

A granular computing method for nonlinear convection-diffusion equation

Ya Lan Tian^{1,2}, Guo Yin Wang^{1,2,a}, Jian Jun Xu² and Ji Xu²

¹Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

²Institute of Electronic Information & Technology, Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing 401122, China

ABSTRACT: This paper introduces a method of solving nonlinear convection-diffusion equation (NCDE), based on the combination of granular computing (GrC) and characteristics finite element method (CFEM). The key idea of the proposed method (denoted as GrC-CFEM) is to reconstruct the solution from coarse-grained layer to fine-grained layer. It first gets the nonlinear solution on the coarse-grained layer, and then the function (Taylor expansion) is applied to linearize the NCDE on the fine-grained layer. Switch to the fine-grained layer, the linear solution is directly derived from the nonlinear solution. The full nonlinear problem is solved only on the coarse-grained layer. Numerical experiments show that the GrC-CFEM can accelerate the convergence and improve the computational efficiency without sacrificing the accuracy.

1 Introduction

As a key branch of PDEs, the convection-diffusion problem plays an important role in the field of theoretical as well as applied research.

In this paper, we consider the following NCDE in \mathbb{R}^2 :

$$\begin{cases} c \cdot u_t + b \cdot \nabla u - \nabla \cdot (a \nabla u) = f(u), & \text{in } \Omega \times (0, T] \\ u(x, t) = 0, & \text{on } \partial\Omega \times (0, T] \\ u(\cdot, 0) = u_0, & \text{on } \Omega \times \{0\} \end{cases} \quad (1.1)$$

Where $\Omega \subset \mathbb{R}^2$ is a bounded domain with boundary $\partial\Omega$, $\nabla = (\partial/\partial x_1, \partial/\partial x_2)^T$, $x = (x_1, x_2)^T$, $a = (a_1, a_2)^T$ and $b = (b_1, b_2)^T$ is a vector function, $f(u) = f(u, x, t)$ is a nonlinear function.

$W^{m,p}(\Omega)$ is Sobolev Space[1] on Ω , m is non-negative integer, the Sobolev Space norm is:

$$\|v\|_{W^{m,p}} = \begin{cases} \left(\sum_{|\alpha| \leq m} \|D^\alpha v\|_{L^p(\Omega)}^p \right)^{\frac{1}{p}}, & 1 \leq p < \infty \\ \max_{|\alpha| \leq m} \text{ess sup} |D^\alpha v|, & p = \infty \end{cases} \quad (1.2)$$

We write: $W^{m,2}(\Omega) = H^m(\Omega)$, $\|v\|_{L^2(\Omega)} = \|v\|_0 = \|v\|$, $\|v\|_{H^m(\Omega)} = \|v\|_m$.

Supposing problem (1.1) satisfies the following assumptions:

- $0 < c_* \leq c \leq c^* < +\infty$, $0 < a_* \leq a \leq a^* < +\infty$, $\max|a| < \max|b| < c_1$
- $|b/c| + |\nabla \cdot (b/c)| \leq c_2$
- $|\partial f(s)/\partial s| + |\partial^2 f(s)/\partial s^2| \leq c_3$
- $u \in L^\infty(0, T; H^{r+1}(\Omega))$
- $u_t \in L^2(0, T; H^{r+s})$, $r=1$ if $s=1$ and $r \geq 2$ if $s=0$
- $\partial^2 u / \partial t^2 \in L^2(0, T; L^2(\Omega))$

where $c_*, c^*, a_*, a^*, c_1, c_2, c_3$ are positive constant.

In most applications, PDEs are often difficult to get the analytic solutions. Therefore, numerical methods must be applied. At present, the major numerical methods are finite element methods(FEM)[2][3][4][5], finite difference methods (FDM)[6][7], finite volume methods(FVM)[8][9][10]. However, when the FEM or FDM are used to solve the NCDE, it exhibits excessive numerical diffusion and nonphysical oscillation[11]. To address this issue, we use the modified method of CFEM developed by Douglas-Russell[12]. But it is of high computational complexity and low efficiency.

Originally, in 1965, Zadeh, proposed the GrC. GrC is a natural mode to simulate human thinking and solve the large-scale problem [13]. For many practical problems, we cannot get the exact solution. Instead of searching for the exact solution, one may search for good approximate solutions. The basic components of GrC include granule, granulated views and levels, hierarchies, and granular structures [14].

As far as we know GrC theory is not applied to the NCDE. In this paper, we consider the GrC for NCDE, and focus on the reconstruction of solution from coarse-grained layer to fine-grained layer. We first gets the nonlinear solution on the coarse-grained layer, and then the function $f(x)$ is applied to linearizing the NCDE. Switch to the fine-grained layer, the linear solution is directly derived from the nonlinear solution. The full nonlinear problem is solved only on the coarse-grained layer. The new way of GrC-CFEM can accelerate the convergence and improve the computational efficiency

^a Corresponding author: wangguoyin@cigit.ac.cn

without sacrificing the accuracy.

The rest of this paper is organized as following. In Section 2, the general theories of Characteristic finite element discretization and Granular computing are described. In Section 3, we illustrate the applications of GrC-CFEM method to the NCDE. Section 4 gives convergence analysis and time complexity analysis. Section 5 presents a numerical experiment. Finally, conclusions are given in Section 6.

2 Preliminaries

2.1 Characteristic finite element discretization

The concept of CFEM is now generally well understood, this article only provides a brief introduction. Let

$$\psi(x) = \sqrt{c^2(x) + b^2(x)}, \quad (2.1)$$

and let the characteristic direction associated with the operator $c(x)\frac{\partial u}{\partial t} + b(x)\frac{\partial u}{\partial x}$ be denoted by $\tau = \tau(x, t)$, where

$$\frac{\partial}{\partial \tau} = \frac{c(x)}{\psi(x)} \frac{\partial}{\partial t} + \frac{b(x)}{\psi(x)} \frac{\partial}{\partial x}. \quad (2.2)$$

Then Eq. (1.1) can be put in form

$$\begin{cases} \psi \cdot u_\tau - \nabla \cdot (a \nabla u) = f(u), & \text{in } \Omega \times (0, T] \\ u(x, t) = 0, & \text{on } \partial \Omega \times (0, T] \\ u(\cdot, 0) = u_0, & \text{on } \Omega \times \{0\} \end{cases} \quad (2.3)$$

Sobolev space $V = H_0^1(\Omega)$, let $A(u, v) = (a \nabla u, \nabla v)$, and the notation $(u, v) = \int_{\Omega} u(x)v(x)dx$, $\forall u, v \in V$. Then, applying Green formula, (2.3) can be written in the equivalent variations:

$$\begin{cases} (\psi \cdot u_\tau, v) + A(u, v) = (f(u), v), & v \in V \\ A(u(0) - u_0, v) = 0, & v \in V \end{cases} \quad (2.4)$$

We consider a time step $\Delta t = T/N$ and approximate the solution at times $t^n = n \cdot \Delta t$, $n = 1, 2, \dots, N$. The characteristic derivative is approximated in the following way at $t = t^n$:

$$(\psi \cdot u_\tau)^n \approx \psi \frac{u(x, t^n) - u(\bar{x}, t^{n-1})}{\sqrt{(x - \bar{x})^2 + \Delta t^2}} = c(x) \frac{u(x, t^n) - u(\bar{x}, t^{n-1})}{\Delta t}. \quad (2.5)$$

Where $\bar{x} = x - [b(x)/c(x)]\Delta t$ is the foot (at level t^{n-1}) of the characteristic corresponding to x at the head (at level t^n).

We denote the grid by Δh whose grid size is h , V_h be a finite-element subspace of $V \cap W^{1,\infty}(\Omega)$. So, the CFEM for (2.4) is defined:

for $n = 1, 2, \dots, N$ find $u_h^n \in V_h$ such that

$$\begin{cases} \left(c \frac{u_h^n - \bar{u}_h^{n-1}}{\Delta t}, v \right) + A(u_h^n, v) = (f(u_h^n), v), & v \in V_h \\ A(u_h^0 - u_0, v) = 0, & v \in V_h \end{cases} \quad (2.6)$$

The initial approximation u_h^0 in V_h can be defined as any reasonable approximation of u_0 such as the interpolation of in V_h . The (2.6) determines $\{u_h^n\}$ uniquely in terms of the data u_0 .

2.2 Granular computing

This paper focuses on the reconstruction of solution based on the transformation between granule spaces [16]. In figure 1, fine-grained layer and coarse-grained layer are different from granular partitions on the same domain. $u(x, t)$ is NCDE. s and s' are numerical solutions of the function $u(x, t)$ on the domain with different granular partitions, respectively.

As we know, it is difficult to directly solve NCDE, furthermore, the acquisition of numerical solution maybe too costly. So, we explore GrC-CFEM to get s' . The key idea of the proposed method is to reconstruct the solution from coarse-grained layer to fine-grained layer. The full nonlinear problem is solved only on the coarse-grained layer. Comparing with the traditional method, the new method is faster and at a smaller cost. The new way can accelerate the convergence and improve the computational efficiency without sacrificing the accuracy.

3 Description of the GrC-CFEM

In this section, describe the new method. We combine GrC with CFEM to accelerate the solving process.

The method consists of two major steps:

Step1. On the coarse-grained ΔH , solve the following nonlinear system for $u_H^n \in V_H$:

$$\begin{cases} \left(c \frac{u_H^n - \bar{u}_H^{n-1}}{\Delta t}, v \right) + A(u_H^n, v) = (f(u_H^n), v), & v \in V_H \\ A(u_H^0 - u_0, v) = 0. & v \in V_H \end{cases} \quad (3.1)$$

Step2. Switch to the fine-grained Δh , $h = H^2$, (2.6) is linearized according to $f(\hat{u}_h^n) = f(u_H^n) + f'(u_H^n)(\hat{u}_h^n - u_H^n)$, the solution is reconstructed as follows:

$$\begin{cases} \left(c \frac{\hat{u}_h^n - \bar{\hat{u}}_h^{n-1}}{\Delta t}, v \right) + A(u_h^n, v) = (f(\hat{u}_h^n), v), & v \in V_h \\ A(u_h^0 - u_0, v) = 0. & v \in V_h \end{cases} \quad (3.2)$$

This linearization is efficient without sacrificing the accuracy associated with the fine-grained solution u_h^n .

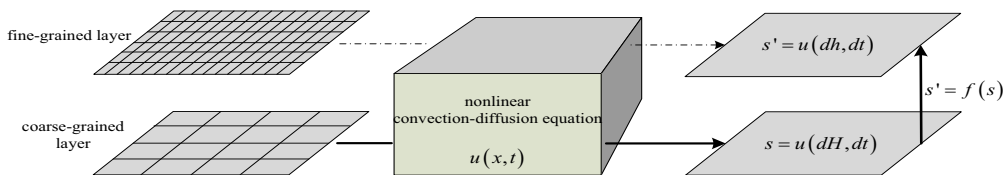


Figure 1. The flow chart of GRC-CFEM

4 Convergence analysis and time complexity analysis

In this section, the convergence analysis of algorithm are performed.

4.1 Convergence analysis

First, we briefly introduce a convergence analysis for the CFEM. Base on it, we give the GrC-CFEM.

We consider (1.1) on Ω . For $v \in V_h$ the following approximation property holds:

$$\inf_{u_h \in V_h} (\|v - v_h\| + h\|v - v_h\|) \leq Ch^{r+1} \|v\|_{r+1}, \quad (4.1)$$

where $r \geq 1$. Let $w_h : [0, T] \rightarrow V_h$, satisfy

$$A(u - w_h, v) = 0, \quad v \in V_h. \quad (4.2)$$

Set $\eta = u - w_h$, $\xi = u_h - w_h$, then $u - u_h = \eta - \xi$. It is well known that [18], for $p = 2, \infty$ and $1 \leq s \leq r + 1$,

$$\|\eta\|_{L^p(0, T; L^2(\Omega))} + h\|\eta\|_{L^p(0, T; H^1(\Omega))} \leq Ch^s \|u\|_{L^p(0, T; H^{r+1}(\Omega))}. \quad (4.3)$$

For $r \geq 2$ and $1 \leq s \leq r + 1$

$$\left\| \frac{\partial \eta}{\partial t} \right\|_{L^2(0, T; H^{-1}(\Omega))} + h \left\| \frac{\partial \eta}{\partial t} \right\|_{L^2(0, T; L^2(\Omega))} \leq Ch^s \left\| \frac{\partial u}{\partial t} \right\|_{L^2(0, T; H^{r+1}(\Omega))}. \quad (4.4)$$

The inverse property on V_h holds [19], namely, for $v_h \in V_h$,

$$\|v_h\|_{\infty} \leq h^{-1/2} \|v_h\|. \quad (4.5)$$

And we also have the approximation property:

$$\|u^n - w_h^n\|_{\infty} \leq \|u^n\|_{r+1, \infty} h^{r+1}. \quad (4.6)$$

Let u and u_h be the respective solutions of (2.4) and (2.6). Under assumptions (a-f) and Δt is sufficient small. To derive the following main result, some useful lemmas are needed [20][21].

When $s = 1, r = 1$ and $s = 0, r > 1$ we have the error estimate:

$$\max_{1 \leq n \leq N} \|u^n - u_h^n\| \leq C(\Delta t + h^{r+1}). \quad (4.7)$$

When $r \geq 2$, we have the error estimate:

$$\max_{1 \leq n \leq N} \|u^n - u_h^n\| \leq C(\Delta t + h^r). \quad (4.8)$$

Let u and \hat{u}_h be the respective solutions of (2.4) and (3.2). Under assumptions (a-f), and also assume that Δt is sufficient small, $r \geq 1$. We have:

$$\max_{1 \leq n \leq N} \|u^n - \hat{u}_h^n\| \leq C(\Delta t + h^{r+1} + H^{2r+3/2}). \quad (4.9)$$

Proof: Let $\eta^n = u^n - w_h^n$, $\hat{\xi}^n = \hat{u}_h^n - w_h^n$ at $t = t^n$. By (2.4), (3.2), choosing $v = \hat{\xi}^n$, we have

$$\begin{aligned} & (c \frac{\hat{\xi}^n - \hat{\xi}^{n-1}}{\Delta t}, \hat{\xi}^n) + A(\hat{\xi}^n, \hat{\xi}^n) \\ &= (\psi \frac{\partial u^n}{\partial \tau} - c \frac{u^n - \bar{u}_h^{n-1}}{\Delta t}, \hat{\xi}^n) + (c \frac{\eta^n - \eta^{n-1}}{\Delta t}, \hat{\xi}^n) + (c \frac{\eta^{n-1} - \bar{\eta}^{n-1}}{\Delta t}, \hat{\xi}^n) \\ & - (c \frac{\hat{\xi}^{n-1} - \bar{\xi}^{n-1}}{\Delta t}, \hat{\xi}^n) + ((f(u^n) - f(u_h^n)) - f'(u_h^n)(\hat{u}_h^n - u_h^n), \hat{\xi}^n) \end{aligned} \quad (4.10)$$

By $f(u^n) = f(u_h^n) + f'(u_h^n)(u^n - u_h^n) + \frac{1}{2} f''(\tilde{u})(u^n - u_h^n)^2$ for some function \tilde{u} . From (4.10), we have

$$\begin{aligned} & (c \frac{\hat{\xi}^n - \hat{\xi}^{n-1}}{\Delta t}, \hat{\xi}^n) + A(\hat{\xi}^n, \hat{\xi}^n) \\ &= (\psi \frac{\partial u^n}{\partial \tau} - c \frac{u^n - \bar{u}_h^{n-1}}{\Delta t}, \hat{\xi}^n) + (c \frac{\eta^n - \eta^{n-1}}{\Delta t}, \hat{\xi}^n) + (c \frac{\eta^{n-1} - \bar{\eta}^{n-1}}{\Delta t}, \hat{\xi}^n) \\ & - (c \frac{\hat{\xi}^{n-1} - \bar{\xi}^{n-1}}{\Delta t}, \hat{\xi}^n) + (f'(u_h^n) \eta^n, \hat{\xi}^n) - (f'(u_h^n) \xi^n, \hat{\xi}^n) \\ & + (\frac{1}{2} f''(\tilde{u})(u^n - u_h^n)^2, \hat{\xi}^n) = \sum_{i=1}^7 T_i'. \end{aligned} \quad (4.11)$$

Using (c) and $a(a-b) \geq \frac{1}{2}(a^2 - b^2)$. From (4.11), we get

$$(c \frac{\hat{\xi}^n - \hat{\xi}^{n-1}}{\Delta t}, \hat{\xi}^n) + A(\hat{\xi}^n, \hat{\xi}^n) \geq \frac{1}{2\Delta t} [(c \hat{\xi}^n, \hat{\xi}^n) - (c \hat{\xi}^{n-1}, \hat{\xi}^{n-1})] + \frac{a_0}{2} \|\hat{\xi}^n\|^2 \quad (4.12)$$

Using arguments similar to [22], plugging (4.12) in to equation (4.11), we can get

$$\begin{aligned} & \frac{1}{2\Delta t} [(c \hat{\xi}^n, \hat{\xi}^n) - (c \hat{\xi}^{n-1}, \hat{\xi}^{n-1})] + \frac{a_0}{2} \|\hat{\xi}^n\|^2 \\ & \leq C \left[\|\hat{\xi}^n\|^2 + \|\hat{\xi}^{n-1}\|^2 + \Delta t \left\| \frac{\partial^2 u}{\partial \tau^2} \right\|_{L^2(\tau^{n-1}, \tau^n; L^2)}^2 + \frac{1}{\Delta t} \left\| \frac{\partial u}{\partial t} \right\|_{L^2(\tau^{n-1}, \tau^n; H^{-1})}^2 \right. \\ & \quad \left. + \|\eta^n\|^2 + \|\eta^{n-1}\|^2 + \|(u^n - u_h^n)^2\|^2 \right]. \end{aligned} \quad (4.13)$$

Now multiplying both sides of (4.13) by $2\Delta t$, summing over n and applying the discrete Gronwall lemma. Note that $\hat{\xi}^0 = 0$, we have:

$$\max_{1 \leq n \leq N} \|\hat{\xi}^n\| \leq C \left[\Delta t \left\| \frac{\partial^2 u}{\partial \tau^2} \right\|_{L^2(0, T; L^2)}^2 + \left\| \frac{\partial u}{\partial t} \right\|_{L^2(0, T; H^{-1})} + \|\eta\|_{L^\infty(0, T; L^2)} + \max_{1 \leq n \leq N} \|u^n - u_h^n\|^2 \right] \quad (4.14)$$

Let $h = H$ in (4.7) and using (4.5) (4.6), we get

$$\begin{aligned} & \|(u^n - u_h^n)^2\| \leq \|u^n - u_h^n\|_{\infty} \|u^n - u_h^n\| \\ & \leq (\|u^n - w_h^n\|_{\infty} + \|w_h^n - u_h^n\|_{\infty}) \|u^n - u_h^n\| \\ & \leq C(H^{r+1} + H^{-1/2}(H^{r+1} + \Delta t))(H^{r+1} + \Delta t) \\ & \leq C(H^{2r+3/2} + \Delta t). \end{aligned} \quad (4.15)$$

Note that $u^n - \hat{u}_h^n = \eta^n - \hat{\xi}^n$. This together with (4.3), (4.4) (4.14) and (4.15) yields (4.9).

Let u and \hat{u}_h be the respective solutions of (2.4) and (3.2). Under assumptions (a-f) and Δt is sufficient small, $r \geq 2$, we have:

$$\max_{1 \leq n \leq N} \|u^n - \hat{u}_h^n\| \leq C(\Delta t + h^r + H^{2r}). \quad (4.16)$$

Proof: An optimal order estimate for $u^n - \hat{u}_h^n$ in $H^1(\Omega)$ can be derived in a similar fashion of the first proof.

4.2 Efficiency analysis

4.2.1 Time complexity analysis for the CFEM

We will analyze the basic calculation method of CFEM about Eq.(2.6):

The pseudo-code for CFEM is as follows:

The CFEM Algorithm

Input: //initial-boundary value
 $u(x,t)=0$, on $\partial\Omega\times(0,T]$; $u(\cdot,0)=u_0$, on $\Omega\times\{0\}$
m // $m=\Omega/h$. Ω is a bounded domain, h is spatial step
n // $n=T/\Delta t$. T is total time, Δt is time step
Output: u_h // Solutions of NCDE
 1: for $i \leftarrow 1$ to n do
 2: for $j \leftarrow 0$ to m do
 3: $u_h[i,j] \leftarrow \text{Eq. (2.6)}$;
 4: return u_h

The input size of this algorithm is $n \times m$. The key step to solve Eq. (2.6) is to solve nonlinear equation $f(u_h^n)$. We have following equation by Thayer Series Expansion:

$$f(x) = f(x_0) + f'(x_0)(x-x_0) + \frac{f''(x_0)(x-x_0)^2}{2!} + \dots + \frac{f^{(p)}(x_0)(x-x_0)^p}{p!} + R_p(x) \quad (4.17)$$

The basic operation of solving Thayer Series Expansion is derivation, exponentiation and factorial. Based on input size, the computation times of basic operation is:

$$C_1 = n \times m \times (1+2+\dots+(p-1)+p) = \frac{1}{2}(p^2+p)(n \times m) \quad (4.18)$$

During the calculation process of solving Thayer Series Expansion, considering p is limited. So the time complexity of CFEM is $o(n \times m)$.

4.2.2 Time complexity analysis for the GrC-CFEM

We will analyze the basic calculation method of GrC-CFEM about Eq. (3.1) and (3.2):

According to the Eq. (3.1) and (3.2), this algorithm has following pseudo-code:

The GrC-CFEM Algorithm

Input: // initial-boundary value
 $u(x,t)=0$, on $\partial\Omega\times(0,T]$; $u(\cdot,0)=u_0$, on $\Omega\times\{0\}$
M // $M=\Omega/H$, H is spatial step, $h=H^2$
n // $n=T/\Delta t$. T is total time, Δt is time step
Output: \hat{u}_h // Solutions of NCDE
 1: for $i \leftarrow 1$ to n do
 2: for $j \leftarrow 0$ to M do
 3: $u_H[i,j] \leftarrow \text{Eq. (3.1)}$;
 4: for $i \leftarrow 1$ to n do
 5: for $j \leftarrow 0$ to M do
 6: $\hat{u}_h[i,j] \leftarrow \text{Eq. (3.2)}$;
 7: return \hat{u}_h

CFEM directly solves the nonlinear problem on

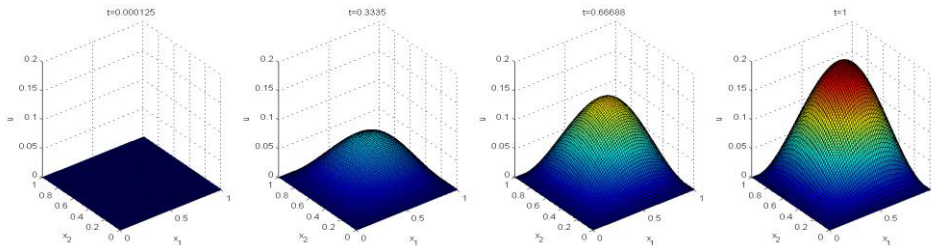


Figure 2. The numerical solution \hat{u}_h^n at different time with $H=1/8, h=1/64$ for example 1

fine-grained space, but GrC-CFEM can linearize the nonlinear problem. GrC-CFEM involves a nonlinear solution on the coarse-grained space and a linear solution on the fine-grained space.

The input size of this algorithm is $n \times M$. Based on input size, the computation times of basic operation is:

$$C_2 = (n \times M \times (1+2+\dots+(p-1)+p)) + (n \times M) = \frac{p^2+p+2}{2}(n \times M). \quad (4.19)$$

During the calculation process of solving Thayer Series Expansion, considering p is limited. So the time complexity of GrC-CFEM is $o(n \times M)$.

So, the GrC-CFEM is faster than the CFEM by one order of magnitude, because there is quadratic relation between m and M .

5 Numerical experiments

In this section, we provide numerical experiments to demonstrate the efficiency of the GrC-CFEM.

Example1: NCDE with constant coefficients:

$$\begin{cases} c \cdot u + b \cdot \nabla u - \nabla \cdot (a \nabla u) = f(u), & \text{in } \Omega \times (0, T] \\ u(x, t) = 0, & \text{on } \partial\Omega \times (0, T] \\ u(\cdot, 0) = 0, & \text{on } \Omega \times \{0\} \end{cases} \quad (5.1)$$

where $f(u) = -u^2 + G(x, t)$ is a nonlinear function, $G(x, t)$ is determined by the exact solution of the $u(x, t) = tx_1x_2(1-x_1)(1-x_2)e^{x_1+x_2}$, $x = (x_1, x_2)$, $c=1$, $b=(1,1)^T$, $a=0.001$, $\Omega=[0,1] \times [0,1]$, $\Delta t=1.25 \times 10^{-4}$.

Figure2 and Figure3 are the solution \hat{u}_h^n of (5.1) and its contour-line. The solution \hat{u}_h^n obtained by the GrC-CFEM method. The graph shows us how the concentration varies with time and how the concentration varied in space. Figure 4 is the coarse-grained ($H=1/4, h=1/16$) and fine-grained ($H=1/8, h=1/64$) solution \hat{u}_h^n of example1 obtained by the GrC-CFEM at $t=0.125$.

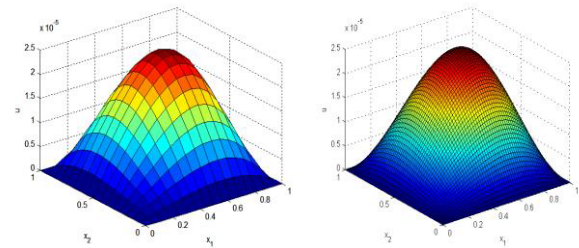


Figure 4. The coarse-grained solution (left) and fine-grained solution (right) \hat{u}_h^n at $t=0.125$ for example1 (GrC-CFEM)

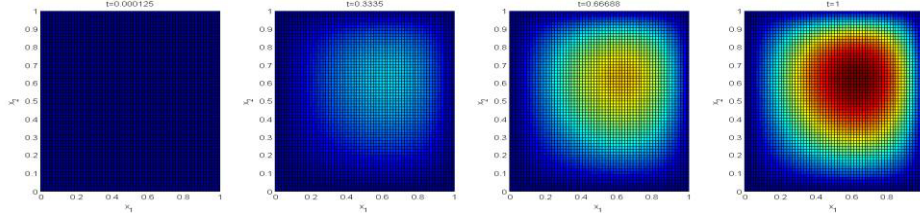


Figure 3. The contour-line of \hat{u}_h^n at different time with $H=1/8, h=1/64$ for example1

Table 1. L^2 errors of u_h at $t=0.125$ for example1 (CFEM)

h	$\ u_h - u\ /\ u\ $	CPU seconds
1/16	3.1163×10^{-4}	1.9052×10^2
1/64	1.2161×10^{-5}	2.5273×10^4

Table 2. L^2 errors of \hat{u}_h at $t=0.125$ for example1 (GrC-CFEM)

H	h	$\ \hat{u}_h - u\ /\ u\ $	CPU seconds
1/4	1/16	3.2151×10^{-4}	3.2916×10^1
1/8	1/64	2.1826×10^{-5}	6.4819×10^3

Table1 shows the L^2 errors of u_h and the CPU time for CFEM over the range of fine-grained Δh . Table2 shows the L^2 errors of \hat{u}_h and the CPU time for GrC-CFEM when $h=H^2$.

Figure 2-4 and Table 1-2 shows that the solution \hat{u}_h and u_h have the same order of accuracy. GrC-CFEM uses less computation time than CFEM.

Example2: In this example, we consider two-dimensional NCDE with anisotropic diffusion and a source term [23]:

$$\frac{\partial u}{\partial t} + \frac{\partial u^m}{\partial x} + \frac{\partial u^n}{\partial y} = D \left[d_1 \frac{\partial^2 u}{\partial x^2} + d_2 \frac{\partial^2 u}{\partial y^2} + 2d_3 \frac{\partial u}{\partial x \partial y} \right] + F. \quad (5.2)$$

The problem has the following analytical solution with proper initial and boundary conditions:

$$u(x, y, t) = \text{sech}[2(x + y - t)]. \quad (5.3)$$

and the source term F is defined as

$$F = 2g(u - mu^m - nu^n) - 4Du(d_1 + d_2 + 2d_3)(2g^2 - 1). \quad (5.4)$$

Where $g = \tanh[2(x + y - t)]$, $m = n = 2$, $d_1 = 1$, $d_2 = 2$, $d_3 = 0$ or 1 . The computational domain is fixed in $[-5, 5] \times [-5, 5]$.

In our simulation, the global relative error (GRE) is used to test the accuracy of the model, and defined by

$$GRE = \frac{\sum_j |\phi(X_j, t) - \phi^*(X_j, t)|}{\sum_j |\phi^*(X_j, t)|}. \quad (5.5)$$

where $\phi(X_j, t)$ and $\phi^*(X_j, t)$ are the numerical and analytical solutions, respectively.

Example2 solved by the GrC-CFEM. We present the results in Figure 5 where $D=0.01$, $dx=0.05$, $c=10$,

$d_3=1$, $t=0.5$. we can see the distribution and contour-line of numerical solution for example2.

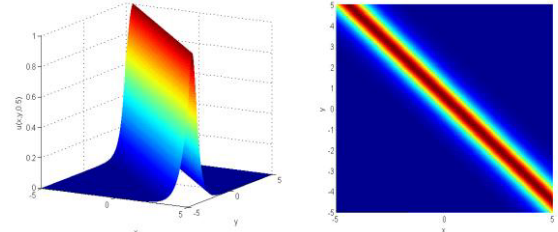


Figure5. the distribution(left) and contour-line(right) of numerical solution for example2

Furthermore, from Figure6, it can be seen that the numerical result also agrees well with the corresponding analytical solution. In addition, a straight comparison between GrC-CFEM and RLBM is presented in Table3.

We can find that, the accuracy of GrC-CFEM could reach the same order of magnitude as the RLBM.

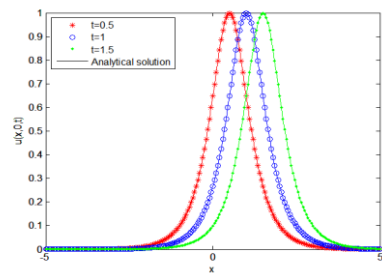


Figure6. The numerical and analytical solution of example2 at different time

Table 3. The distribution and contour-line of numerical solution for example2 (Blank mean that the model is unstable)

dx	c	Method	t	$D=0.1$		$D=0.01$	
				$d_3=0$	$d_3=1$	$d_3=0$	$d_3=1$
0.1	10	GrC-CFEM	0.5	6.7245×10^{-3}	5.8131×10^{-3}	1.1016×10^{-2}	2.0251×10^{-2}
			1.0	1.8240×10^{-2}	1.5024×10^{-2}	2.9481×10^{-2}	4.6162×10^{-2}
			1.5	3.0216×10^{-2}	2.4672×10^{-2}	4.6104×10^{-2}	6.7225×10^{-2}
		RLBM	0.5	6.5691×10^{-3}	5.4372×10^{-3}	1.1138×10^{-2}	2.0670×10^{-2}
			1.0	1.7245×10^{-2}	1.5144×10^{-2}	2.8303×10^{-2}	4.5020×10^{-2}
			1.5	2.8083×10^{-2}	2.5276×10^{-2}	4.4329×10^{-2}	6.5755×10^{-2}

0.05	15	GrC-CFEM	0.5	7.5312×10^{-3}	7.9412×10^{-3}	1.1802×10^{-2}	2.6115×10^{-2}	
			1.0	2.5512×10^{-2}	2.7514×10^{-2}	3.5456×10^{-2}	6.4008×10^{-2}	
			1.5	3.8412×10^{-2}	4.0215×10^{-2}	4.7145×10^{-2}	6.9541×10^{-2}	
	10	RLBM	0.5	7.4175×10^{-3}	7.8539×10^{-3}	1.1749×10^{-2}	2.3511×10^{-2}	
			1.0	2.0575×10^{-2}	2.2402×10^{-2}	3.0116×10^{-2}	6.4079×10^{-2}	
			1.5	3.4330×10^{-2}	3.8004×10^{-2}	4.7029×10^{-2}		
	0.05	10	GrC-CFEM	0.5	6.8154×10^{-4}	9.5479×10^{-4}	2.9004×10^{-3}	4.9214×10^{-3}
				1.0	1.4515×10^{-3}	2.0154×10^{-3}	7.0845×10^{-3}	9.0415×10^{-3}
				1.5	7.5124×10^{-4}	2.9054×10^{-3}	9.7145×10^{-3}	9.6418×10^{-3}
		5	RLBM	0.5	6.5523×10^{-4}	9.8464×10^{-4}	2.7444×10^{-3}	4.5346×10^{-3}
				1.0	1.2211×10^{-3}	1.9854×10^{-3}	6.5336×10^{-3}	8.9663×10^{-3}
				1.5	1.8660×10^{-3}	2.9241×10^{-3}	9.9516×10^{-3}	1.2648×10^{-2}
		15	GrC-CFEM	0.5	8.5012×10^{-4}	7.2148×10^{-4}	1.9152×10^{-3}	3.8415×10^{-3}
				1.0	2.7081×10^{-3}	1.1040×10^{-3}	4.8415×10^{-3}	6.7147×10^{-3}
				1.5	3.4243×10^{-3}	2.1017×10^{-3}	7.0145×10^{-3}	9.9621×10^{-3}
10	RLBM		0.5	8.4940×10^{-4}	7.4292×10^{-4}	1.7846×10^{-3}	3.1030×10^{-3}	
			1.0	2.1072×10^{-3}	1.2356×10^{-3}	4.2660×10^{-3}	6.7396×10^{-3}	
			1.5	3.5255×10^{-3}	2.1970×10^{-3}	6.9221×10^{-3}	9.9502×10^{-3}	

6 Conclusions

The key feature of the GrC-CFEM is that it can accelerate the solving process without sacrificing the order of accuracy. Solving the nonlinear problems on the fine-grained layer is reduced to solving a nonlinear system on coarse-grained layer and a linear system on the fine-grained layer. The GrC combined with the CFEM cannot only decreases the numerical oscillation caused by dominating convection, but also saves much computational time for solving the NCDE.

Acknowledgements

This work is supported by National Science and Technology Major Project (2014ZX07104-006).

References

- [1] Attouch H, Buttazzo G, Michaille G. *Variational analysis in Sobolev and BV spaces: applications to PDEs and optimization* [M]. Society for Industrial and Applied Mathematics: Mathematical Optimization Society, 2014.
- [2] Deckelnick K, Elliott C M, Ranner T. Unfitted finite element methods using bulk meshes for surface partial differential equations [J]. *Siam Journal on Numerical Analysis*, 2013, 52.
- [3] Wachter A. A comparison of the String Gradient Weighted Moving Finite Element method and a Parabolic Moving Mesh Partial Differential Equation method for solutions of partial differential equations[J]. *Central European Journal of Mathematics*, 2013, 11(4):642-663.
- [4] Ford N, Xiao J, Yan Y. A finite element method for time fractional partial differential equations [J]. *Fractional Calculus & Applied Analysis*, 2011, 14(3):454-474.
- [5] Tang Y, Xiang J, Xu J, et al. Adaptive Wavelet Finite Element Method for Partial Differential Equation [J]. *Advanced Science Letters*, 2011, volume 4:3151-3154(4).
- [6] Mohanty R K. An unconditionally stable finite difference formula for a linear second order one space dimensionally hyperbolic equation with variable coefficients [J]. *Applied Mathematics & Computation*, 2005, 165(1):229-236.
- [7] Fan C, Yeih P L W. Generalized finite difference method for solving two-dimensional inverse Cauchy problems [J]. *Inverse Problems in Science & Engineering*, 2014:1-23.
- [8] Shakeri F, Dehghan M. A high order finite volume element method for solving elliptic partial integro-differential equations [J]. *Applied Numerical Mathematics*, 2013, 65(2):105-118.
- [9] McCorquodale P, Dorr M R, Hittinger J A F, et al. High-order finite-volume methods for hyperbolic conservation laws on mapped multiblock grids [J]. *Journal of Computational Physics*, 2015, 288.
- [10] Feng L B, Zhuang P, Liu F, et al. Stability and convergence of a new finite volume method for a two-sided space-fractional diffusion equation[J]. *Applied Mathematics & Computation*, 2015, 257:52-65.
- [11] Chen Z, Chou S, Kwak D Y. Characteristic-mixed covolume methods for advection-dominated diffusion problems [J]. *Numerical Linear Algebra with Applications*, 2006, 13(9):677-697.
- [12] Douglas J, Russell A T F. *Numerical Methods for Convection-Dominated Diffusion Problems Based on Combining the Method of Characteristics with Finite Element or Finite Difference Procedures*[J]. *Siam Journal on Numerical Analysis*, 1982, 19(5):871-885.
- [13] Guoyin Wang, Qinghua Zhang, Jun Hu. An overview of granular computing [J]. *CAAI Transactions on Intelligent Systems*, 2008, 2(6): 8-26.
- [14] Yao Y Y. Granular Computing: basic issues and possible solutions [J]. *Proceedings of Joint Conference on Information Sciences*, 2002:186-189.
- [15] Qian Y, Zhang H, Li F, et al. Set-based granular computing: A lattice model [J]. *International Journal of Approximate Reasoning*, 2014, 55(3): 834-852.
- [16] Ji Xu, Guoyin Wang, Hong Yu. Review of Big Data Processing Based on Granular Computing [J]. *Chinese Journal of Computers*, 2015(8):1497-1517.
- [17] Gacek A. Granular modelling of signals: a framework of granular computing [J]. *Information Sciences*, 2013, 221: 1-11.
- [18] Douglas J, Russell T F. *Numerical Methods for Convection-Dominated Diffusion Problems Based on Combining the Method of Characteristics with Finite Element or Finite Difference Procedures*[J]. *Siam Journal on Numerical Analysis*, 1982, 19(5):871-885.
- [19] Pasciak J E. *The Mathematical Theory of Finite Element Methods* (Susanne C. Brenner and L. Ridgway Scott)[J]. *Siam Review*, 1995, 37(3):472-473.
- [20] Fu H. A characteristic finite element method for optimal control problems governed by convection-diffusion equations [J]. *Journal of Computational & Applied Mathematics*, 2010, 235(3):825-836.
- [21] Ciarlet P G. *The finite element method for elliptic problems* [M]// *Society for Industrial and Applied Mathematics*, 2002:460.
- [22] Kaitai Li. *The finite element method and its application* [M]// *XI'AN Jiaotong University Press*, 1984.
- [23] Wang L, Shi B, Chai Z. Regularized lattice Boltzmann model for a class of convection-diffusion equations[J]. *Physical Review E*, 2015, 92(4)