The Composite Grid Method for Singular Problems of Partial Differential Equations

Hai Ye 1,* and Yajun Xie 2,†

1 Department of Mathematics and Physics, Fujian Health College, Fuzhou 350101, China
2 School of Big Data, Fuzhou University of International Studies and Trade, Fuzhou 350202, China;
xyj@fzfu.edu.cn
* Correspondence: yehai@fjwzy.edu.cn or yehai2023@aliyun.com
† These authors contributed equally to this work.

Abstract: Partial differential equations are crucial in scientific computing, and this paper will consider some of the problems of partial differential equation singularities. The Composite Mesh Method (CGM) is a new and improved numerical method for solving partial differential equations based on existing numerical methods for finite elements. The method has two meshes over the entire domain—a coarse and a fine set. The two sets of meshes generated by Mesh3 are separate in their respective regions and do not nest or interact. This method improves the accuracy of solving the numerical solution of partial differential equations. This paper discusses the CGM method based on the Finite Element Program Generator (FEPG) and uses it to simulate several singular problems. The numerical simulation results show that the proposed method can obtain more satisfactory simulation results for global problems and use a smaller number of computational generations than the general finite element method.

Keywords: partial differential equations; composite grid method; finite element method; numerical simulation

MSC: 65M60; 35M32

1. Introduction

It is generally known that the following two-stage procedure can be used to derive the discretization of various time-varying problems in partial differential equations (PDEs). The PDE issue is first converted into a system of typically rigid ordinary differential equations with time as the independent variable by discretizing the spatial variables on a chosen spatial grid, primarily using finite difference or finite element approximations [1,2]. Singularity problems and problems with significant changes in local gradients, such as welding problems, often arise in scientific and engineering calculations. Over the past several years the interest in moving-grid methods has rapidly increased. For example, Williamson [3] analyses the reasons for the incomplete success of early grid methods in solving atmospheric evolution problems and briefly reviews test cases for dynamic core evaluation, focusing on spherical geometry and pole problems. More work can be viewed in the literature [4–6]. Excellent work has been conducted by many scholars on these two practical problems. For example, the local mesh refinement method, the fast adaptive composite grid method (FAC), and the global-local method (GLM) [7] are the most commonly used methods to solve these problems. These methods have been proven effective in solving these problems, but each has some things that could be improved. For example, the construction of local mesh refinement methods is based on smooth scale transitions over the global domain and high-quality irregular meshes, which are difficult to refine locally and cannot be better applied to 3D real-world environments; similarly, regular and nested meshes are required in FAC, but they require overly stringent conditions...
in practice; for GLM methods [8], coarse and fine meshes are only computed once. For the GLM method, the coarse and fine meshes are calculated only once, and there is no iterative process, so the accuracy of the solutions obtained by this method is low. The shortcomings of these methods have led to some research interest in the CGM method [9–11].

The CGM is constructed based on the method described above. Therefore, the CGM method is still a FAC method and can be seen as an improvement of the FAC method. Compared to methods such as GLM, the CGM method does not require regular and nested meshes, and it overcomes the difficulties of mesh re-division. In the CGM method, the need to use a set of coarse meshes for the whole domain and not to consider the effect of singularities is an advantage of the method. Secondly, in using a set of fine meshes for a local singular region, the singularities and the lack of influence of the two sets of meshes on each other need to be considered separately. This paper’s new iterative method is constructed to solve the global coarse grid problem and the local delicate grid problem from these advantages. Numerical experiments show that the iterative method we construct based on CGM can obtain better results with less computational consumption and higher accuracy of numerical solutions to coarse and fine grid accuracy, respectively.

The rest of the paper is organized as follows. Section 2 introduces the composite mesh method and a new iterative method for solving singular problems for partial differential equations constructed based on this method. Section 3 performs numerical simulations in which the proposed iterative scheme is applied to the elliptic chi-squared problem and the laser welding simulation. The conclusions of the paper are given in Section 4.

In this paper, we will use these notations for the following: \( \Omega \) denotes the full domain of the polygon; \( T_H \) and \( T_h \) represent the coarseness of the mesh that divides the region, using \( T_H \) for a coarse mesh and \( T_h \) for a fine mesh; regions divided by meshes of different sizes are denoted by \( S_H \) and \( s_H \), respectively; and in each region, their bases are \( \{ \psi_j \}_{j=1}^{n_c} \) and \( \{ \phi_i \}_{i=1}^{n_f} \), respectively, and then an arbitrary element in the region can be linearly tabulated by the base.

2. Composite Grid Method

As mentioned in the introduction, the numerical methods of common singular partial differential equations typically involve reducing them to their corresponding variational form and then discretizing the boundary equations for solving. Thus, the finite element method was born. It is widely used, versatile, and very effective. The main steps in solving differential equation boundary value problems are as follows:

Step 1, find the variational problem corresponding to the original equation.

Step 2, partition the region to be solved \( \Omega \) and discretize it into a collection of finite cells. The shape of these cells can be any mesh size, but since we are now studying low-dimensional differential problems, triangles or quadrilaterals are generally chosen as the shape of the cell mesh.

Step 3, each cell is analyzed and the approximate variational equations are solved.

Next, we elaborate on the CGM algorithm and give an iterative method for solving singular partial differential equations following the above ideas.

2.1. Algorithm Description

In this section, we discuss the construction process of the CGM algorithm. For simplicity, the following chi-square elliptic boundary problem needs to be considered first:

\[
\begin{align*}
L_c v &= f_c, \quad \text{in} \ \Omega_c, \\
L_f u &= f_f, \quad \text{in} \ \Omega_f, \quad v|_\Gamma = u|_\Gamma.
\end{align*}
\] (1)
Moreover, the variational form of (1) will be used in the subsequent derivation. So it is necessary to give the following variational problem corresponding to the chi-square elliptic boundary problem:

\[
\begin{align*}
    (Lc v, \bar{v})_{\Omega_c} &= (f_c, \bar{v})_{\Omega_c}, \\
    (Lf u, \bar{u})_{\Omega_f} &= (f_f, \bar{u})_{\Omega_f}, \\
    v|_\Gamma &= u|_\Gamma.
\end{align*}
\] (2)

where \( \Omega \) is a polygonal global domain, \( \Omega = \Omega_c \cup \Omega_f \), \( \partial \Omega \) is the boundary of domain \( \Omega \) and \( \Gamma \) is the intersection of domain \( \Omega \) and \( \Omega_f \). \( Lc, Lf \) are elliptic differential operators on \( \Omega_c \) and \( \Omega_f \), respectively.

Then, meshing \( \Omega \) and \( \Omega_f \), respectively, we get coarse grid \( T_H \) and fine grid \( T_h \), see Figure 1 (here \( T_h \) is much smaller than \( T_H \) in mesh size, and the size of the grid depends on the particular application). \( T_H \) and \( T_h \) can be either a regular or irregular grid. \( S_H \) and \( S_h \) are finite element spaces, correspondingly.

Finally, we combine Equation (2) to get the following:

\[
\begin{align*}
    (Lc v - f_c, \bar{v})_{\Omega} &= (Lc u_c - f_c, \bar{v})_{\Omega_f} - (Lf u_f - f_f, \bar{v})_{\Omega_f}, \\
    (Lf u, \bar{u})_{\Omega_f} &= (f_f, \bar{u})_{\Omega_f}, \\
    v|_\Gamma &= u|_\Gamma.
\end{align*}
\] (3a)

\[
(\bar{L}_f u, \bar{u})_{\Omega_f} = (\bar{f}_f, \bar{u})_{\Omega_f},
\]

\[
v|_\Gamma = u|_\Gamma.
\] (3b)

\( u_c \) and \( u_f \) are solutions of the coarse and fine grid operator on the fine grid region, respectively.

CGM calculates on two sets of grids, which are generated separately and do not restrict each other. The finite element method solves equations on coarse and fine grids iteratively. Calculated results in the coarse grid are mapped into the boundary of the refined grid through an interpolation matrix. On the boundary of the fine grid, interpolation matrix \( D \) is introduced to transform the energy integral between basis functions of two finite element spaces (coarse \( S_H \) and fine \( S_h \)) into the energy integral of bases of the same finite element space. Then, those results in the coarse grid are corrected by residuals calculated in the fine grid. This method can also be suitable for irregular grids. Authors J. Xu and A. Zhou [12,13] demonstrated that with the same material parameters in global and local domains, the numerical solution of this method can reach its precision on the coarse and fine grids.

Let \( \{\phi_i\}_{i=1}^{n_c} \) be the basis of \( S_h \) and \( \{\phi_i\}_{j=1}^{n_f} \) be the basis of \( S_H \). For functions of coarse-grid finite element space, which are restricted to the fine grid region, they can be expressed as a linear combination of the basis of fine-grid finite element space, so that \( \Psi = D\Phi \),

Figure 1. Composite grid model.
$D_{n_c \times n_f}$ is an interpolating matrix; here, $n_f$ and $n_c$ are element nodal elements of fine and coarse grids. We have

$$u = \sum_i u_i \phi_i = U^T \Phi, \quad v = \sum_j v_j \psi_j = V^T \Psi = V^T D \Phi, \quad \text{(in $\Omega_f$)}.$$ 

(4)

Set

$$U = (u_1, u_2, \cdots, u_{n_f})^T, \quad V = (v_1, v_2, \cdots, v_{n_c})^T,$$

$$\Phi = (\phi_1, \phi_2, \cdots, \phi_{n_f})^T, \quad \Psi = (\psi_1, \psi_2, \cdots, \psi_{n_c})^T.$$

Equation (3a) yields the following:

$$\left( L_c v - f_c, v \right)_{\Omega_f} = \left( L_c V^T \Psi - f_c, V^T \Psi \right)_{\Omega_f} = \left( A_c V - F_{\Omega_f} \right);$$

$$\left( L_c u - f_c, u \right)_{\Omega_f} = \left( L_c U^T \Phi - f_c, U^T \Phi \right)_{\Omega_f} = \left( A_c U - F_{\Omega_f} \right).$$

Therefore,

$$(A_c V - F_{\Omega_f}) = D \left( A_c^{\Omega_f} U_c - F_{\Omega_f} \right) - D \left( A_f^{\Omega_f} U_f - F_f^{\Omega_f} \right).$$

(6)

Here,

$$A_c = (L_c \Psi, \Psi)_{\Omega_f}, \quad A_f^{\Omega_f} = (L_f \Phi, \Phi)_{\Omega_f}, \quad A_f^{\Omega_f} = (L_c \Phi, \Phi)_{\Omega_f},$$

$$F_{\Omega_f} = (f_c, \Psi)_{\Omega_f}, \quad F_f^{\Omega_f} = (f_c, \Phi)_{\Omega_f}, \quad F_f^{\Omega_f} = (f_c, \Phi)_{\Omega_f}.$$ 

Similarly, from Equation (3b) we have the following:

$$\left( L_f u_f, u_f \right)_{\Omega_f} = \left( L_f A_f^{\Omega_f} U_f \right),$$

$$\left( f_f, u_f \right)_{\Omega_f} = \left( L_f A_f^{\Omega_f} U_f \right), \quad u |_{\Gamma} = v |_{\Gamma}.$$ 

Hence,

$$\left( L_f U^T A_f^{\Omega_f} U \right) = \left( L_f F_f^{\Omega_f}, u \right) |_{\Gamma} = v |_{\Gamma}.$$ 

(7)

**Theorem 2.1.** For any initial vectors $u_0^c$, $u_0^f$. The CGM algorithm produces an iterative sequence $v^n$ that is convergent and satisfies the following relation for any $\epsilon > 0$:

$$\|v^{n+1} - v^n\| < \epsilon$$

The theory of convergence of CGM algorithms is very well established and can be seen in the theoretical analyses of many papers in the lattice direction; the theoretical analysis of Theorem 2.1 is omitted here.

**2.2. Algorithm Scheme**

The process of CGM is as follows:

**Step 1 (Initialize)** Set $n = 0$, $u_0^c = 0$, $u_0^f = 0$;
When PDE is written in FE Language \cite{14,15}, the FEPG will generate calculation programs (see Figure 2). After discretization, the algebraic equations are solved by the successive over-relaxation (SOR) iterative method. The simulation uses finite element software FEPG (Finite Element Program Generator). Consider the elliptical homogeneous boundary value problems:

\begin{equation}
\left\{ \begin{array}{ll}
\Delta u = 0, & \Omega = (-1,1) \times (-1,1), \\
u = r^{0.5} \cos \left( \frac{\theta}{2} \right), & \partial \Omega
\end{array} \right. \tag{8}
\end{equation}

The genuine solution is \( u = r^{0.5} \cos \left( \frac{\theta}{2} \right) \), where \( r \) is obviously the long radius of the ellipse. There is a crack on the sub-domain \( \{ 0 \leq x < 1, y = 0 \} \) of \((-1,1) \times (-1,1)\), the genuine solution on the upper (\( \theta = 0 \)) and lower (\( \theta = 2\pi \)) boundary of the crack are \( r^{0.5} \) and \(-r^{0.5}\) respectively. For antisymmetry \((u(x,-y) = -u(x,y))\), we only consider the solution on \((-1,1) \times (0,1)\) \cite{16}.

A rectangular element is used in this experiment. In CGM, the fine grid is used in the local region where the crack is included, its mesh sizes are \( h = \frac{1}{8} \) and \( h = \frac{1}{16} \), respectively (see Figure 2). After discretization, the algebraic equations are solved by the successive over-relaxation (SOR) iterative method. \( v^{n+1} \) is the iterative solution between the coarse and fine grid, \( ||v^{n+1} - v^n|| < \epsilon \) is stopping condition (\( \epsilon = 10^{-5} \)).

Calculate with the general finite element method again; uniform grids are applied; the mesh sizes are \( h = \frac{1}{8} \) and \( h = \frac{1}{16} \), respectively. Choose \( A(-\frac{1}{4}, \frac{1}{4}), B(0, \frac{1}{4}), C(\frac{1}{4}, -\frac{1}{4}), D(\frac{1}{2}, \frac{1}{4}) \) and \( E(\frac{3}{4}, \frac{1}{4}), F(\frac{1}{4}, \frac{3}{4}) \) as observation points. The error of general FEM and CGM on these points are shown in Table 1.
Figure 2. Composite Grid (mesh size of fine grid $h = \frac{1}{8}$).

Table 1. The error of two methods ($\| \cdot \|_\infty$).

<table>
<thead>
<tr>
<th>Genuine Solution</th>
<th>General FEM Mesh Size</th>
<th>Error</th>
<th>Mesh Size of Fine Grid</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.00469</td>
<td>A 0.00458</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B 0.00790</td>
<td>B 0.00779</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\frac{1}{8}$</td>
<td>C 0.00801</td>
<td></td>
<td>$\frac{1}{8}$</td>
<td>C 0.00790</td>
</tr>
<tr>
<td>A 0.2275</td>
<td>D 0.00229</td>
<td></td>
<td>D 0.00229</td>
<td></td>
</tr>
<tr>
<td>B 0.3536</td>
<td>E 0.00070</td>
<td></td>
<td>E 0.00070</td>
<td></td>
</tr>
<tr>
<td>C 0.5493</td>
<td>A 0.00170</td>
<td></td>
<td>A 0.00149</td>
<td></td>
</tr>
<tr>
<td>D 0.7277</td>
<td>B 0.00309</td>
<td></td>
<td>B 0.00298</td>
<td></td>
</tr>
<tr>
<td>E 0.8777</td>
<td>D 0.00081</td>
<td></td>
<td>D 0.00079</td>
<td></td>
</tr>
<tr>
<td>$\frac{1}{16}$</td>
<td>E 0.00020</td>
<td></td>
<td>E 0.00020</td>
<td></td>
</tr>
</tbody>
</table>

3.2. The Simulation of Laser Welding

Now, we shall apply the CGM algorithm to simulate the result of laser welding. Laser welding, using a laser beam of high energy density as a heat source, is a highly efficient and precise welding method [17].

For the weldment to be symmetrical, the weld joint must be on the axis of symmetry of the weld bit. Therefore, the result of the welding process can be analyzed on the hemi-section. So numerical analysis is practiced on a 2D axisymmetrical cylinder coordinate system.

Coarse grid size is $20 \times 10$ mm, divided into 50 elements; fine grid is $5 \times 5$ mm in 625 elements, see Figure 3:
Figure 3. Composite Grid of weldment.

For the transient heat transfer problem, according to the Fourier heat transfer theorem and the energy conservation theorem, the heat-conduction control equation can be described as:

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot (\lambda \nabla T) + q$$  \hspace{1cm} (9)

In the equation, $\rho$ is the density of material, $c$ is the specific thermal capacity of the material, $\lambda$ is the heat conductivity of the material, $T$ is the temperature, $q$ is the heat source.

The Navier-Stokes equation is the partial differential equation that generally governs the flow phenomenon:

$$\left\{ \begin{aligned} \rho \frac{\partial u}{\partial t} + \rho u \cdot \nabla u + \nabla p &= \mu \Delta u + \rho f \\ \nabla \cdot u &= 0 \end{aligned} \right. $$  \hspace{1cm} (10)

where $t$ is the time, $x$ is the position vector in the Cartesian coordinate system, respectively; $u = u(x,t)$, $p = p(x,t)$, $\rho$ and $\mu$ denote velocity vector, pressure, density and kinematic viscosity coefficient, respectively. $f$ is the force.

In order to avoid the difficulties in computation, when we compute the approximate solution of the Navier-Stokes equations, the operator splitting method (OSM) is used. Here, the physical process described by the equations is decomposed into three processes: a diffusion process, a pressure-projection term and a convection process, and the finite element equation is established [18].

The governing equation corresponding to the Fluid Diffusive Part is as follows:

$$\rho \frac{\partial u}{\partial t} = \mu \Delta u + \rho f$$  \hspace{1cm} (11)

The Pressure-projection part is as follows:

$$\left\{ \begin{aligned} \rho \frac{\partial u}{\partial t} + \nabla p &= 0 \\ \nabla \cdot u &= 0 \end{aligned} \right. $$  \hspace{1cm} (12)

The corresponding governing equation of the fluid diffusive part is:

$$\rho \frac{\partial u}{\partial t} + \rho u \cdot \nabla u = 0$$  \hspace{1cm} (13)
The diffusion Equation (11) is solved first. The OSM algorithm is described as follows:

step 1: \( u_{n+\frac{1}{2}} = u_n + \left( \mu \Delta u_{n+\frac{1}{2}} + \rho f_n \right) \Delta t \)

step 2:

\[
\begin{cases} 
    u_{n+\frac{2}{3}} = u_{n+\frac{1}{2}} + \nabla p_{n+1} \Delta t \\
    \nabla \cdot u_{n+\frac{2}{3}} = 0 
\end{cases}
\]

step 3: \( u_{n+1} = u_{n+\frac{2}{3}} - u_{n+\frac{2}{3}} \cdot \nabla u_{n+\frac{2}{3}} \Delta t \)

The initial condition is as follows: when \( t = 0 \), the workpiece has a uniform primary temperature. The environmental temperature is commonly chosen as 27 °C.

In order to simulate the typical molten pool of high energy beam welding more precisely, which is in the shape of a nailhead with a large aspect ratio, the Rotary-Gauss body heat source model is established. The mathematical formula of the Rotary-Gauss body heat source model is as follows: ref. [19]:

\[
q(x, y, z) = \frac{c_s Q}{\pi H \left( 1 - \frac{1}{e^2} \right)} \exp \left[ \frac{-3c_s}{\ln \left( \frac{H}{z} \right)} \left( x^2 + y^2 \right) \right] \tag{14}
\]

\( c_s = \frac{3}{R_0^2} \) is the concentration coefficient of the shape of the heat source, \( R_0 \) is the opening radius of the heat source, \( H \) is the height of the heat source, \( Q \) is the heat source power. See Figure 4.

The time step is 1 ms and the heating time is 10 ms. Calculate by CGM. Rectangular element is used. The fine grid’s result can be see in Figure 5–7. The horizontal coordinate of Figures 5 and 6 consists of the time step and the vertical coordinate is the solder joint temperature.

Figure 4. Rotary-Gauss body heat source model.
We have selected the welding simulation graphs from the simulation software for two different time steps, see Figure 7. As can be seen in Figure 7, the left side graph is the temperature graph of the interface at the first time step and the right side is the temperature graph at the 22nd time step. Comparing the two plots, it can be seen that the MCG algorithm solves the soldering problem well.
Calculate using the general finite element method. Uniform grids are applied, with subdivision scales the same as the fine grid of CGM. See the temperature result of the solder joint in Table 2.

4. Conclusions

In this paper, we introduced the CGM, an improved FAC method. From the numerical examples, we can see that, because the fine grid only generates on the local singular domain, the grid number and nodal number are reduced effectively, which causes the labor of calculation to be reduced, saving the system resource and calculating time considerably. Thus, complex problems can be solved on low-configuration machines. This method improves the accuracy at the local singular domain, and the local solution can be obtained effectively. Compared with the global fine grid, the CGM’s accuracy nearly reached the FE analysis accuracy of the global fine grid. However, there is a massive order of magnitude difference in the labor of calculation between them. In CGM, $L_c$ and $L_f$ can be different operators. It is more suitable to solve more complex problems than other methods. The iterative number is reduced in CGM. When operators $L_c$ and $L_f$ satisfy $L_c = L_f$, iteration is convergent in one iterative step. When treating the problem of the local singular domain being moved to the global domain relatively, it is important to note that the two sets of grids are not nested. However, it is still possible to move a fine grid with its corresponding one by adding a control Fortran program.

The method can effectively improve the convergence speed of the iterative algorithm through the conversion between coarse and fine meshes, in addition to improving the accuracy of the numerical solution. Future research will focus on how to design a fast and stable composite mesh method to reduce the computation time and improve the numerical accuracy further.
Author Contributions: Methodology, H.Y.; Writing—original draft, H.Y.; Writing—review & editing, Y.X.; Visualization, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 11901098) and the Natural Science Foundation of Fujian Province, China (No. 2020J05034).

Data Availability Statement: Data will be made available on reasonable request.

Conflicts of Interest: The author declares that they have no conflict of interest.

References

5. Mak, H.W.L. Improved Remote Sensing Algorithms and Data Assimilation Approaches in Solving Environmental Retrieval Problems; Hong Kong University of Science and Technology: Hong Kong, China, 2019.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.