# An Investigation of Multi-Layer Scheme for Grid Refinement Methods

**Leina Wu[1], Tsun-Zee Mai[2]**

[1]Department of Mathematics and Physics, Queens University of Charlotte, Charlotte, North Carolina, USA

[2]Sichuan University Pittsburgh Institute, Sichuan University, Chengdu, Sichuan, China

**Abstract:** ***Grid Refinement Method has been developed to solve linear systems developed from partial differential equations along with iterative methods. A multi-layer scheme in the Grid Refinement Method is developed and presented in the paper. Accuracy and efficiency of the multi-layer scheme is investigated for comparison with the solutions obtained by uniform grid and the two layer scheme approach.***

**Keywords:** Grid Refinement Method, linear system, partial differential equation, iterative methods, stopping procedure, Successive Over Relaxation Method, matrix decomposition, Dirichlet boundary conditions

## 1. Introduction

Groundwater problems [1, 2] have always been around our daily lives. Contamination of the groundwater source can cause serious environmental hazards. Regions with sink holes can be particularly susceptible to the groundwater contamination. Once water enters a sinkhole, it receives litter filtration or chance for degradation of the chemicals. Then the water will migrate down to the groundwater table where it will disperse in the groundwater and migrate in the direction of the groundwater flow.

Grid Refinement methods [8, 9] have been used to solve large linear systems developed from partial differential equations. In this research, we study a problem that arises from the traveling of groundwater flow [1, 2] and the method to estimate how fast the contaminant disperses around the sinkhole in geological sciences. Suppose there is a rectangular domain where the boundary conditions are given and the initial contamination value at the sinkhole is also known, we would like to know the values in the region around the sinkhole. Researchers have been using the model "MODFLOW" to solve the problem by a two-step procedure. The major drawback of this two-step method is the computational time and the inconvenience of the interpolation process due to the two linear systems generated from procedure, as well as considerably more computational time to solve the systems.

Grid Refinement methods improve the above method by generating only one system of linear equations, which contains the information of all points that we are interested in, with the consideration of the treatments of the interface boundary points. In the existing two-step scheme, the approximations for interface boundary points are obtained by the interpolation technique, while the grid refinement method uses a simple "modified" finite difference scheme.

## 2. Grid Refinement Method

In this section, we describe a grid refinement method for solving a partial differential equation of the form:

$$A(x,y)u_{xx} + C(x,y)u_{yy} + D(x,y)u_x + E(x,y)u_y + F(x,y)u = G(x,y) \quad (2.1)$$

where *A,C,D,E,F* are functions of *x* and *y*, with Dirichlet boundary conditions[11] on a rectangular region.

The basic idea of the grid refinement method is to decompose the original spatial domain into several sub-domains. For simplicity, we describe the method using two subdomains, namely interested domain and less interested domain. The coarse grids are put on the less interested domain; therefore it is also referred to as coarse grid domain. And the fine grids are put on the interested domain which is then also referred to as fine grid domain. Once the grids are placed, one linear system is generated. Then we are able to solve both subdomains simultaneously. We note that the fine grid subdomain could be formed in a rectangular shape or in other shapes, such as L shape or circular shape. In this research, we focus on rectangular shape. We also note that the fine grid subdomain could be placed anywhere within the original region to fit physical needs; again for simplicity, we assume in this research that the fine-grid subdomain is located in the center of the region, see Figure 2.1.

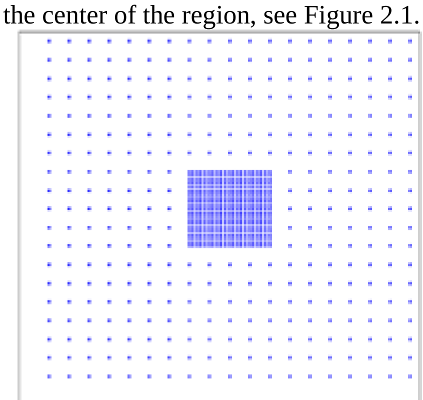

**Figure 2.1**

Once the linear system is generated according to the above grid pattern, we proceed with the solving the linear system in iterative methods [6, 7], such as Richardson's Method [10], Jacobi Method [3] and Gauss-Seidel [4] with Successive Over Relaxation Method [5]. The iterative algorithm produces a sequence of approximations $\{x^{(i)}\}$to the exact solution $\overline{x}$ of the linear system (2.1), it is necessary to have a stopping procedure to determine whether the approximation is accurate enough to terminate the iterative procedure. In this paper, if the exact solution $\bar{x}$ is known, then it would be reasonable to accept the approximate solution $x^{(i)}$ if

$$\frac{\left\|x^{(i)}-\bar{x}\right\|_2}{\left\|\bar{x}\right\|_2} \leq \varepsilon \tag{2.2}$$

where $\varepsilon$ is a preset small tolerance, say $10^{-6}$. We call the test (2.2) an exact stopping test.

## 3. Multi-Layer Scheme in Grid Refinement Method

It would be very interesting to apply the grid refinement idea to a telescoping refined domain, inside which there are one or more refined regions, see Figure 3.1.

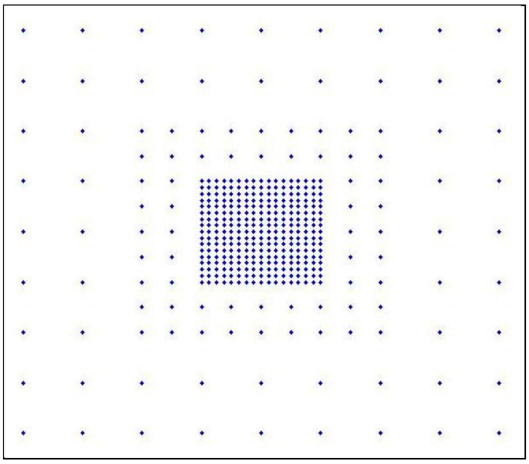

**Figure 3.1:** Grid pattern for the three-layer scheme

We call this scheme a multiple-layer scheme. In this research we extend the grid refinement technique into a three-layer domain. The three-layer domain consists of three different grid sizes, namely, coarse, fine and finest grid size. The regions covered with coarse grids, fine grids and finest grids are referred to as the first layer, the second layer and the third layer, respectively. It is assumed that the third layer is the inner most region and the first layer is the outer most region.

The treatments of the interface between the first and the second layers were described in the previous chapter. The same treatments can also be used on the interface between the second and the third layers. Therefore, this three-layer scheme should be also of first order. Table 4.9 shows the results on the interested region which is the third layer. In this experiment, the third layer with a mesh size h is located at [0.4, 0.6] × [0.4, 0.6]; the second layer with a mesh size h2 is located within [0.3, 0.7]×[0.3, 0.7], and the outside is the third layer covered by coarse grids with mesh size H. When H, h2 and h are halved, the error is reduced by approximately 1/2.

## 4. Numerical Experiment

We now attempt to apply the three-layer scheme of the grid refinement method to the following model problems for investigation of its accuracy and efficiency.

### 4.1 Model Problem 1 (MP1)

$$u_{xx} + u_{yy} = 0$$

over the region Ω= [0, 1] × [0, 1]. The boundary conditions are given by

$$u(0,y) = \cos y,\ u(1,y) = e^{-1}\cos y,$$

$$u(x,0) = e^{-x},\ u(x,1) = e^{-x}\cos 1.$$

The exact solution is $u = e^{-x}\cos y$. In this paper, we use the stopping criteria (2.2).

Table 4.1 shows the results on the interested region which is the third layer. In this experiment, the third layer with a mesh size h is located at [0.4, 0.6] × [0.4, 0.6]; the second layer with a mesh size h2 is located within [0.3, 0.7] × [0.3, 0.7], and the outside is the third layer covered by coarse grids with mesh size H. When H, h2 and h are halved, the error is reduced by approximately 1/2.

**Table 4.1:** Error on the interested domain with various H, h2, h for MP 1

| *H* | *h2* | *h* | Error on Interested Domain for MP11 |
|---|---|---|---|
| 1/10 | 1/20 | 1/40 | 5.7478E-04 |
| 1/20 | 1/40 | 1/80 | 3.2757E-04 |
| 1/40 | 1/80 | 1/160 | 1.8158E-04 |
| 1/80 | 1/160 | 1/320 | 9.8135E-05 |

Next we investigate the effectiveness of the three-layer scheme compared with the uniform grid scheme.

Table 4.2 shows the comparison of computation time and 2-norm error with various H, h2 and h for the model problem. When h is very small, the experiments show that the uniform grid scheme obtained much better accuracy than that of the three-layer scheme. However, the time spent on the uniform grid scheme is about six times more than the time used in the multi-layer scheme.

**Table 4.2:** Computation time and the 2-norm error with various H, h2, h for MP1 (The three-layer scheme vs. the uniform grid scheme)

| *H* | *h2* | *h* | Matrix Size | Error on the Interested Domain | Iteration Number | Matrix Gen. Time | Iteration Time | Total Time |
|---|---|---|---|---|---|---|---|---|
| 1/10 | 1/20 | 1/40 | 193 | 5.7478E-04 | 14 | 0.6890 | 0.0800 | 0.7690 |
| | | 1/40 | 1521 | 5.8146E-06 | 32* | 3.7720 | 0.1020 | 3.8740 |
| 1/10 | 1/20 | 1/80 | 401 | 3.2757E-04 | 25 | 2.4760 | 0.1160 | 2.5920 |
| | | 1/80 | 6241 | 1.4530E-06 | 65 | 15.5610 | 0.8180 | 16.3790 |
| 1/10 | 1/20 | 1/160 | 1201 | 1.8158E-04 | 40 | 10.2080 | 0.2630 | 10.4710 |
| | | 1/160 | 25281 | 3.6263E-07 | 130* | 71.0110 | 7.3750 | 78.3860 |
| 1/10 | 1/20 | 1/320 | 4337 | 9.8135E-05 | 83* | 73.8190 | 2.2850 | 76.1040 |
| | | 1/320 | 101761 | 9.0732E-08 | 250* | 383.5680 | 75.2930 | 458.8610 |

Both the two-layer scheme and the three-layer scheme took much less computation time than the uniform scheme. We now investigate the effectiveness of the three-layer scheme that is compared with the two-layer scheme.

The following Table 4.3 shows the error of the third layer located at [0.4, 0.6] × [0.4, 0.6], and the computation time for both the two-layer scheme and the three-layer scheme.

**Table 4.3:** Computation time and the 2-norm error with various H, h2, h for MP1 (The three-layer scheme vs. the

two-layer scheme)

| H | h2 | h | Second Layer | Size | Error on Interested Domain | Iteration Number | Split Time | Total Time |
|---|---|---|---|---|---|---|---|---|
| 1/10 | 1/20 | 1/40 | [0.3,0.7] | 193 | 5.75E-04 | 14 | 0.69 | 0.77 |
| | | | | | | | 0.08 | |
| 1/10 | 1/20 | 1/40 | [0.2,0.8] | 257 | 4.28E-04 | 15 | 0.94 | 1.01 |
| | | | | | | | 0.07 | |
| 1/10 | | 1/40 | | 153 | 6.25E-04 | 11 | 0.42 | 0.43 |
| | | | | | | | 0.01 | |
| 1/20 | | 1/40 | | 417 | 2.33E-04 | 18* | 1.07 | 1.10 |
| | | | | | | | 0.03 | |
| 1/20 | 1/40 | 1/80 | [0.3,0.7] | 777 | 3.28E-04 | 25 | 2.48 | 2.60 |
| | | | | | | | 0.12 | |
| 1/20 | 1/40 | 1/80 | [0.2,0.8] | 1025 | 2.45E-04 | 31* | 2.87 | 2.94 |
| | | | | | | | 0.07 | |
| 1/20 | | 1/80 | | 625 | 3.87E-04 | 21 | 1.64 | 1.67 |
| | | | | | | | 0.03 | |
| 1/40 | | 1/80 | | 1729 | 1.24E-04 | 35 | 4.45 | 4.58 |
| | | | | | | | 0.13 | |
| 1/40 | 1/80 | 1/160 | [0.3,0.7] | 3121 | 1.82E-04 | 40 | 10.21 | 10.47 |
| | | | | | | | 0.26 | |
| 1/40 | 1/80 | 1/160 | [0.2,0.8] | 4097 | 1.37E-04 | 52 | 14.04 | 14.47 |
| | | | | | | | 0.43 | |
| 1/40 | | 1/160 | | 2529 | 2.39E-04 | 46 | 6.58 | 6.81 |
| | | | | | | | 0.23 | |
| 1/80 | | 1/160 | | 7041 | 6.57E-05 | 58 | 22.74 | 23.96 |
| | | | | | | | 1.22 | |
| 1/80 | 1/160 | 1/320 | [0.3,0.7] | 12513 | 9.81E-05 | 83* | 73.82 | 76.11 |
| | | | | | | | 2.29 | |
| 1/80 | 1/160 | 1/320 | [0.2,0.8] | 16385 | 7.50E-05 | 97 | 119.26 | 122.75 |
| | | | | | | | 3.49 | |
| 1/80 | | 1/320 | | 10177 | 1.39E-04 | 83 | 29.94 | 31.80 |
| | | | | | | | 1.86 | |
| 1/160 | | 1/320 | | 28417 | 3.63E-05 | 104* | 138.92 | 146.73 |
| | | | | | | | 7.81 | |

For all the experiments, we let the interested area be [0.4, 0.6] × [0.4, 0.6]. In the above tables, we also consider two different second-layers that are [0.3, 0.7] × [0.3, 0.7] and [0.2, 0.8] × [0.2, 0.8]. Of course, the matrix size is larger when the second layer is larger. In our experiments, the two different second-layers give us a slight difference in the matrix size when h2 is relatively large. But when h2 is small, the matrix size may differ by 30% and the computation time may have a significant difference.

There are two different coarse grid sizes in the two-layer scheme. One is the same as the coarse grid size H of the three-layer scheme; the other is the same as the grid size h2. As expected, the larger linear system which corresponds to finer mesh size produces the better accuracy but requires more time. In the table, there is a "split time" column which contains the matrix generation time on the top of the iteration time. The following figures show the comparisons of the computation time with $n = \frac{1}{10h}$.

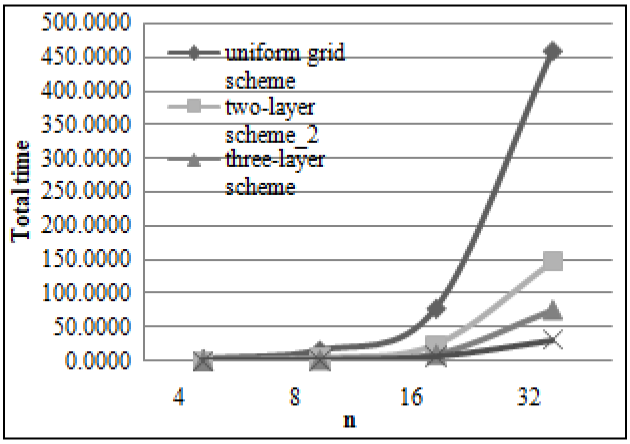


**Figure 4.1:** Total time vs. n for MP1 (second layer: [0.3, 0.7] × [0.3, 0.7])

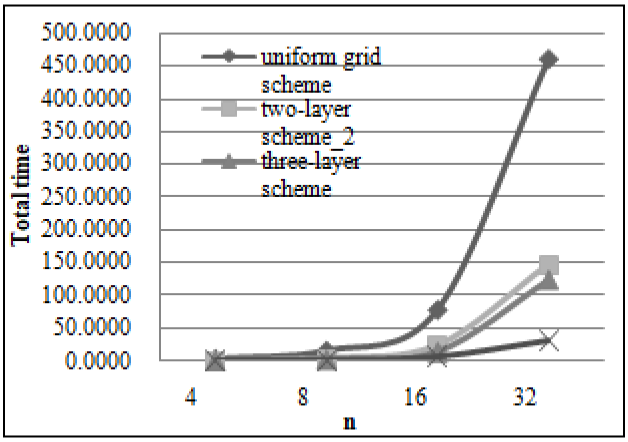


**Figure 4.2:** Total time vs. n for MP1 (second layer: [0.2, 0.8] × [0.2, 0.8])

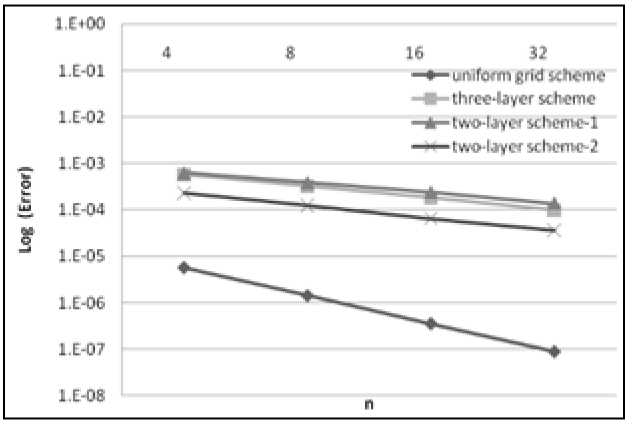


**Figure 4.3:** Log(error) vs. n for MP1 (second layer: [0.3, 0.7] × [0.3, 0.7])

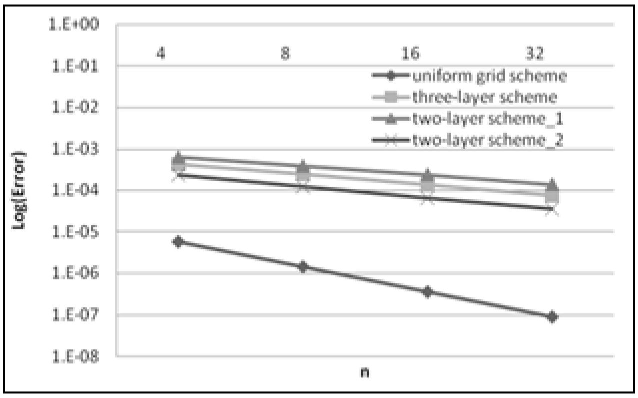


**Figure 4.4:** Log(error) vs. n for MP1 (second layer: [0.2, 0.8] × [0.2, 0.8])

## 5. Conclusions

In this paper, we have discussed the three layer scheme of a grid refinement scheme for solving a partial differential equation over a rectangular domain with Dirichlet boundary conditions. A model problem is conducted. The solutions obtained by the multi-layer scheme are more efficient than that of the uniform grid scheme. When compared with two layer scheme, it is hoped that the multi-layer scheme might produce better improvement; our experimental outcome does not show the desired results. In our experiments, the two-layer scheme is more economical, and the optimal performance is obtained when the mesh ratio between the coarse grid and the fine grid equals to 2.

## References

[1] S. Mehl and M. C. Hill, Development and evaluation of a local grid refinement method for block-centered finite-difference groundwater models using shared nodes, *Advances in Water Resources*, vol. 25**,** 487-511, 2002.

[2] S. Mehl and M.C. Hill, Three-dimensional local grid refinement for block-centered finite-difference groundwater models using iteratively coupled shared nodes: a new method of interpolation and analysis of errors, *Advances in Water Resources*, vol. 27, 899-912, 2004.

[3] I.N. Bronshtein and K. A. Semendyayev, Handbook of Mathematics, 3rd ed. Springer-Verlag, New York, 1997.

[4] H. Jeffreys and B.S. Jeffreys, Methods of Mathematical Physics, 3rd ed. Cambridge, Cambridge University Press, England, 1988.

[5] David M. Young, Iterative Solution of Large Linear Systems, Academic Press, 1971.

[6] Y. Saad and Yousef, Iterative Methods for Sparse Linear Systems, PWS Publishing Company, 1996.

[7] K. Atkinson, An introduction to numerical analysis (2nd ed.), John Wiley and Sons, 1988.

[8] Mai, T. Z., & Wu, L. Multi-layer Grid Refinement Method. *International Journal of Engineering Science and Technology*, 4(7), 3521− 3530, 2012.

[9] Mai, T. Z., & Wu, L. The successive over relaxation method in multi-layer grid refinement scheme. *Advances in Applied Science Research*, 4(2), 163-168, 2013.

[10] Stoer, Josef; Bulirsch, Roland. Introduction to Numerical Analysis (3rd ed.). Berlin, New York: Springer-Verlag, 2002.

[11] Cheng, A. and D. T. Cheng. Heritage and early history of the boundary element method, *Engineering Analysis with Boundary Elements*, vol. 29, 268–302, 2005