for solving the Jacobian systems in the Newton-MG method. In the V-cycle multigrid, the numbers of pre-smoothing and post-smoothing both set as 1. Zero vector is chosen as an initial iterative guess to start with the two methods (Newton-MG and ECNMG), which will be terminated when the k-th iterative solution u_j^k satisfies

$$\max\{\ \|F(u_j^k)\|,\ \ \|u_j^k - u_j^{k-1}\|\ \} \leq \epsilon_j$$

on the grid Ω_j.

Using 8 embedded grids with the coarsest grid 16×16, and the finest grid is 2048×2048 which leads to 4 million unknowns, we perform numerical computation using the ECNMG and the Newton-MG methods with five different fourth-order compact difference schemes (HOC4a, HOC4b, HOC4c, HOC4d, HOC4e), respectively. The iteration numbers (#I), initial guess errors ($\|U_j^0 - u\|_2$), approximation errors ($\|U_j^* - u\|_2$), extrapolation solution errors ($\|Eu_j - u\|_2$), and convergence rates ($\#R_j$) from the 3-th level of grid 64×64 to the 8-th grid 2048×2048 are given in the Tables 1- 10. Here, the convergence rate is defined as

$$\#R_j = \frac{\log(\|e_{j-1}\|/\|e_j\|)}{\log(h_{j-1}/h_j)}.$$

where $\|e_j\|$ is the L^2-norm error on the j-th level grid with mesh-size h_j.

Example 4.1 *[19] Consider the nonlinear Poisson-Boltzmann equation in the presence of source term*

$$\lambda^2(u_{xx}(x,y) + u_{yy}(x,y)) + \exp(-u(x,y)) = f(x,y), \tag{4.27}$$

in which, the exact solution is

$$u(x,y) = \exp(y-x) + 2^{-1/\sigma}(1+y)^{1+1/\sigma}. \tag{4.28}$$

Here, $\lambda = 0.005$, $\sigma = 0.01$.

The numerical experiment of Example 4.1 with five different schemes (HOC4a, HOC4b, HOC4c, HOC4d, HOC4e) are given in Tables 1 to 5. In the respect of accuracy, we can see from Tables 1 to 5 that the two methods are both able to obtain full fourth-order accurate difference solutions (see the sixth and the eleventh columns for details). But the convergence rate of extrapolation solutions in the proposed method are close to theoretical value 6 (see the ninth column for details). Hence, fewer grid points are required for the ECNMG algorithm than those for the Newton-MG algorithm to achieve a certain computational accuracy. For instance, from the last rows of Table 1 for Example 4.1 with HOC4a scheme, we see that the approximation error of Newton-MG method on the finest grid 2048×2048 is 7.21×10^{-11}, while the extrapolation solution error of the ECNMG algorithm on the 1024×1024 grid is 3.14×10^{-12}. Besides, we given Figure 4 to describe the errors of the initial guesses, the difference approximations and the extrapolation solutions from the 3-th level to the final grid in the ECNMG method. It shows that the precision of extrapolation solutions Eu_j^* are more than the difference approximations U_j^* (before extrapolated) and the initial values U_j^0 on the j-th ($j = 3, 4, \dots, 8$) grid level. When the reciprocal of mesh-size (i.e., $1/h_j$) increases, the superiority of the new method on accuracy is more obvious.

As for computational cost, one can see from Tables 1 to 5 that there is only a few iterations are required on each grid level of our ECNMG method for Example 4.1 with different schemes, compared to the Newton-MG method. This is particularly true when mesh is refined. The reason for this phenomenon is that we taken the extrapolation interpolation and the bi-quartic polynomial interpolation techniques in our proposed method, to provide the fifth-order accurate initial guesses, which are already extremely accurate approximations on each grid level (see the fifth column for details). Particularly for large scale grid levels, the given initial guesses are close enough to the difference solutions (see Figure 4. Taking Table 1 as an example, only two times iterations are required on the finest grid 2048×2048 in our ECNMG method, since the error of the initial guess is 8.01×10^{10}, while the error of the difference solution is 7.21×10^{11}. It is well known that when the scale of grid increases, decreasing iterations on the finest grid will effectively reduce the total computational cost. Therefore, the ECNMG method spent less total computational cost than that of the Newton-MG method for solving large scale problems.

Moreover, Tables 1 to 5 demonstrate that the proposed indeed works well (accuracy and cost) for the given different schemes (HOC4a, HOC4b, HOC4c, HOC4d and HOC4e).

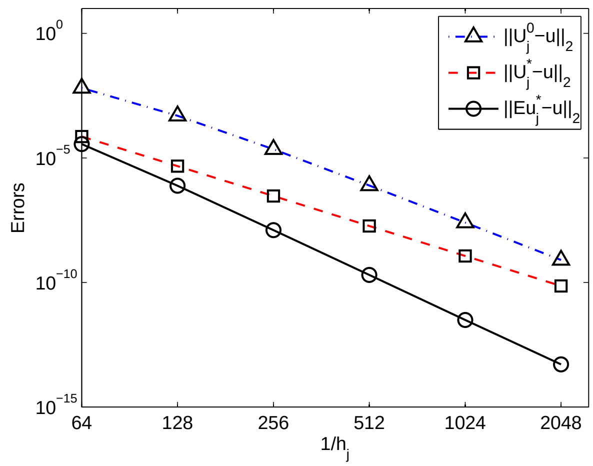


Figure 4: Comparison of the L_2-norm errors of the initial guesses U_j^0, the difference solutions U_j^* and the extrapolation approximations Eu_j on different grid levels with mesh-size h_j (Example 4.1, HOC4a).

Table 1: Numerical results of Example 4.1 with HOC4a

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|\|U_j^0-u\|\|_2$	$\#R_j$	$\|\|U_j^*-u\|\|_2$	$\#R_j$	$\|\|Eu_j-u\|\|_2$	$\#R_j$	$\#I$	$\|\|U_j^*-u\|\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	6.45(-3)	–	7.24(-5)	–	3.59(-5)	–	6	7.24(-5)	–	11
4	128	10^{-11}	4.93(-4)	3.71	4.69(-6)	3.95	7.61(-7)	5.56	7	4.69(-6)	3.95	12
5	256	10^{-10}	2.23(-5)	4.46	2.95(-7)	3.99	1.26(-8)	5.91	5	2.95(-7)	3.99	11
6	512	10^{-9}	7.86(-7)	4.83	1.85(-8)	4.00	2.00(-10)	5.98	4	1.85(-8)	4.00	11
7	1024	10^{-8}	2.54(-8)	4.95	1.15(-9)	4.00	3.14(-12)	5.99	3	1.15(-9)	4.00	10
8	2048	10^{-7}	8.01(-10)	4.98	7.21(-11)	4.00	5.15(-14)	5.93	2	7.21(-11)	4.00	10

Table 2: Numerical results of Example 4.1 with HOC4b

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|\|U_j^0-u\|\|_2$	$\#R_j$	$\|\|U_j^*-u\|\|_2$	$\#R_j$	$\|\|Eu_j-u\|\|_2$	$\#R_j$	$\#I$	$\|\|U_j^*-u\|\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	6.45(-3)	–	7.24(-5)	–	3.58(-5)	–	6	7.24(-5)	–	11
4	128	10^{-11}	4.93(-4)	3.71	4.69(-6)	3.95	7.61(-7)	5.56	6	4.69(-6)	3.95	11
5	256	10^{-10}	2.23(-5)	4.46	2.95(-7)	3.99	1.26(-8)	5.91	5	2.95(-7)	3.99	11
6	512	10^{-9}	7.86(-7)	4.83	1.85(-8)	4.00	2.00(-10)	5.98	4	1.85(-8)	4.00	11
7	1024	10^{-8}	2.54(-8)	4.95	1.15(-9)	4.00	3.14(-12)	5.99	3	1.15(-9)	4.00	10
8	2048	10^{-7}	8.01(-10)	4.98	7.21(-11)	4.00	5.15(-14)	5.93	2	7.21(-11)	4.00	10

Table 3: Numerical results of Example 4.1 with HOC4c

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0-u\|_2$	$\#R_j$	$\|U_j^*-u\|_2$	$\#R_j$	$\|Eu_j-u\|_2$	$\#R_j$	$\#I$	$\|U_j^*-u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	6.45(-3)	–	7.24(-5)	–	3.59(-5)	–	7	77.24(-5)	–	11
4	128	10^{-11}	4.93(-4)	3.71	4.69(-6)	3.95	7.61(-7)	5.56	7	74.69(-6)	3.95	12
5	256	10^{-10}	2.23(-5)	4.46	2.95(-7)	3.99	1.26(-8)	5.91	6	72.95(-7)	3.99	11
6	512	10^{-9}	7.86(-7)	4.83	1.85(-8)	4.00	2.00(-10)	5.98	4	71.85(-8)	4.00	11
7	1024	10^{-8}	2.54(-8)	4.95	1.15(-9)	4.00	3.14(-12)	5.99	3	71.15(-9)	4.00	10
8	2048	10^{-7}	8.01(-10)	4.98	7.21(-11)	4.00	5.15(-14)	5.93	2	77.21(-11)	4.00	10

Table 4: Numerical results of Example 4.1 with HOC4d

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0-u\|_2$	$\#R_j$	$\|U_j^*-u\|_2$	$\#R_j$	$\|Eu_j-u\|_2$	$\#R_j$	$\#I$	$\|U_j^*-u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	6.45(-3)	–	7.24(-5)	–	3.59(-5)	–	7	7.24(-5)	–	12
4	128	10^{-11}	4.93(-4)	3.71	4.69(-6)	3.95	7.61(-7)	5.56	7	4.69(-6)	3.95	12
5	256	10^{-10}	2.23(-5)	4.46	2.95(-7)	3.99	1.26(-8)	5.91	6	2.95(-7)	3.99	11
6	512	10^{-9}	7.86(-7)	4.83	1.85(-8)	4.00	2.00(-10)	5.98	4	1.85(-8)	4.00	11
7	1024	10^{-8}	2.54(-8)	4.95	1.15(-9)	4.00	3.14(-12)	5.99	3	1.15(-9)	4.00	10
8	2048	10^{-7}	8.01(-10)	4.98	7.21(-11)	4.00	5.17(-14)	5.93	2	7.21(-11)	4.00	10

Table 5: Numerical results of Example 4.1 with HOC4e

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0-u\|_2$	$\#R_j$	$\|U_j^*-u\|_2$	$\#R_j$	$\|Eu_j-u\|_2$	$\#R_j$	$\#I$	$\|U_j^*-u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	6.45(-3)	–	7.24(-5)	–	3.59(-5)	–	6	7.24(-5)	–	11
4	128	10^{-11}	4.93(-4)	3.71	4.69(-6)	3.95	7.61(-7)	5.56	7	4.69(-6)	3.95	11
5	256	10^{-10}	2.23(-5)	4.46	2.95(-7)	3.99	1.26(-8)	5.91	5	2.95(-7)	3.99	11
6	512	10^{-9}	7.86(-7)	4.83	1.85(-8)	4.00	2.00(-10)	5.98	4	1.85(-8)	4.00	11
7	1024	10^{-8}	2.54(-8)	4.95	1.15(-9)	4.00	3.15(-12)	5.99	3	1.15(-9)	4.00	10
8	2048	10^{-7}	8.01(-10)	4.98	7.22(-11)	4.00	2.52(-13)	3.64	2	7.22(-11)	4.00	10

Example 4.2 *[19] Consider the following nonlinear Poisson-Boltzmann equation*

$$u_{xx}(x,y) + u_{yy}(x,y) - \sigma^2 u(x,y)^3 = f(x,y) \tag{4.29}$$

with the exact solution

$$u(x,y) = x(1-x)y(1-y)sin(k^2(x+\sqrt{3}y)). \tag{4.30}$$

Here, $k = 4$, $\sigma = 0.01$.

Table 6: Numerical results of Example 4.2 with HOC4a

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0 - u\|_2$	$\#R_j$	$\|U_j^* - u\|_2$	$\#R_j$	$\|Eu_j^* - u\|_2$	$\#R_j$	$\#I$	$\|U_j^* - u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	1.98(-4)	–	3.31(-6)	–	2.39(-7)	–	6	3.31(-6)	–	7
4	128	10^{-11}	6.67(-6)	4.89	2.06(-7)	4.01	3.73(-9)	6.00	6	2.06(-7)	4.01	7
5	256	10^{-10}	2.13(-7)	4.97	1.28(-8)	4.00	5.83(-11)	6.00	5	1.28(-8)	4.00	7
6	512	10^{-9}	6.72(-9)	4.99	8.01(-10)	4.00	9.10(-13)	6.00	3	8.01(-10)	4.00	7
7	1024	10^{-8}	2.15(-10)	4.97	5.01(-11)	4.00	1.86(-14)	5.61	2	5.01(-11)	4.00	6
8	2048	10^{-7}	7.24(-12)	4.89	3.14(-12)	4.00	9.54(-15)	0.96	1	3.13(-12)	4.00	6

Table 7: Numerical results of Example 4.2 with HOC4b

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0 - u\|_2$	$\#R_j$	$\|U_j^* - u\|_2$	$\#R_j$	$\|Eu_j^* - u\|_2$	$\#R_j$	$\#I$	$\|U_j^* - u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	1.94(-4)	–	1.59(-6)	–	1.12(-7)	–	6	1.59(-6)	–	7
4	128	10^{-11}	6.61(-6)	4.87	9.80(-8)	4.02	1.72(-9)	6.02	6	9.80(-8)	4.02	7
5	256	10^{-10}	2.11(-7)	4.97	6.11(-9)	4.00	2.69(-11)	6.00	4	6.11(-9)	4.00	7
6	512	10^{-9}	6.65(-9)	4.99	3.81(-10)	4.00	4.19(-13)	6.00	3	3.81(-10)	4.00	7
7	1024	10^{-8}	2.09(-10)	4.99	2.38(-11)	4.00	1.07(-14)	5.29	2	2.38(-11)	4.00	6
8	2048	10^{-7}	6.66(-12)	4.97	1.50(-12)	3.99	9.49(-15)	0.18	1	1.49(-12)	4.00	6

Table 8: Numerical results of Example 4.2 with HOC4c

j	$1/h_j$	ϵ_j	ECNMG							Newton-MG		
			$\|U_j^0 - u\|_2$	$\#R_j$	$\|U_j^* - u\|_2$	$\#R_j$	$\|Eu_j^* - u\|_2$	$\#R_j$	$\#I$	$\|U_j^* - u\|_2$	$\#R_j$	$\#I$
3	64	10^{-11}	2.30(-4)	–	1.41(-5)	–	1.22(-6)	–	7	1.41(-5)	–	7
4	128	10^{-11}	7.23(-6)	4.99	8.79(-7)	4.00	1.93(-8)	5.98	6	8.79(-7)	4.00	7
5	256	10^{-10}	2.30(-7)	4.97	5.49(-8)	4.00	3.03(-10)	5.99	5	5.49(-8)	4.00	7
6	512	10^{-9}	7.77(-9)	4.89	3.43(-9)	4.00	4.74(-12)	6.00	3	3.43(-9)	4.00	7
7	1024	10^{-8}	3.06(-10)	4.67	2.15(-10)	4.00	7.46(-14)	5.99	2	2.15(-10)	4.00	6
8	2048	10^{-7}	1.50(-11)	4.34	1.34(-11)	4.00	9.56(-15)	2.96	1	1.34(-11)	4.00	6

Again, we use 8 level of grids with the finest grid 2048×2048 for Example 4.2, and list the experimental results (from the 3−th level grid to the final one) of Newton-MG and ECNMG methods with different compact

Table 9: Numerical results of Example 4.2 with HOC4d

j	$1/h_j$	ϵ_j	ECNMG $\Vert U_j^0-u\Vert_2$	$\#R_j$	$\Vert U_j^*-u\Vert_2$	$\#R_j$	$\Vert Eu_j^*-u\Vert_2$	$\#R_j$	$\#I$	Newton-MG $\Vert U_j^*-u\Vert_2$	$\#R_j$	$\#I$
3	64	10^{-11}	2.15(-4)	–	9.72(-6)	–	8.19(-7)	–	6	9.72(-6)	–	7
4	128	10^{-11}	6.96(-6)	4.95	6.06(-7)	4.00	1.30(-8)	5.98	6	6.06(-7)	4.00	7
5	256	10^{-10}	2.21(-7)	4.97	3.79(-8)	4.00	2.04(-10)	5.99	5	3.79(-8)	4.00	7
6	512	10^{-9}	7.22(-9)	4.94	2.37(-9)	4.00	3.18(-12)	6.00	3	2.37(-9)	4.00	7
7	1024	10^{-8}	2.60(-10)	4.80	1.48(-10)	4.00	5.14(-14)	5.95	2	1.48(-10)	4.00	6
8	2048	10^{-7}	1.14(-11)	4.51	9.25(-12)	4.00	9.49(-15)	2.44	1	9.24(-12)	4.00	6

Table 10: Numerical results of Example 4.2 with HOC4e

j	$1/h_j$	ϵ_j	ECNMG $\Vert U_j^0-u\Vert_2$	$\#R_j$	$\Vert U_j^*-u\Vert_2$	$\#R_j$	$\Vert Eu_j^*-u\Vert_2$	$\#R_j$	$\#I$	Newton-MG $\Vert U_j^*-u\Vert_2$	$\#R_j$	$\#I$
3	64	10^{-11}	1.98(-4)	–	3.31(-6)	–	2.39(-7)	–	6	3.31(-6)	–	7
4	128	10^{-11}	6.67(-6)	4.89	2.06(-7)	4.01	3.73(-9)	6.00	6	2.06(-7)	4.01	7
5	256	10^{-10}	2.13(-7)	4.97	1.28(-8)	4.00	5.83(-11)	6.00	5	1.28(-8)	4.00	7
6	512	10^{-9}	6.72(-9)	4.99	8.01(-10)	4.00	9.10(-13)	6.00	3	8.01(-10)	4.00	7
7	1024	10^{-8}	2.15(-10)	4.97	5.01(-11)	4.00	1.86(-14)	5.61	2	5.01(-11)	4.00	6
8	2048	10^{-7}	7.24(-12)	4.89	3.14(-12)	4.00	9.52(-15)	0.97	1	3.13(-12)	4.00	6

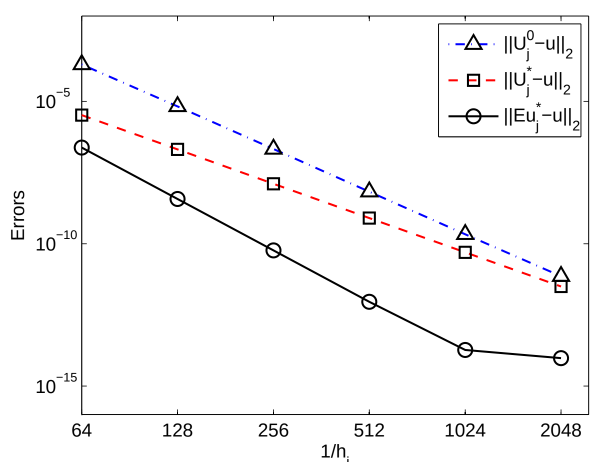


Figure 5: Comparison of the L_2-norm errors of the initial guesses U_j^0, the difference solutions U_j^* and the extrapolation approximations Eu_j on different grid levels with mesh-size h_j (Example 4.2, HOC4a).

schemes in Tables 6 to 10. We can find that the errors of fourth-order solutions from the Newton-MG method are exactly the same as the errors of fourth-order solutions from the ECNMG method. But, they are much less accurate than the extrapolation solutions Eu_j produced by the completed extrapolation operator in the ECNMG method. The extrapolated solutions Eu_j achieves the sixth-order precision on most grids and starts to lose convergent order on the final grid, since the approximation on the final grid already has sufficient accuracy $O(10^{-15})$. Meanwhile, the present method took less grid points than those of the Newton-MG method for a certain accuracy of approximation.

Compared to the computational expenses of the Newton-MG method, one can find that the number of iterations of ECNMG method on grids Ω_j ($j \geq 3$) are reduced obviously, which is particularly true when the finest grid size h_8 satisfy $h_8^{-1} = 2048$. Hence, the total computational cost of ECNMG method are evidently lower than that of the Newton-MG method. The reason is that we took the extrapolation strategy and the bi-quadratic interpolation to provide a better initial solution U_j^0 with the fifth-order precision, which is one higher than the order of convergence of the fourth-order difference approximations U_j^*. As a consequence, the relative effect of how initial guess U_j^0 approximates difference solution U_j^* becomes better when mesh is refined (see Figure 5), and the number of iterations is reduced most significantly on the finest grid (see Tables 6 to 10).

Finally, as we can see that the ECNMG algorithm is also effective for Example 4.2 with different fourth-order compact difference schemes (HOC4a, HOC4b, HOC4c, HOC4d and HOC4e).

5 CONCLUSIONS

In this work, we extend the idea of extrapolation multigrid computations to the nonlinear problems. We present an extrapolation cascadic Newton multigrid (ECNMG) method combined with some nine-point fourth-order compact difference schemes to solve the two dimensional (2D) Poisson equations with non-linear forcing term. We employ the extrapolation interpolation and the bi-quartic interpolation for two approximation of nonlinear systems on the two-level grids (current and previous grids) to construct a better initial guess on the next finer grid, which greatly reduces the iteration numbers of the Newton-MG method for computing the converged fourth-order accurate solution. Then, we take a completed extrapolation technique to generate a sixth-order accurate extrapolation solution on the entire fine grid from two fourth-order accurate approximations on two different scale grids. Numerical experiments show that the proposed method successfully enhance the accuracy of numerical solutions and keep less costs simultaneously. Therefore, it is much more efficient comparing to the classical Newton multigrid and particularly suitable for solving large scale nonlinear systems.

ACKNOWLEDGMENTS

Ming Li was supported by the National Natural Science Foundation of China (51974377,12161033), the Major Science and Technology Project of Precious Metal Materials Genetic Engineering in Yunnan Province (2019ZE001-1, 202002AB080001), the National Key Research, Development Program of China (2017YFB0305601, 2017YFB0701700), the Natural Science Foundation of Yunnan Province of China (2017FH001-012), the Scientific Research Foundation of Yunnan Education Department (2021J0543), and the Reserve Talents Foundation of Honghe University (2015HB0304).

CONFLICT OF INTEREST

The author declare no potential conflict of interests.

ORCID

Ming Li, http://orcid.org/0000-0003-4183-2404

References

[1] Blum H, Lin Q, Rannacher R. Asymptotic error expansion and Richardson extrapolation for linear finite elements. *Numer Math.* 1986; 49: 11-37.

[2] Briggs WL, Henson VE, McCormick SF. *A multigrid tutorial* (second edition). SIAM. Philadelphia, Pennsylvania, PA. 2000.

[3] Chen CM, Lin Q. Extrapolation of finite element approximation in a rectangular domain. *J Comput Math.* 1989; 7(3): 227-233.

[4] Chen CM, Hu HL, Xie ZQ, et al. Analysis of extrapolation cascadic multigrid method (EXCMG). *Sci. China Ser A-Math*. 2008; 51: 1349-1360.

[5] Chen CM, Shi ZC, Hu HL. On extrapolation cascadic multigrid method. *J Comput Math.* 2011; 29(6): 684-697.

[6] Dai RX, Lin PP, Zhang J. An efficient sixth-order solution for anisotropic Poisson equation with completed Richardson extrapolation and multiscale multigrid method. *Comput Math Appl.* 2017; 73(8): 1865-1877.

[7] Dai RX, Lin PP, Zhang J. An EXCMG accelerated multiscale multigrid computaion for 3D Poisson equation. *Comput Math Appl.* 2019; *77*: 2051-2060.

[8] Dai RX, Wang Y, Zhang J. Fast and high accuracy multiscale multigrid method with multiple coarse grid updating strategy for the 3D convection-diffusion equation. *Comput Math Appl.* 2013; *66*: 542-559.

[9] Dai RX, Zhang J, Wang Y. Multiple coarse grid acceleration for multiscale multigrid computation. *J Comput Appl Math.* 2014; *269*: 75-85.

[10] Dai RX, Zhang J, Wang Y. Higher order ADI method with completed Richardson extrapolation for solving unsteady convection-diffusion equations. *Comput Math Appl.* 2016; *71*: 431-442.

[11] Fairweather G., Lin Q., Lin YP, et al. Asymptotic expansions and Richardson extrapolation of approximate solutions for second order elliptic problems on rectangular domains by mixed finite element methods. *SIAM J Numer Anal.* 2006; 44(3): 1122-1149.

[12] Hu HL, Chen CM, Pan KJ. Asymptotic expansions of finite element solutions to Robin problems in H^3 and their application in extrapolation cascadic multigrid method. *Sci. China Math.* 2014; 57: 687-698.

[13] Hu HL, Chen CM, Pan KJ. Time-extrapolation algorithm (TEA) for linear parabolic problems. *J Comput Math.* 2014; 32: 183-194.

[14] Hu HL, Ren ZY, He DD, et al. On the convergence of an extrapolation cascadic multigrid method for elliptic problems. *Comput Math Appl.* 2017; 74: 759-771.

[15] Li M, Li CL, Cui XZ, et al. Cascadic multigrid methods combined with sixth-order compact scheme for Poisson equation. *Numer Algorithms.* 2016; 71(4): 715-727.

[16] Li M, Li CL. New cascadic multigrid methods for two-dimensional Poisson problem based on the fourth-order compact difference scheme. *Math Meth. Appl. Sci.* 2018; 41(3): 920-928.

[17] Li M, Zheng ZS, Pan KJ. Extrapolation multiscale multigrid method for solving 2D Poisson equation with sixth order compact scheme. *J Appl Math Comput.* 2019; 60: 589ĺC604.

[18] Li M, Zheng ZS, Pan KJ. An efficient extrapolation full multigrid method for elliptic problems in two and three dimensions. *Int J Comput Math.* 2020; 98(6): 1183-1198.

[19] Li M, Zheng ZS, Pan KJ, et al. An efficient Newton multiscale multigrid method for 2D semilinear Poisson equations. *East Asian J Appl Math.* 2020; 10: 620-634.

[20] Lin Q, Lu T, Shen SM. Maximum norm estimate, extrapolation and optimal point of stresses for finite elelment methods on strogle regular triangulation. *J Comput Math.* 1983; 1(4): 376-383.

[21] Manohar R, Stephenson JW. High order difference schemes for linear partial differential equations. *SIAM J Sci Stat Comput.* 1984; 5(1): 69-77.

[22] Marchuk .I, Shaidurov VV. *Difference methods and their extrapolations*. Springer-Verlag, New York (1983)

[23] Pan KJ, He DD, Hu HL. An extrapolation cascadic multigrid method combined with a fourth-order compact scheme for 3D Poisson equation. *J Sci Comput.* 2017; 70: 1180-1203.

[24] Pan KJ, He DD, Chen CM. An extrapolation cascadic multigrid method for elliptic problems on reentrant domains. *Adv Appl Math Mech.* 2016; 9(6): 1347-1363.

[25] Pan KJ, He DD, et al. A new extrapolation cascadic multigrid method for three dimensional elliptic boundary value problems. J Comput Phys. 2017; 344: 499-515.

[26] Pan KJ, Sun HW, Xu Y, et al. An efficient multigrid solver for two-dimensional spatial fractional diffusion equations with variable coefficients. *Appl. Math Comput.* 2021; 402: 126091.

[27] Richardson LF. The approximate solution of physical problems involving differential equations using finite differences, with an application to the stress in a masonry dam, Philos. Trans. Roy. Soc., London, Ser. A. 1910; 210: 307-357.

[28] Richardson LF. The deferred approach to the limit. I: The Single lattice. Philos , Trans. Roy. Soc., London, Ser. A. 1927,; 226: 299-349.

[29] Roache PJ, Knupp PM. Completed Richardson extrapolation. *Communications in Numerical Methods in Engineering.* 1993; 9: 365-374.

[30] Sun HW, Zhang J. A high order finite difference discretization strategy based on extrapolation for convection diffusion equations. *Numer Meth Part D E.* 2004; 20(1): 18-32.

[31] Tian ZF, Dai SQ. High-order compact exponential finite difference methods for convection-diffusion type problems. *J Comput Phys.* 2007; 220: 952-974.

[32] Wang Y, Zhang J. Sixth-order compact scheme combined with multigrid method and extrapolation technique for 2D poisson equation. *J Comput Phys.* 2009; 228(1): 137-146.

[33] Wang Y, Zhang J. Integrated fast and high-accuracy computation of convection diffusion equations using multiscale multigrid method. *Numer Meth Part D E.* 2011; 27: 399-414.

[34] Zhai SY, Feng XL, He YN. A new method to deduce high-order compact difference schemes for two-dimensional Poisson equation. *Appl Math Comput.* 2014; 230: 9-26.

[35] Zhang J. Multigrid method and fourth-order compact difference scheme for 2D Poisson equation with unequal meshsize discretization. *J Comput Phys.* 2002; 179, 170-179.