

# Sixth Order Compact Scheme Combined with Multigrid Method and Extrapolation Technique for 2D Poisson Equation \*

Yin Wang<sup>1†</sup> and Jun Zhang<sup>2‡</sup>

## Abstract

We develop a sixth order finite difference discretization strategy to solve the two dimensional Poisson equation, which is based on the fourth order compact discretization, multigrid method, Richardson extrapolation technique, and an operator based interpolation scheme. We use multigrid V-Cycle procedure to build our multiscale multigrid algorithm, which is similar to the full multigrid method (FMG). The multigrid computation yields the fourth order accurate solution on both the fine grid and the coarse grid. A sixth order accurate coarse grid solution is computed by using the Richardson extrapolation technique. Then we apply our operator based interpolation scheme to get the sixth order accurate solution on the fine grid. Numerical experiments are conducted to show the accuracy and the efficiency of our new method, compared to Sun-Zhang's sixth order Richardson extrapolation compact (REC) discretization strategy using Alternating Direction Implicit (ADI) method and the standard fourth order compact difference (FOC) scheme using a multigrid method.

**Keywords:** Poisson equation, compact difference scheme, multigrid method, Richardson extrapolation.

**Mathematics Subject Classification:** 65N06, 65N55, 65F10.

---

\*Technical Report CMIDA-HiPSCCS 003-08, Department of Computer Science, University of Kentucky, Lexington, KY, 2008.

<sup>†1</sup> Laboratory for High Performance Scientific Computing and Computer Simulation, Department of Computer Science, University of Kentucky, Lexington, KY 40506-0046, USA. E-mail: ywangf@csr.uky.edu, URL: <http://www.csr.uky.edu/~ywangf>.

<sup>‡2</sup> Laboratory for High Performance Scientific Computing and Computer Simulation, Department of Computer Science, University of Kentucky, Lexington, KY 40506-0046, USA. E-mail: jzhang@cs.uky.edu, URL: <http://www.cs.uky.edu/~jzhang>. This author's research work was supported in part by NSF under grants CCF-0727600 and CCF-0527967, in part by the Kentucky Science and Engineering Foundation under grant KSEF-148-502-06-186, in part by the Alzheimer's Association under grant NIGR-06-25460, and in part by NIH under grant 1 R01 HL086644-01.

# 1 Introduction

Poisson equation is a partial differential equation with broad applications in mechanical engineering, theoretical physics and other fields. The 2D Poisson equation can be written in the form of

$$u_{xx}(x, y) + u_{yy}(x, y) = f(x, y), \quad (x, y) \in \Omega, \quad (1)$$

where  $\Omega$  is a rectangular domain, or a union of rectangular domains, with suitable boundary conditions defined on  $\partial\Omega$ . The solution  $u(x, y)$  and the forcing function  $f(x, y)$  are assumed to be sufficiently smooth and have the necessary continuous partial derivatives up to certain orders.

The second order accurate solution can be computed by applying the standard second order central difference operator to  $u_{xx}(x, y)$  and  $u_{yy}(x, y)$  in Eq. (1). Higher order (more than two) accurate discretization methods need more complex procedure than the second order accurate discretization method to compute the coefficient matrix, but they usually generate linear systems of much smaller size, compared with that from the lower order accurate discretization methods [1, 7, 10]. There has been growing interest in developing higher order accurate discretization methods, especially the high order compact difference schemes, to solve partial differential equations (PDEs) [11, 15, 18, 23, 25, 26]. We call them “compact” because these schemes only use the minimum three grid points in one dimension in the discretization formulas.

Previously, Chu and Fan [5, 6] proposed a three point combined compact difference (CCD) scheme for solving two dimensional Stommel Ocean model, which is a special two dimensional convection diffusion equation. They use Hermitian polynomial approximation to achieve sixth order accuracy for the inner grid points and fifth order accuracy for the boundary grid points. The advantage of the CCD scheme is that it can be used to solve many types of PDEs without major modifications. And the Alternating Direction Implicit (ADI) [14] method can be used to reduce the higher dimensional problems to a series of lower dimensional problems. So, their scheme is referred to as implicit high order compact scheme because they do not compute the solution of the variables directly.

In contrary, the explicit fourth order compact schemes [9, 10, 12, 13, 18], compute the solution of the variables directly. Some accelerating iterative methods like multigrid method and preconditioned iterative method have been used to efficiently solve the resulting sparse linear systems [22, 24, 25]. But the higher order explicit compact schemes are more complicated to develop in higher dimensions [8, 27], compared with the implicit compact schemes. As far as we know, there is no existing explicit compact scheme on a single scale grid that is higher than the fourth order accuracy.

Since a sixth order explicit scheme may be impossible to develop on a single scale grid, the multiscale grid method has been considered to achieve the sixth order accuracy for the explicit compact formulations. Sun and Zhang [20] proposed a sixth order explicit finite

difference discretization strategy for solving 2D convection diffusion equation. They used ADI method to compute the fourth order accurate solution on the fine and the coarse grids first, then apply the Richardson extrapolation technique and an operator based interpolation scheme in each ADI iteration to achieve the sixth order accurate solution on the fine grid. The major disadvantage of Sun-Zhang's method is that the ADI iteration is not scalable with respect to the meshsize, when the mesh becomes finer, the number of iterations increases quickly.

By using the idea of two scale grid computation from Sun-Zhang's method, we intend to develop a new explicit sixth order compact scheme computing strategy for the 2D Poisson equation, which can efficiently solve the resulting linear system and is scalable with respect to the problem size. We do not use the ADI method, instead we develop a multigrid method that is similar to the full multigrid method as our convergence acceleration method. With point Gauss-Seidel relaxation method and line Gauss-Seidel relaxation method, we iteratively solve the resulting sparse linear system to get the fourth order accurate solution on both the fine and the coarse grids. Then we apply the Richardson extrapolation technique combined with our new operator based interpolation scheme to get the sixth order accurate solution on the fine grid.

In this paper, we present the sixth order compact difference discretization strategy for 2D Poisson equation in Section 2. In Section 3, we develop our modified multigrid method to solve the fourth order accurate solution on the fine and the coarse grids. Section 4 contains the numerical experiments to demonstrate high accuracy of the sixth order compact difference scheme, as well as efficiency of our modified multigrid method. Concluding remarks are given in Section 5.

## 2 Sixth Order Compact Approximations

Our explicit sixth order compact difference scheme is based on the fourth order compact discretization on the two scale grids. In this section, we first introduce the fourth order compact difference scheme for the 2D Poisson equation. The basic idea is from Zhang's previous papers [20, 25, 28]. More detailed discussions about the fourth order compact difference schemes can be found in [9, 17].

In order to discretize the Eq. (1), let us consider a rectangular domain  $\Omega = [0, L_x] \times [0, L_y]$ . We discretize  $\Omega$  with uniform meshsizes  $\Delta x = L_x/N_x$  and  $\Delta y = L_y/N_y$  in the  $x$  and  $y$  coordinate directions, respectively. Here  $N_x$  and  $N_y$  are the number of uniform intervals in the  $x$  and  $y$  coordinate directions. The mesh points are  $(x_i, y_j)$  with  $x_i = i\Delta x$  and  $y_j = j\Delta y$ ,  $0 \leq i \leq N_x$ ,  $0 \leq j \leq N_y$ .

We write the standard second order central difference operators as

$$\delta_x^2 u_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}, \quad \delta_y^2 u_{i,j} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}.$$

Using Taylor series expansions, at the grid point  $(x_i, y_j)$ , we have

$$\delta_x^2 u_{i,j} = u_{xx} + \frac{\Delta x^2}{12} u_x^4 + \frac{\Delta x^4}{360} u_x^6 + O(\Delta x^6), \quad (2)$$

and

$$\delta_y^2 u_{i,j} = u_{yy} + \frac{\Delta y^2}{12} u_y^4 + \frac{\Delta y^4}{360} u_y^6 + O(\Delta y^6), \quad (3)$$

From previous studies [20, 25, 28], we know that with Eqs. (2) and (3), we can apply the symbolic fourth order compact approximation operator to the second derivatives  $u_{xx}$  and  $u_{yy}$  in Eq. (1), respectively. The discrete 2D Poisson equation will be formulated symbolically as [19]

$$\left(1 + \frac{\Delta x^2}{12} \delta_x^2\right)^{-1} \delta_x^2 u + \left(1 + \frac{\Delta y^2}{12} \delta_y^2\right)^{-1} \delta_y^2 u = f + \tau_1 \Delta x^4 + \tau_2 \Delta y^4 + O(\Delta^6), \quad (4)$$

where  $\tau_1$  and  $\tau_2$  are used to denote some complex representations that will be canceled in the Richardson extrapolation procedure,  $\Delta^6$  denotes the truncated terms in the order of  $O(\Delta x^6 + \Delta y^6)$ . By applying the symbolic operators, setting  $\tau_1$  and  $\tau_2$  both equal to zero and dropping the  $\Delta^6$  terms, the Eq. (4) can be rewritten as

$$\begin{aligned} \left(1 + \frac{\Delta y^2}{12} \delta_y^2\right) \delta_x^2 u + \left(1 + \frac{\Delta x^2}{12} \delta_x^2\right) \delta_y^2 u &= \left(1 + \frac{\Delta x^2}{12} \delta_x^2\right) \left(1 + \frac{\Delta y^2}{12} \delta_y^2\right) f + O(\Delta^4) \\ &= \left[1 + \frac{1}{12} (\Delta x^2 \delta_x^2 + \Delta y^2 \delta_y^2)\right] f + O(\Delta^4). \end{aligned} \quad (5)$$

If we set the mesh aspect ratio  $\gamma = \Delta x / \Delta y$ , we can rewrite the Eq. (5) into the following form [25]

$$\begin{aligned} a u_{i,j} + b(u_{i+1,j} + u_{i-1,j}) + c(u_{i,j+1} + u_{i,j-1}) + d(u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j+1} + u_{i-1,j-1}) \\ = \frac{\Delta x^2}{2} (8f_{i,j} + f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1}), \end{aligned} \quad (6)$$

which has a nine point computational stencil. Here the coefficients are

$$a = -10(1 + \gamma^2), \quad b = 5 - \gamma^2, \quad c = 5\gamma^2 - 1, \quad d = (1 + \gamma^2)/2.$$

**Extrapolation and operator interpolation.** With Eq. (6) we can first use the multigrid method to compute the fourth order accurate solutions  $u_{i,j}^{2h}$  and  $u_{i,j}^h$  on the  $\Omega_{2h}$  grid and  $\Omega_h$  grid, respectively. Then we apply the Richardson extrapolation technique, to compute a sixth order accurate solution  $\tilde{u}_{i,j}^{2h}$  on  $\Omega_{2h}$ .

The general Richardson extrapolation can be written as [4]

$$\tilde{u}_{i,j}^{2h} = \frac{(2^p u_{2i,2j}^h - u_{i,j}^{2h})}{2^{p-1}}, \quad (7)$$

where  $p$  is the order of accuracy before the extrapolation, and the order of accuracy will be increased to  $p + 2$  after the extrapolation. In this paper, the order  $p$  is equal to four, so the Richardson extrapolation formula we use is

$$\tilde{u}_{i,j}^{2h} = \frac{(16u_{2i,2j}^h - u_{i,j}^{2h})}{15}. \quad (8)$$

By using Eq. (8) we can compute the sixth order accurate solution  $\tilde{u}_{2h}$  on the coarse grid  $\Omega_{2h}$ . Then we need some interpolation technique, as shown in Fig. 1, to interpolate the sixth order accurate solution from the coarse grid  $\Omega_{2h}$  to the fine grid  $\Omega_h$ . The *(even, even)* indexed grid points on  $\Omega_h$  can be directly interpolated as  $\tilde{u}_{2i,2j}^h = \tilde{u}_{i,j}^{2h}$  and it keeps the sixth order accuracy. For other grid points, we need an operator based interpolation scheme to achieve the sixth order accuracy.

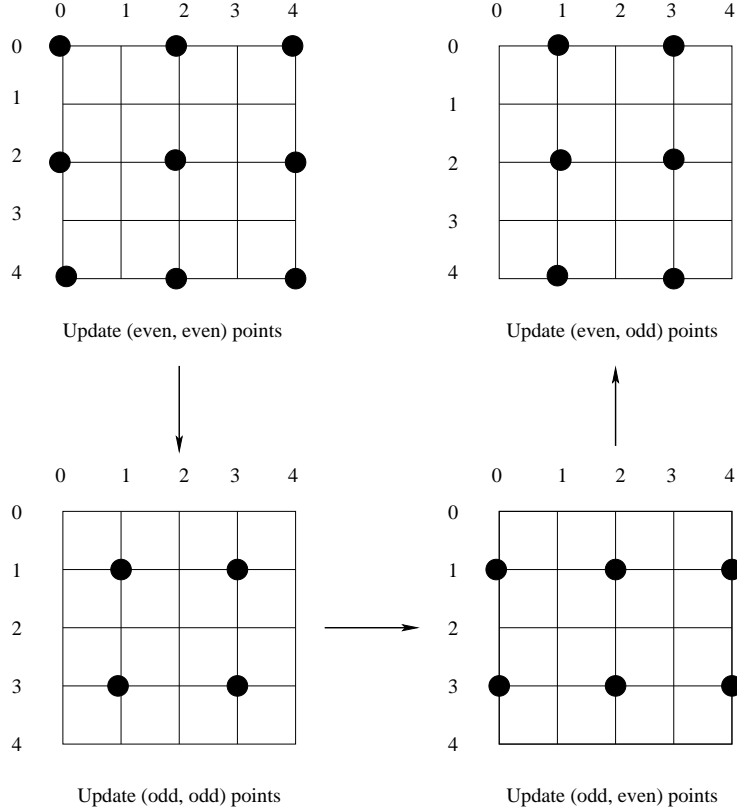


Figure 1: Illustration of the operator based interpolation scheme for a  $5 \times 5$  fine grid.

In Sun-Zhang's sixth order method [20] for the 2D convection diffusion equation, they also apply an operator based interpolation scheme together with the Richardson extrapolation in each ADI iteration. Since the number of ADI iteration will increase quickly when the meshsize becomes finer, their extrapolation and interpolation parts will take a large amount of CPU cost. In order to avoid this, our new operator based interpolation procedure and the Richardson extrapolation are carried out only after we get the converged fine and coarse grid fourth order accurate solutions.

We assume that  $N_x$  and  $N_y$  are both even numbers, our operator based interpolation scheme is an iterative procedure. In each iteration, it will run the Richardson extrapolation first to get the sixth order solution on the coarse grid, then it will use a different interpolation strategy to interpolate the sixth order solution for different grid points on the fine grid. One

interpolation iteration (from step  $k$  to  $k + 1$ ) is outlined in Algorithm 1.

---

**Algorithm 1** Operator based interpolation iteration combined with the sixth order Richardson extrapolation technique.

---

- 1: Let  $u_{old}^h = \tilde{u}^{h,k}$ .
- 2: Update every (*even, even*) grid point on  $\Omega_h$ .  
From  $\tilde{u}_{i,j}^{2h,k} \in \Omega_{2h}^4$  and  $\tilde{u}_{2i,2j}^{h,k} \in \Omega_h^4$ , we first compute  $\tilde{u}_{i,j}^{2h,k+1} \in \Omega_{2h}^6$  by Eq. (8), then use direct interpolation to get  $\tilde{u}_{2i,2j}^{h,k+1} \in \Omega_h^6$ .
- 3: Update every (*odd, odd*) grid point on  $\Omega_h$ .  
From Eq. (6), for each (*odd, odd*) point  $(i, j)$ , the updated solution is

$$\begin{aligned} \tilde{u}_{i,j}^{h,k+1} = & \frac{1}{a}[F_{i,j} - b(\tilde{u}_{i+1,j}^{h,k} + \tilde{u}_{i-1,j}^{h,k}) - c(\tilde{u}_{i,j+1}^{h,k} + \tilde{u}_{i,j-1}^{h,k}) \\ & - d(\tilde{u}_{i+1,j+1}^{h,k+1} + \tilde{u}_{i+1,j-1}^{h,k+1} + \tilde{u}_{i-1,j+1}^{h,k+1} + \tilde{u}_{i-1,j-1}^{h,k+1})] \end{aligned}$$

Here,  $F_{i,j}$  represents the right-hand side part of Eq. (6).

- 4: Update every (*odd, even*) grid point on  $\Omega_h$ .  
From Eq. (6), the idea is similar to the (*odd, odd*) grid point.
  - 5: Update every (*even, odd*) grid point on  $\Omega_h$ .  
From Eq. (6), the idea is similar to the (*odd, even*) grid point.
  - 6: Compute the 2-norm  $R = \|\tilde{u}^{h,k+1} - u_{old}^h\|_2$ . If not converged, go back to Step 1.
- 

In Algorithm 1,  $\Omega_h^4$  and  $\Omega_{2h}^4$  denote the fourth order accurate solution space,  $\Omega_h^6$  and  $\Omega_{2h}^6$  mean the sixth order accurate solution space.  $\tilde{u}^{h,k}$  is the approximate solution for the fine grid after  $k$  iterations. The operator based interpolation iteration will continue until the 2-norm  $R$  of the correction vector is reduced to below a certain tolerance.

### 3 Modified Multiscale Multigrid Method

The convergence rate of the multigrid method is independent of the grid size [2, 3, 21]. It is a very efficient method to solve large sparse linear systems arising from PDEs. Various multigrid implementation strategies with the fourth order compact schemes to solve the 2D and 3D Poisson equations or other PDEs like convection diffusion equations are discussed in [7, 10, 16]. In this paper, we use a geometric multiscale multigrid method [3, 16], similar to the full multigrid method, to compute the fourth order accurate solution on both the fine and the coarse grids.

We use the notations  $u_{lh}$ ,  $f_{lh}$  and  $L_{lh}$  to represent the approximate solution, right-hand side vector and the finite difference operator for the grid  $\Omega_{lh}$ , respectively.  $I_{(l-1)h}^{lh}$  is the restriction operator from the grid  $\Omega_{(l-1)h}$  to the grid  $\Omega_{lh}$  and  $I_{lh}^{(l-1)h}$  is the interpolation operator from the grid  $\Omega_{lh}$  to the grid  $\Omega_{(l-1)h}$ . The procedure of our multiscale multigrid method is shown in Fig. 2. The gray color circle indicates the unconverged solution  $u_{4h}$  and the black color circles are the fourth order converged solutions  $u_{2h}$  and  $u_h$ .

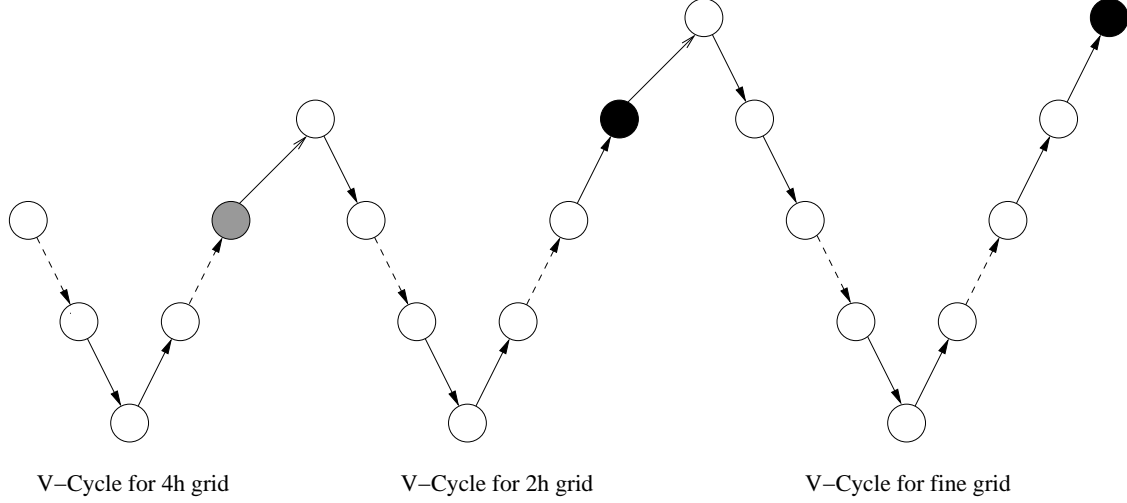


Figure 2: Representation of our multiscale multigrid method.

Below we describe a multigrid V-cycle based algorithm to solve the 2D Poisson equation.

---

**Algorithm 2** Multiscale Multigrid Method

---

- 1: Run the multigrid V-Cycle algorithm  $MG(u_{4h}, f_{4h})$  on the coarser grid  $\Omega_{4h}$  as in Fig. 2 for one or two cycles to get an approximate solution  $u_{4h}$ .
  - 2: Use some high order interpolation schemes, like the bicubic interpolation or operator based interpolation, to interpolate  $u_{4h}$  to the coarse grid  $\Omega_{2h}$ ,  $u_{2h} = I_{4h}^{2h} u_{4h}$ .
  - 3: Relax  $\nu_1$  times on  $L_{2h} u_{2h} = f_{2h}$ .
  - 4: Use  $u_{2h}$  from the previous step as the initial guess to run the multigrid V-Cycle algorithm  $MG(u_{2h}, f_{2h})$  on the coarse grid  $\Omega_{2h}$  until it converges. We can get the converged fourth order accurate solution  $u_{2h}$ .
  - 5: Use a high order interpolation to interpolate  $u_{2h}$  to the fine grid  $\Omega_h$  like  $u_h = I_{2h}^h u_{2h}$ .
  - 6: Relax  $\nu_1$  times on  $L_h u_h = f_h$ .
  - 7: Use  $u_h$  from the previous step as the initial guess to run the multigrid V-Cycle algorithm  $MG(u_h, f_h)$  on the fine grid  $\Omega_h$  until it converges. We can get the converged fourth order accurate solution  $u_h$ .
- 

The Algorithm 2 is similar to the full multigrid method, but we do not start from the coarsest grid. Since we use the interpolated coarse grid solution as the initial guess for the fine grid V-Cycle, this algorithm will need fewer number of multigrid cycles than we run the V-Cycle on  $\Omega_h$  and  $\Omega_{2h}$  separately to get the converged fourth order accurate solutions  $u_h$  and  $u_{2h}$  [3, 16].

For relaxation methods (smoothers) in this paper, we use point Gauss-Seidel relaxation and line Gauss-Seidel relaxation. The standard multigrid method with point Gauss-Seidel

relaxation does not work well for the anisotropic Poisson equation [21, 25], instead we can use line Gauss-Seidel relaxation method. The line relaxation scheme has three strategies [28], X-Line scheme, Y-Line scheme and X-Y Line scheme. In our multiscale multigrid method with both relaxation methods, we use standard bilinear interpolation to transfer corrections from the coarse grid to the fine grid, full weighting scheme to project residual from the fine grid to the coarse grid, and bicubic interpolation to interpolate the initial guess in Step 2 and Step 5.

## 4 Numerical Results

In this section, we compare our new sixth order multigrid method with Richardson extrapolation (MG-Six) strategy with Sun-Zhang's sixth order REC method (REC-ADI) [20] and with the standard fourth order compact difference scheme using multigrid (MG-FOC). The codes are written in Fortran 77 programming language and run on one processor of an IBM HS21 blade cluster at the University of Kentucky. The processor has 2GB of local memory and runs at 2.0GHZ.

The initial guess for the V-Cycle on  $\Omega_{4h}$  is the zero vector. For the Problem 1, the multigrid V-Cycle for the  $\Omega_{2h}$  and  $\Omega_h$  grids will stop when the 2-norm of the residual vector is reduced by  $10^{-13}$ , the iterative interpolation procedure will stop when the 2-norm of the correction vector of the approximate solution is less than  $10^{-13}$ . For the Problem 2, both of the stopping criteria will be changed to  $10^{-10}$ . The errors reported are the maximum absolute errors over the discrete grid of the finest level. For the relaxation methods in our test cases, we use point Gauss-Seidel relaxation and X-Y Line Gauss-Seidel relaxation for the Problem 1, and use point Gauss-Seidel relaxation and X-Line Gauss-Seidel relaxation for the Problem 2.

**Problem 1.** In order to compare with Sun-Zhang's sixth order method, we consider one of the test cases in Sun-Zhang's paper [20]. Sun and Zhang used a 2D convection diffusion equation. We set the convection coefficient as zero, then the equation is equal to a 2D Poisson equation. The test case is as follows:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -\alpha \sin\left(\frac{\pi}{b}y\right), \quad (x, y) \in \Omega = [0, \lambda] \times [0, b], \quad (9)$$

where the boundary conditions are

$$u(0, y) = u(\lambda, y) = u(x, 0) = u(x, b) = 0.$$

In this equation, the parameter  $\alpha$  is chosen as

$$\alpha = \frac{F\pi}{Rb}.$$

The analytic solution of Eq. (9) is:

$$u = -\alpha \left(\frac{b}{\pi}\right)^2 \sin\left(\frac{\pi y}{b}\right) \left(e^{\frac{\pi x}{b}} - 1\right).$$

The other parameters are chosen as

$$\lambda = 10^7 m, b = 2\pi \times 10^6 m, F = 0.3 \times 10^{-7} m^2 s^{-2}, R = 0.6 \times 10^{-3} m s^{-1}.$$

In the following, we define  $N_x = N_y = n$ . The mesh sizes  $\Delta_x$  and  $\Delta_y$  are equal to  $\lambda/n$  and  $b/n$ , respectively. Table 1 and Fig. 3 show the comparison results for the Problem 1.

Table 1: Numerical comparison results for the Problem 1.

$n$	strategy	# iteration	CPU	error	order
16	REC-ADI	237	0.020	1.32e-6	5.7
	MG-Six(point)	(11,11), <b>40</b>	0.001	1.32e-6	5.7
	MG-FOC(point)	14	0.002	1.63e-5	3.9
	MG-Six(line)	(5,6), <b>40</b>	0.001	1.32e-6	5.7
	MG-FOC(line)	7	0.001	1.63e-5	3.9
32	REC-ADI	901	0.302	2.27e-8	5.9
	MG-Six(point)	(11,12), <b>39</b>	0.007	2.27e-8	5.9
	MG-FOC(point)	14	0.004	1.02e-6	4.0
	MG-Six(line)	(6,7), <b>39</b>	0.008	2.27e-8	5.9
	MG-FOC(line)	8	0.006	1.02e-6	4.0
64	REC-ADI	3447	4.662	3.68e-10	5.9
	MG-Six(point)	(12,13), <b>37</b>	0.029	3.67e-10	6.0
	MG-FOC(point)	15	0.022	6.39e-8	4.0
	MG-Six(line)	(7,7), <b>36</b>	0.033	3.67e-10	6.0
	MG-FOC(line)	9	0.024	6.40e-8	4.0
128	REC-ADI	>5000	–	–	–
	MG-Six(point)	(13,13), <b>33</b>	0.129	5.26e-12	6.1
	MG-FOC(point)	15	0.093	3.99e-9	4.0
	MG-Six(line)	(7,8), <b>34</b>	0.161	5.87e-12	6.0
	MG-FOC(line)	9	0.140	4.00e-9	4.0
256	REC-ADI	>5000	–	–	–
	MG-Six(point)	(14,14), <b>30</b>	0.880	1.27e-13	5.4
	MG-FOC(point)	16	0.465	2.47e-10	4.0
	MG-Six(line)	(8,8), <b>30</b>	1.202	1.11e-13	5.7
	MG-FOC(line)	8	0.737	2.50e-10	4.0

Table 1 shows the number of iterations and other information for different solution strategies that we compared. We can find that when the mesh becomes finer, the number of ADI iterations increases very quickly. When  $n > 64$ , the number of ADI iterations is greater than 5000, which means it could not converge within the maximum number of iterations we set. For the MG-Six method, the number of iterations contains three parts. They are the number of V-Cycles for  $\Omega_{2h}$ , the number of V-Cycles for  $\Omega_h$ , and the number of iterations for the iterative interpolation combined with Richardson extrapolation. These three numbers are listed in the iteration columns for MG-Six(point) and MG-Six(line) strategies in Table 1. We can see that, by using our new sixth order compact scheme, the number of V-Cycles for  $\Omega_h$  and  $\Omega_{2h}$  are reduced compared to the traditional multigrid V-Cycle with the FOC scheme. We can also see that the REC-ADI method takes much more iterations than the MG-FOC strategies from Fig. 3, and the multigrid method combined with the line relaxation takes less iterations, but more CPU time, than the point relaxation.

The data in Table 1 also indicates that the accuracy of the solution computed by our new sixth order method and the Sun-Zhang's method is comparable. When  $n > 64$ , our multigrid method can still compute the high accuracy solution. The CPU time for our method is much less than that needed by the ADI iteration, and is better than running the MG-FOC separately twice to get  $u_h$  and  $u_{2h}$ .

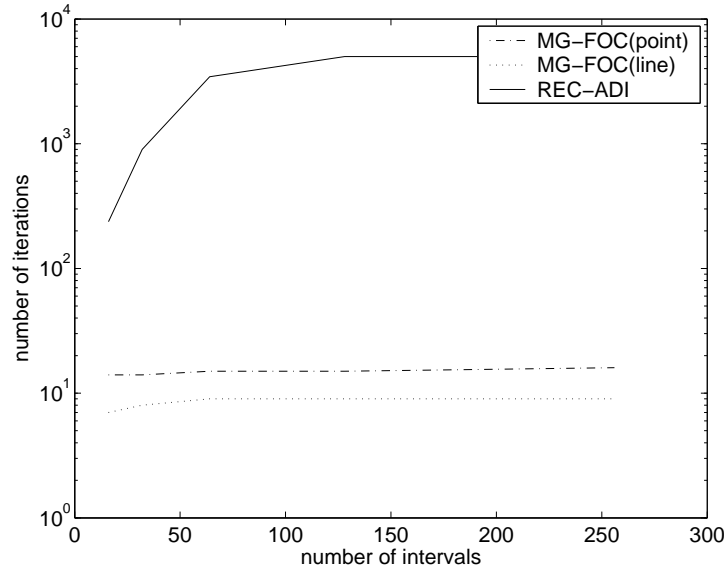


Figure 3: Comparison of the number of iterations with REC-ADI, MG-FOC(point) and MG-FOC(line) methods for the Problem 1.

**Problem 2.** We consider another Poisson equation in the form of

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -104\pi^2 \sin(10\pi x) \cos(2\pi y), \quad (x, y) \in \Omega = [0, 1] \times [0, 1], \quad (10)$$

which has a zero boundary condition.

The analytic solution of Eq. (10) is:

$$u(x, y) = \sin(10\pi x) \cos(2\pi y).$$

Table 2: Numerical comparison results for the Problem 2.

$n$	strategy	# iteration	CPU	error	order
16	REC-ADI	37	0.003	6.29e-2	–
	MG-Six(point)	(14,13), <b>20</b>	0.001	6.29e-2	–
	MG-FOC(point)	9	0.002	6.33e-2	–
	MG-Six(line)	(9,8), <b>20</b>	0.001	6.29e-2	–
	MG-FOC(line)	7	0.001	6.33e-2	–
32	REC-ADI	121	0.041	1.42e-3	5.5
	MG-Six(point)	(13,11), <b>21</b>	0.006	1.42e-3	5.5
	MG-FOC(point)	11	0.004	3.39e-3	4.2
	MG-Six(line)	(8,8), <b>21</b>	0.006	1.42e-3	5.5
	MG-FOC(line)	8	0.003	3.39e-3	4.2
64	REC-ADI	476	0.689	2.95e-5	5.6
	MG-Six(point)	(11,11), <b>29</b>	0.026	2.95e-5	5.6
	MG-FOC(point)	12	0.015	2.01e-4	4.1
	MG-Six(line)	(8,7), <b>29</b>	0.025	2.95e-5	5.6
	MG-FOC(line)	8	0.016	2.01e-4	4.1
128	REC-ADI	1848	10.802	5.10e-7	5.9
	MG-Six(point)	(11,10), <b>32</b>	0.115	5.10e-7	5.9
	MG-FOC(point)	12	0.063	1.24e-5	4.0
	MG-Six(line)	(7,7), <b>32</b>	0.113	5.10e-7	5.9
	MG-FOC(line)	9	0.070	1.24e-5	4.0
256	REC-ADI	>5000	–	–	–
	MG-Six(point)	(10,10), <b>31</b>	0.796	8.21e-9	6.0
	MG-FOC(point)	12	0.355	7.66e-7	4.0
	MG-Six(line)	(8,7), <b>31</b>	1.041	8.21e-9	6.0
	MG-FOC(line)	10	0.670	7.66e-7	4.0

Like for the Problem 1, we also compare the different solution strategies indexed by the number of multigrid cycles or iterations, CPU time, the maximum absolute errors and the

order of accuracy. The results are shown in Table 2 and Fig. 4. For the Problem 2, we use X-Line Gauss-Seidel relaxation instead of X-Y Line relaxation. The results of the order of accuracy in Table 2 do not contain the value for  $n = 16$ . The reason is we cannot get the accurate solution when  $n = 8$ , and we need the maximum error from both grids to compute the order of accuracy for  $n = 16$ . That is also the reason why we need more iterations for the MG-Six(point) and MG-Six(line) strategies when  $n = 16$ .

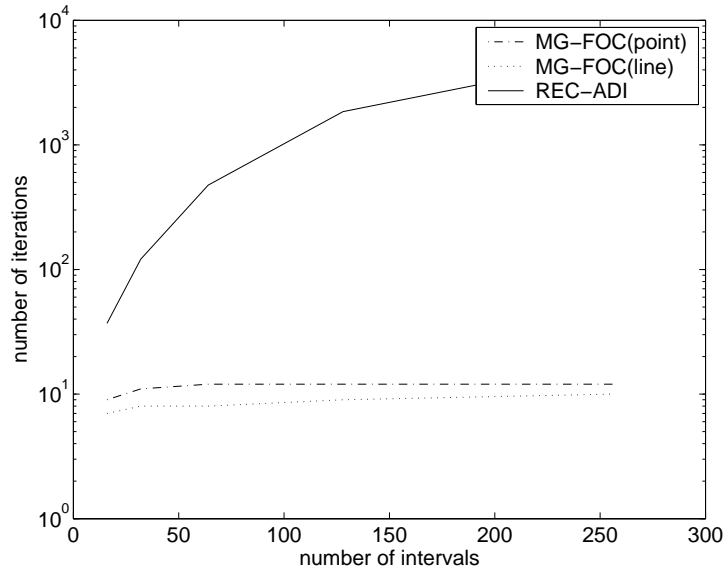


Figure 4: Comparison of the number of iterations with REC-ADI, MG-FOC(point) and MG-FOC(line) methods for the Problem 2.

Once again, we see that the sixth order multigrid method computes accurate solution with a convergence rate that is independent of the problem size. It is clear that MG-Six method outperforms other methods.

## 5 Concluding Remarks

We designed a new sixth order compact computation scheme with a multigrid method and Richardson extrapolation to solve the 2D Poisson equation. This new idea is based on designing a geometric multiscale multigrid method, similar to the full multigrid method, to compute the approximate solution using the fourth order compact scheme in both the fine and the coarse grids. We also present a new iterative interpolation scheme, which is combined with the Richardson extrapolation to achieve the sixth order accuracy on the fine grid.

Numerical results show that the new numerical solution method can solve the 2D Poisson equation with highly accurate solution compared with other sixth order compact schemes,

and also keep the low CPU cost. This idea can also be extended to solve other PDEs such as the 3D Poisson equation, 2D and 3D convection diffusion equations.

## References

- [1] Y. Adam, Highly accurate compact implicit methods and boundary conditions. *J. Comput. Phys.*, 24(1):10-22, 1977.
- [2] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comput.*, 31(138):333-390, 1977.
- [3] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM, Philadelphia, PA, 2nd edition, 2000.
- [4] W. Cheney and D. Kincard. *Numerical Mathematics and Computing*, 4th Ed., Brooks/Cole Publishing, Pacific Grove, CA, 1999.
- [5] P. C. Chu and C. Fan. A three-point combined compact difference scheme. *J. Comput. Phys.*, 140:370-399, 1998.
- [6] P. C. Chu and C. Fan. A three-point six-order nonuniform combined compact difference scheme. *J. Comput. Phys.*, 148:663-674, 1999.
- [7] M. M. Gupta, J. Kouatchou, and J. Zhang. Comparison of second and fourth order discretization for multigrid Poisson solver. *J. Comput. Phys.*, 132:226-232, 1997.
- [8] M. M. Gupta and J. Kouatchou. Symbolic derivation of finite difference approximations for the three dimensional Poisson equation. *Numer. Methods Partial Differential Equations*, 18(5):593-606, 1998.
- [9] M. M. Gupta, R. P. Manohar, and J. W. Stephenson. A single cell high order scheme for the convection-diffusion equation with variable coefficients. *Int. J. Numer. Methods Fluids*, 4:641-651, 1984.
- [10] M. M. Gupta and J. Zhang. High accuracy multigrid solution of the 3D convection-diffusion equation. *Appl. Math. Comput.*, 113(2-3):249-274, 2000.
- [11] S. K. Lele. Compact finite difference schemes with spectral-like resolution. *J. Comput. Phys.*, 103:16-42, 1992.
- [12] M. Li, T. Tang, and B. Fornberg. A compact fourth-order finite difference scheme for the steady incompressible Navier-Stokes equations. *Int. J. Numer. Methods Fluids*, 20:1137-1151, 1995.
- [13] R. J. MacKinnon and R. W. Johnson. Differential-equation-based representation of truncation errors for accurate numerical simulation. *Int. J. Numer. Methods Fluids*, 13:739-757, 1991.

- [14] D. Peaceman and H. Rachford. The numerical solution of elliptic and parabolic differential equations. *Journal. of SIAM*, 3:28-41, 1955.
- [15] K. Sakurai, T. Aoki, W. H. Lee, and K. Kato. Poisson equations solver with fourth-order accuracy by using interpolated differential operator scheme. *Comput. Math. Appl.*, 43:621-630, 2002.
- [16] S. Schaffer. Higher order multigrid method. *Math. Comput.*, 43(167):89-115, 1984.
- [17] W. F. Spotz. *High-Order Compact Finite Difference Schemes for Computational Mechanics*. PhD thesis, University of Texas at Austin, Austin, TX, 1995.
- [18] W. F. Spotz and G. F. Carey. High-order compact scheme for the steady streamfunction vorticity equations. *Int. J. Numer. Methods Engrg.*, 38:3497-3512, 1995.
- [19] J. C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations*, Chapman & Hall, New York, NY, 1989.
- [20] H. Sun and J. Zhang. A high order finite difference discretization strategy based on extrapolation for convection diffusion equations. *Numer. Methods Partial Differential Eq.*, 20(1):18-32, 2004.
- [21] P. Wesseling. *An Introduction to Multigrid Methods*. Wiley, Chichester, England, 1992.
- [22] J. Zhang. *Multigrid Acceleration Techniques and Applications to the Numerical Solution of Partial Differential Equations*. PhD thesis, The George Washington University, Washington, DC, 1997.
- [23] J. Zhang. An explicit fourth-order compact finite difference scheme for three dimensional convection-diffusion equation. *Commun. Numer. Methods Engrg.*, 14:209-218, 1998.
- [24] J. Zhang. Preconditioned iterative methods and finite difference schemes for convection diffusion. *Appl. Math. Comput.*, 109(1):11-30, 2000.
- [25] J. Zhang. Multigrid method and fourth order compact difference scheme for 2D Poisson equation with unequal meshsize discretization. *J. Comput. Phys.*, 179:170-179, 2002.
- [26] J. Zhang, L. Ge, and J. Kouatchou. A two colorable fourth-order compact difference scheme and parallel iterative solution of the 3D convection diffusion equation. *Math. Comput. Simulation*, 54(1-3):65-80, 2000.
- [27] J. Zhang and L. Ge. Symbolic computation of high order compact difference schemes for three dimensional linear elliptic partial differential equations with variable coefficients. *J. Comput. Appl. Math.*, 143(1):9-27, 2002.
- [28] J. Zhang, H. Sun, and J. J. Zhao. High order compact scheme with multigrid local mesh refinement procedure for convection diffusion problems. *Comput. Methods Appl. Mech. Engrg.*, 191(41-42):4661-4674, 2002.