Iterative solvers for generalized finite element solution of boundary-value problems

M Shadi Mohamed[*] Mohammed Seaid[†] Abderrahman Bouhamidi[‡]

Abstract

Most of generalized finite element methods use dense direct solvers for the resulting linear systems. This is mainly the case due to the ill-conditioned linear systems that are associated with these methods. In the current study we investigate the performance of a class of iterative solvers for the generalized finite element solution of time-dependent boundary-value problems. A fully implicit time-stepping scheme is used for the time integration in the finite element framework. As enrichment we consider a combination of exponential functions based on an approximation of the internal boundary layer in the problem under study. As iterative solvers we consider the generalized minimal residual method and the changing minimal residual method based on the Hessenberg reduction. Compared to dense direct solvers, the proposed solvers achieve high accuracy and efficiency at low computational cost and memory storage. Two test examples for boundary-value problems in two space dimensions are used to assess the performance of the iterative solvers. Comparison to dense direct solvers widely used in the framework of generalized finite element methods is also presented. The obtained results demonstrate the ability of the considered iterative solvers to capture the main solution features. It is also illustrated that this class of iterative solvers can be efficient in solving the ill-conditioned linear systems resulting from the generalised finite element methods.

Keywords. Generalized finite element methods; Partition of unity; Boundary-value problems; Krylov subspace methods; Iterative solvers; Composite materials; Ill-conditioning

1 Introduction

Although they have different names such as the Generalized Finite Element Method (GFEM) [32], the eXtended Finite Element Method (XFEM) [34] and the Partition of Unity Finite Element Method (PUFEM) [17] but the enrichment idea in all these methods remains the same [19]. The idea is to enrich the finite element space with functions that can reproduce oscillatory solutions or solutions with singularities and/or discontinuities. So far it has been proven that the enrichment can circumvent the need for highly refined meshes that are otherwise necessary to recover oscillations or irregularities. Hence, the enrichment significantly increases the efficiency of the finite element method. Since their introduction, the enrichment functions became very popular when dealing with challenging numerical issues. In the past two decades a large amount of work was dedicated to developing enrichment methods and their applications, not only in the finite element method but also in the boundary element method [26, 27] and other numerical methods [18]. For a general review on these methods and related techniques we refer to [4, 12] and further references are therein.

More recently enriched finite element methods were also extended to deal with time-dependent problems including conduction-radiation applications, see for example [21, 22, 6, 23]. A common requirement for all generalized finite element methods is the solution of linear systems of algebraic equations resulting from the spatial discretization of the boundary-value problem under study. The structure of these linear

[*] School of Energy, Geoscience, Infrastructure and Society, Heriot-Watt University, WA 2.28A, Edinburgh EH14 4AS, UK

[†] School of Engineering and Computing Sciences, University of Durham, South Road, Durham DH1 3LE, UK

[‡] Université Lille-Nord de France, ULCO, L.M.P.A, F-62228 Calais-Cedex, France

systems substantially differs from those associated with the non-enriched finite element methods. For time-dependent boundary-value problems these linear systems need to be solved at each timestep during the time integration process. The above mentioned generalized finite element methods have shown the potential to provide highly accurate results using a far lower number of degrees of freedom compared to their conventional finite element methods counterpart. However, the inherited linear systems of algebraic equations require in many cases the inversion of dense and ill-conditioned matrices which may limit the performance of the generalized finite element methods for applications requiring large number of enrichments to be taken into account in the simulations. Needless to mention that increasing the number of enrichments in the generalized finite element methods results in a deterioration of the condition numbers in these linear systems and it may also lead to singular matrices.

Due to the ill-conditioning characteristics, iterative solvers are often avoided and instead dense direct methods are commonly used for solving the linear systems resulting from the generalized finite element methods. For instance, the Singular Value Decomposition (SVD) algorithm and Gaussian elimination techniques have been widely used in literature, see [33, 32, 13, 2, 6] among others. It is also known that the performance of these solvers depends on the number of enrichment and also the number/size of the timesteps used in the time integration of the boundary-value problem under study. A study on the performance of the preconditioned conjugate gradient method with the GFEM can be found in [15] while earlier preconditioners were also proposed for the XFEM [20, 3] and the PUFEM [9]. In all these references, iterative solvers were used for time-independent problems. But the accumulation of errors in the time domain can have a crucial effect on the convergence of iterative solvers for the generalized finite element methods. Our objective in the current study is to examine the performance of iterative solvers in generalized finite element methods for transient boundary-value problems. Although the conditioning of the system matrix may not change at different time steps but the error introduced in the time integration may expose cases that get close to the worst behaviour for which the considered iterative solver does not converge. We assess the numerical performance of the well-known Krylov subspace methods such as Generalized Minimal Residual Method (GMRES) [29] and the Changing Minimal Residual method based on the Hessenberg process (CMRH) [30]. Both GMRES and CMRH solvers do not require the storage of the full matrix in the linear system as only matrix-vector products are involved in their implementation. In addition, at each iteration in the GMRES and CMRH solvers, only evaluation of one matrix-vector product is needed and the approximate solution is obtained by solving a small least squares problem. Most of research works published in the generalized finite element methods for computational mechanics including acoustic waves and heat transfer employ dense direct solvers in their implementations. In te current study we examine the numerical performance of a class of iterative solvers for different applications in diffusion problems. We present numerical comparisons between dense direct solvers widely used in the literature for generalized finite element methods and the considered iterative solvers based on the Krylov methods. We numerically demonstrate that in the framework of generalized finite element methods, iterative solvers could be an alternative for their direct counterparts. Limited publications are available in the literature on this topic and the current study is the first to consider the GFEM solution of time-domain problems using the Krylov subspace methods.

To verify the numerical performance of the considered iterative solvers we present numerical results for two test examples on transient heat conduction. The examples are considered without applying a preconditioner. For the first test problem, the analytical solution is given in a squared domain which is used to quantify the errors of the generalized finite element solution. Effects mesh densities, number of enrichments, and size of timesteps on the performance of the linear solvers are also examined for this example. In the second test example we consider a heat conduction problem in a circular enclosure with discontinuous conduction coefficients and subject to an internal heat source. This test example is more difficult to solve than the previous example and the performance of the considered iterative solvers is examined for this case using different numbers of enrichments. Comparisons between iterative solvers and dense direct solvers are also carried out in our study for both test problems. A difficulty which we try to address in this work is the ability to split between on one hand the discretization errors caused by the generalized finite element method and the time integration scheme and on the other hand by the errors remaining from the solvers. To deal with this we assume that the SVD method is the most accurate linear solver considered in this paper and that the error achieved with SVD solver is the minimum possible error. Hence, only the discretization in space and time are dominating the computed errors. Conversely, the errors achieved with other solvers (direct or iterative) include also errors remaining from the solver itself. Hence, a contribution of the linear solver to the error is the difference between the SVD error and this solver error. In the same manner, for a given problem if a solver shows instabilities in a solution while SVD method still produces a stable solution then we will attribute the instability to the linear solver. It should be stressed that comparisons between the conventional finite element method and the GFEM have been carried out in [21] among others and it will not be repeated in this study.

The present paper is organized as follows. In section 2 we present the generalized finite element method for solving boundary-value problems. Iterative solvers for the associated linear systems are described in section 3. In section 4, we present numerical results for two test examples of boundary-value problems. The accuracy and the robustness of the Krylov subspace methods in general and the CMRH in particular when combined with the GFEM for time-domain problems, are investigated. Some concluding remarks are given in section 5.

2 Generalized finite element solution of boundary-value problems

To explain in details the formulation of the generalized finite element method, we consider the following time-dependent boundary-value problem

$$\begin{aligned}
\frac{\partial u(t,\xi)}{\partial t} - \nabla\cdot\Big(D\nabla u(t,\xi)\Big) &= f(t,\xi), \qquad (t,\xi)\in\ [0,T]\times\Omega,\\
u(t,\xi) + D\frac{\partial u(t,\xi)}{\partial \mathbf{n}} &= g(t,\xi), \qquad (t,\xi)\in\ [0,T]\times\partial\Omega, \qquad (1)\\
u(0,\xi) &= u_0(\xi), \qquad \xi\in\ \Omega,
\end{aligned}$$

where Ω is a two-dimensional spatial domain with boundary $\partial\Omega$, $t\in[0,T]$ the time interval, $\xi=(x,y)^\top$ the space coordinates, and $\mathbf{n}$ the outward unit normal on the boundary $\partial\Omega$. Here, D is the diffusion coefficient which may depend on space while $f(t,\xi)$ represents internal source/sink terms. Both the boundary function $g(t,\xi)$ and the initial function $u_0(\xi)$ are given. It should be noted that the boundary-value problem (1) has been widely used in the literature to model heat transfer, groundwater flow and reaction-diffusion among many other applications.

To solve problem (1) we first divide the time domain into equal intervals $[t_n,t_{n+1}]$ each of the duration $\Delta t=t_{n+1}-t_n$ where $n=0,1,\ldots$. The notation w^n is used to denote the value of a generic function w at time t_n. To integrate in time we use the first-order implicit Euler scheme which is unconditionally stable, so that the choice of Δt may be based only on the accuracy to be achieved in the computed solutions. This choice of a first-order scheme is easy to implement but other high-order implicit time integration schemes can also be used. However, the first-order time integration may introduce damping into the solution and, hence, affect the convergence of the iterative solvers. Therefore it is more critical to study the convergence of the solvers here rather than using a higher order time integration scheme. A semi-discrete form of the problem (1) is

$$\frac{u^{n+1}-u^n}{\Delta t} - \nabla\cdot\left(D\nabla u^{n+1}\right) = f^{n+1},$$

or simply

$$u^{n+1} - \Delta t\nabla\cdot\left(D\nabla u^{n+1}\right) = u^n + \Delta t f^{n+1}, \qquad (2)$$

To solve the problem in space we use the finite element method by establishing a variational formulation for equation (2). On the domain Ω we consider the Sobolev space of square integrable functions with existing first order derivatives $H^1(\Omega)$. We multiply (2) by a weighting function $\varphi(\xi)$, and then integrate over Ω

$$\int_\Omega\Big(\varphi(\xi)u^{n+1} - \Delta t\varphi(\xi)\nabla\cdot\left(D\nabla u^{n+1}\right)\Big)\,d\Omega = \int_\Omega\Big(\varphi(\xi)\left(u^n+\Delta t f^{n+1}\right)\Big)\,d\Omega, \qquad (3)$$

Applying the divergence theorem and substituting the boundary condition one obtains the weak formulation of the problem i.e. find $u^{n+1} \in H^1(\Omega)$ such that:

$$\int_\Omega \Big(D\Delta t \nabla\varphi \cdot \nabla u^{n+1} + \varphi u^{n+1} \Big)\, d\Omega + \oint_\Gamma \Delta t \Big(u^{n+1} - g^{n+1} \Big) \varphi \, d\Gamma = \int_\Omega \Big(u^n + \Delta t f^{n+1} \Big) \varphi \, d\Omega, \quad \varphi \in H^1(\Omega), \tag{4}$$

To solve the weak variational formulation (4) using the finite element method, we discretize the spatial domain Ω into a set of M finite elements $\mathcal{T}_i$ with the index i referring to the ith element. The combination of all these elements forms a quasi-uniform partition Ω_h with $\Omega_h \subseteq \Omega$. For any two different elements $\mathcal{T}_i$ and $\mathcal{T}_j$ of Ω_h we have

$$\mathcal{T}_i \cap \mathcal{T}_j = \begin{cases} P_{ij}, & \text{a mesh point, or} \\ \Gamma_{ij}, & \text{a common side, or} \\ \emptyset, & \text{empty set.} \end{cases}$$

The conforming finite element space for the solution that we use is defined as

$$V_h = \left\{ u_h^{n+1}(\xi) \in C^0(\Omega) : \quad u_h^{n+1}(\xi)\Big|_{\mathcal{T}_i} \in \varPsi(\mathcal{T}_i), \quad \forall\, \mathcal{T}_i \in \Omega_h \right\}, \tag{5}$$

with

$$\varPsi(\mathcal{T}_i) = \text{span}\left\{ \psi(\xi) : \quad \psi(\xi) = \widehat{\psi} \circ Y_j^{-1}(\xi), \quad \widehat{\psi} \in \varPsi_m(\widehat{\Omega}_e) \right\},$$

where $\widehat{\psi}$ is a basis function defined on the reference element while $\varPsi_m(\widehat{\Omega}_e)$ is the set of all basis functions defined on the reference element $\widehat{\Omega}_e$. Here, $Y_i(\xi) : \widehat{\Omega}_e \longrightarrow \mathcal{T}_i$ is an invertible one-to-one mapping.
Next, we formulate the generalized finite element solution to $u_h^{n+1}(\xi)$ as

$$u_h^{n+1}(\xi) = \sum_{j=1}^{N} \sum_{q=1}^{Q} U_j^{q,n+1} \Phi_j G_q(\xi). \tag{6}$$

where $\{\Phi_j\}_{j=1}^N$ are a set of polynomial functions with N being the total number of nodes in Ω_h. Notice that each of these functions is associated with one mesh node in Ω_h and it is characterized by the property $\Phi_i(\xi_j) = \delta_{ij}$ with δ_{ij} denoting the Kronecker delta. In any element $\mathcal{T}_i$ the total number of nodes is M.
In (6), $\{G_q(\xi)\}_{q=1}^Q$ are known functions that have better approximation properties compared to the polynomial functions Φ_j. The enrichment functions $\{G_q(\xi)\}_{q=1}^Q$ exist in many forms and their selections make the difference between a generalized finite element method to another. A set of functions can be chosen to enrich the finite element solution space either because they comprise the asymptotic solution space or because they own better approximation properties to the problem at hand.
For the transient boundary-value problems (2), there are two ways to enrich the finite element solution (6) depending on whether the functions $\{G_q(\xi)\}_{q=1}^Q$ depend on the time variable or not. Time-dependent enrichment functions are known by local enrichment for which a linear system has to be solved at each timestep. On the other hand, global enrichment uses time-independent functions $\{G_q(\xi)\}_{q=1}^Q$ and in this case the matrix in the associated linear system is constant which can be assembled and inverted once during the time integration process. Time-dependent enrichment functions can be found in [25, 24] among others. An example for the two-dimensional diffusion problems can be the fundamental solution used in [10]

$$G_q(\xi) = \frac{1}{4\pi\,(t-\tau)} \exp\left(\frac{-r^2}{4\,(t-\tau)} \right) H\,(t, \tau)\,,$$

where $r = \|\xi - \xi_c\|$ is the distance from the function control point $\xi_c = (x_c, y_c)^\top$ to the point $\xi = (x, y)^\top$. The time parameter τ refers to the delay time and H is the Heaviside function.

Global enrichments for steady problems include among others the use of plane waves to enrich the solution space for Helmholtz problems [17, 18] or the use of fundamental solutions for the enrichment functions [24]. Time-independent global enrichment was first introduced in [21] to solve transient diffusion problems of the form (2). This consists of a set of exponential functions defined as

$$G_q(\xi) = \frac{\exp\left(-\left(\frac{r}{C}\right)^{m_q}\right) - \exp\left(-\left(\frac{R_c}{C}\right)^{m_q}\right)}{1-\exp\left(-\left(\frac{R_c}{C}\right)^{m_q}\right)}, \qquad q = 1, 2, \dots, Q, \tag{7}$$

where the constants R_c, C and m_q control the steepness of the Gaussian function G_q depending on the boundary-value problem under study. For transient problems (2) with boundary layers, authors in [21] proposed global enrichment functions based on hyperbolic tangent functions to be applied to the boundary part of $\partial\Omega$ where high solution gradients are localized. These hyperbolic functions are defined as

$$G_q(\xi) = \frac{V_{q,1}+V_{q,2}}{2} + \frac{V_{q,1}-V_{q,2}}{2}\tanh\left(\frac{r}{R_c}\right), \qquad q = 1, 2, \dots, Q, \tag{8}$$

Again here $r = \|\xi - \xi_e\|$ is the distance from the function boundary point $\xi_e = (x_e, y_e)^\top$ to the point $\xi = (x, y)^\top$. The remaining constants $V_{q,1}$, $V_{q,2}$ and R_c are the control parameters for the steepness of the hyperbolic functions G_q depending on the boundary-value problem under study. Note that other expressions for the enrichment functions $\{G_q\}_{q=1}^{Q}$ can also be inserted in the generalized finite element approximation (6) without major modifications. The enriched finite element methods consistently show higher convergence rates compared to the standard finite element method. The convergence of such methods is studied in details in previous works (see for example [21, 24])
The set of unknowns $U_j^{q,n+1}$ are associated with the jth node and the qth enrichment function. The approximation space can then be defined as

$$\widetilde{V}_h = \text{span}\left\{\Phi_j G_q(\xi), \qquad u_h^{n+1}(\xi) = \sum_{j=1}^{N}\sum_{q=1}^{Q} U_j^{q,n+1}\Phi_j G_q(\xi)\right\}.$$

In the present work, the enrichment functions (7) introduced in [21] are adopted for the numerical tests. It is worth remarking that the enrichment functions $G_q(\xi)$ are written in terms of the global coordinates ξ, but they are multiplied by the nodal shape functions Φ_j. An elementary matrix of an element $\mathcal{T}_i$ is built of blocks $\mathcal{A}_{\mathbf{ij}}$ as

$$\begin{pmatrix} \mathcal{A}_{11} & \mathcal{A}_{12} & \dots & \mathcal{A}_{1M} \\ \mathcal{A}_{21} & \mathcal{A}_{22} & \dots & \mathcal{A}_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}_{M1} & \mathcal{A}_{M2} & \dots & \mathcal{A}_{MM} \end{pmatrix} \left\{ \begin{matrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_M \end{matrix} \right\} = \left\{ \begin{matrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_M \end{matrix} \right\}. \tag{9}$$

It should be stressed that in the GFEM, the enrichment functions are associated with element nodes. Hence, each block $\mathcal{A}_{ij}$ in (9) is associated with the nodes i and j while the corresponding blocks $\mathbf{x}_i$ and $\mathbf{b}_i$ are associated with the node i. The block sizes varies depending on the number of enrichment functions Q. A higher Q leads to a larger block size. If the considered nodes are not on the domain boundary then

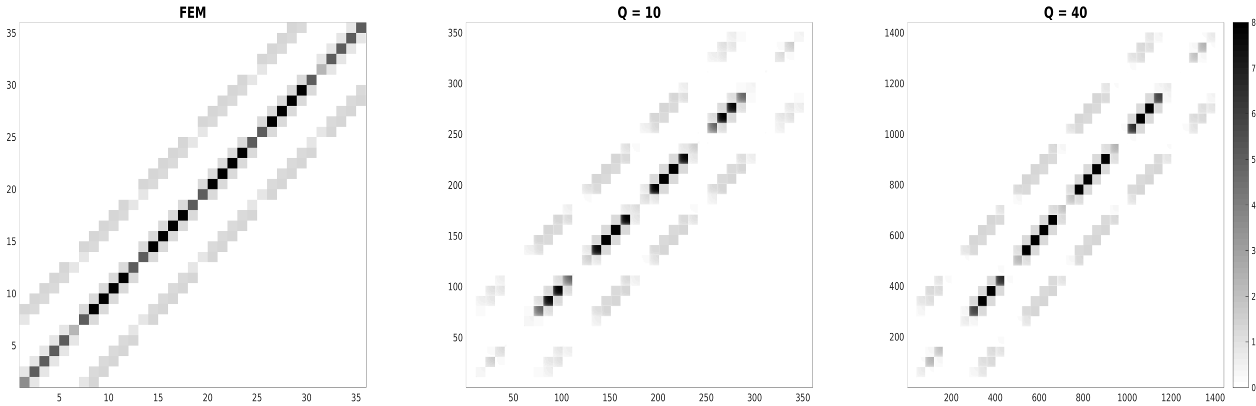


Figure 1: Illustration of the structure of matrix $\mathbf{A}$ in (10) for the standard FEM (left plot), GFEM with $Q = 10$ (middle plot) and GFEM with $Q = 40$ (right plot). Dark color in the plots refers to non-zero entries in the matrix.

individual blocks can be written as

$$\mathcal{A}_{ij} = \begin{pmatrix} a_{ij}^{11} & a_{ij}^{12} & \dots & a_{ij}^{1Q} \\ a_{ij}^{21} & a_{ij}^{22} & \dots & a_{ij}^{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{ij}^{Q1} & a_{ij}^{Q2} & \dots & a_{ij}^{QQ} \end{pmatrix}, \qquad \mathbf{x}_i = \begin{Bmatrix} U_i^{1,n+1} \\ U_i^{2,n+1} \\ \vdots \\ U_i^{Q,n+1} \end{Bmatrix}, \qquad \mathbf{b}_i = \begin{Bmatrix} \displaystyle\int_\Omega \left(\Delta t f^{n+1} + u^n\right) \Phi_i G_1 \, d\Omega \\ \displaystyle\int_\Omega \left(\Delta t f^{n+1} + u^n\right) \Phi_i G_2 \, d\Omega \\ \vdots \\ \displaystyle\int_\Omega \left(\Delta t f^{n+1} + u^n\right) \Phi_i G_Q \, d\Omega \end{Bmatrix},$$

with

$$a_{ij}^{pq} = \int_\Omega \Big(D\Delta t \nabla \left(\Phi_i G_p\right) \cdot \nabla \left(\Phi_j G_q\right) + \Phi_i G_p \Phi_j G_q \Big) \, d\Omega.$$

Assembling the elementary matrices one can obtain a linear system which can be then written as

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{10}$$

The system matrix $\mathbf{A}$ is symmetric and composed of the block matrices $\mathcal{A}_{ij}$. Similar to the standard finite element method the assembly process will overlap all the common nodes between the elements, however, in this case the overlap will include an entire block matrix $\mathcal{A}_{ij}$ rather than a single entry. Figure 1 shows an illustration of the non-zero entries of the system matrix for a standard finite element mesh compared to the same mesh enriched with 10 and 40 enrichment functions. From the figure it can be seen that the three plots show similar patterns. However, the size of the matrix changes significantly with respect to the number of enrichment functions. Needless to mention that the size of the matrix $\mathbf{A}$ increases if the size of individual block matrices $\mathcal{A}_{ij}$ is increased. For example Figure 1 shows that a single entry in the finite element matrix (left plot) becomes a 10×10 block matrix in the GFEM with $Q = 10$ (middle plot). Furthermore, it can be seen in the figure that increasing Q to 10 or 40 will maintain a similar sparsity pattern in general but with some block matrices $\mathcal{A}_{ij}$ starting to show zero rows and columns, hence, leading to a singular system. To understand this better we plot in Figure 2 the enrichment functions for different values of q. The left plot in this figure corresponds to the enrichment functions for $q = 1, 10, 20, 30, 40, 50, 60$ and 70. As we increase the values of q, the difference between one function and the next becomes smaller. To have a closer look to the right of the figure we also plot the enrichment function at $q = 39, 40$ and 41. In this plot it can be

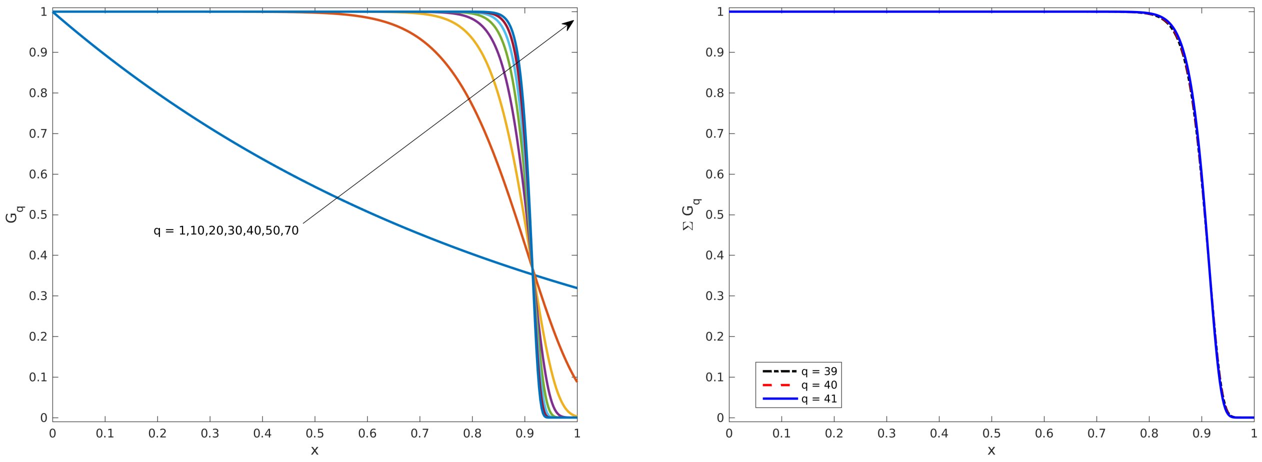


Figure 2: Enrichment functions for different values of q. The left plot shows the enrichment functions for $q = 1$, 10, 20, 30, 40, 50, 60 and 70. The right plot shows the high similarity between enrichment functions for $q = 39, 40$ and 41.

seen that the difference between these enrichment functions is negligible. Including such enrichment has the effect of enriching with the same function two or three times. This repetition of enrichments results in a singular linear system which can be seen for example for $Q = 40$ in Figure 1. As a consequence, often less than ten enrichment functions are used in the literature, see for instnace [14, 21]. However, even for a small number of enrichment functions, it can be seen in Figure 2 that within the range $0 \leq x \leq 0.4$ all enrichment functions are similar. In this case, if a finite element method falls within this range the degrees of freedom corresponding to the enrichment functions would lead to an ill-conditioned linear system or even a singular matrix in (10).

In the present study we propose using Krylov subspace methods to deal with such systems. This class of methods are based on minimizing the residual which can still be an effective strategy even when dealing with singular systems.

3 Changing minimal residual solver

In this section, we describe the Changing Minimal Residual solver based on the Hessenberg reduction (CMRH) algorithm for solving the linear system (10). The CMRH solver was first introduced by Sadok in [30] and was further developed and used in [1, 7, 31] among others. The CMRH solver is an alternative method to the classical Generalized Minimal RESidual (GMRES) algorithm developed in [28]. In [8, 11] a new implementation of CMRH was developed for solving nonsymmetric linear systems of algebraic equations with dense and large matrices as those resulted from the generalized finite element discretization (10). Comparison studies between the CMRH and GMRES methods in terms of floating point operations and memory requirements for dense matrices can be found in [11, 30, 31] among others. At this stage, we give a brief description for the CMRH method and for more details see [7, 8, 11, 31]. To formulate the CMRH solver for the linear system (10) we define the maximum and Euclidean norms of a vector $\mathbf{v} \in \mathbb{R}^N$ as

$$\|\mathbf{v}\|_\infty = \max_{1\leq i\leq N} |v_i|, \qquad \|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^{N} |v_i|^2},$$

where v_i is the ith component of the vector $\mathbf{v}$. We also use the notation $\mathbf{A}_{i:j,k:l}$ to denote the submatrix of $\mathbf{A}$ consisting of rows i to j and columns k to l. Here, $TriU(\mathbf{A})$ and $TriL(\mathbf{A})$ denote respectively, the upper and lower triangular parts of the matrix $\mathbf{A}$. We also use $Diag(\mathbf{v})$ to denote the N-valued square diagonal matrix with entries $\mathbf{v}$ on the diagonal.

Applied to the linear system (10), the CMRH solver uses an initial guess $\mathbf{y}_0$ for the exact solution to construct approximate solutions $\mathbf{y}_k$ of the form

$$\mathbf{y}_k = \mathbf{y}_0 + \mathbf{w}_k, \qquad k = 1, 2, \ldots, \tag{11}$$

where $\mathbf{w}_k \in K_k(\mathbf{A}, \mathbf{r}_0)$ and $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{y}_0$ is the initial residual, with $K_k(\mathbf{A}, \mathbf{r}_0)$ is the Krylov sub-space defined as

$$K_k(\mathbf{A}, \mathbf{r}_0) = \text{span}\left\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0\right\}.$$

Notice that the CMRH solver uses the Hessenberg process to compute a basis $\{\mathbf{l}_1, \ldots, \mathbf{l}_k\}$ of the sub-space $K_k(\mathbf{A}, \mathbf{r}_0)$ which is reconstructed such that the matrix $\mathbf{L}_k = [\mathbf{l}_1, \ldots, \mathbf{l}_k]$ is unit lower trapezoidal. The Hessenberg reduction process (with pivoting strategy) computes a unit trapezoidal matrix $\mathbf{L}_m$ by first computing

$$\mathbf{l}_1 = \frac{\mathbf{r}_0}{(\mathbf{r}_0)_{p_1}},$$

where $p_1 \in \{1, \ldots, N\}$ is selected such that $|(r_0)_{p_1}|= \|\mathbf{r}_0\|_\infty$. Then, at each iteration k, it proceeds by computing $\mathbf{l}_{k+1}$ which satisfies

$$h_{k+1,k}\mathbf{l}_{k+1} = \mathbf{A}\mathbf{l}_k - \sum_{j=1}^{k} h_{j,k}\mathbf{l}_j, \qquad k = 1, \ldots, m,$$

with the parameters $h_{j,k}$ are determined such that

$$\mathbf{l}_{k+1} \perp \mathbf{e}_{p_1}, \ldots, \mathbf{e}_{p_k} \qquad \text{and} \qquad (\mathbf{l}_{k+1})_{p_{k+1}} = 1,$$

where $p_j = 1, \ldots, N$ and $\mathbf{e}_i$ is the ith vector of the canonical basis. Next let $\mathbf{L}_k$ and $\overline{\mathbf{H}}_k$ be the matrices generated by the Hessenberg process and let $\mathbf{H}_k$ be the matrix obtained from $\overline{\mathbf{H}}_k$ by deleting its last row. The correction $\mathbf{w}_k$ given in (11) can be written in the form $\mathbf{w}_k = \mathbf{L}_k\mathbf{d}_k$, where $\mathbf{d}_k \in \mathbb{R}^k$ is the solution of the following least squares problem

$$\min_{d \in \mathbb{R}^{k+1}} \left\|\beta\mathbf{e}_1 - \overline{\mathbf{H}}_k\mathbf{d}\right\|_2,$$

where $\beta = (\mathbf{r}_0)_{p_1}$. Hence, the implementation of the CMRH solver for the linear system (10) can be carried out using the following algorithm:

Algorithm: CMRH Method

```
Inputs : $\mathbf{A} \in \mathbb{R}^{N\times N}, \mathbf{b} \in \mathbb{R}^N, \mathbf{x} \in \mathbb{R}^N$ an initial guess, $tol$ a tolerance;
Output: $\mathbf{x}_k$ the $k$h approximate solution. ($\mathbf{x}_k$ is stored in $\mathbf{b}$);
Compute $\mathbf{b} = \mathbf{b} - \mathbf{Ax}$;
% Hessenberg process;
Let $\mathbf{p} = [1, \dots, N]^T$;
Determine $i_0$ such that $|b_{i_0}| = ||\mathbf{b}||_\infty$;
$\beta = b_{i_0}$, $\mathbf{b} = \mathbf{b}/\beta$;
$p_1 \leftrightarrow p_{i_0}$, $b_1 \leftrightarrow b_{i_0}$, $\mathbf{A}_{1,:} \leftrightarrow \mathbf{A}_{i_0,:}$, $\mathbf{A}_{:,1} \leftrightarrow \mathbf{A}_{:,i_0}$;
for $k = 1, \dots,$ until convergence do
    $\mathbf{u} = \mathbf{A}_{:,\mathbf{k}} + \mathbf{A}_{:,\mathbf{k+1:N}} \mathbf{b}_{\mathbf{k+1:N}}$;
    $\mathbf{A}_{k+1:N,k} = \mathbf{b}_{k+1:N}$;
    for $j = 1, \dots, k$ do
        $\mathbf{A}_{j,k} = \mathbf{u}_j$, $\mathbf{u}_j = 0$, $\mathbf{u}_{j+1:N} = \mathbf{u}_{j+1:N} - \mathbf{A}_{j,k}\mathbf{A}_{j+1:N,j}$;
    end
    Determine $i_0 \in \{k+1, \dots, N\}$ such that $|u_{p_{i_0}}| = ||\mathbf{u}_{p_{k+1}:p_N}||_\infty$;
    $h = u_{p_{i_0}}$, $\mathbf{b} = \mathbf{u}/h$;
    $p_{k+1} \leftrightarrow p_{i_0}$, $b_{k+1} \leftrightarrow b_{i_0}$;
    $\mathbf{A}_{k+1,:} \leftrightarrow \mathbf{A}_{i_0,:}$, $\mathbf{A}_{:,k+1} \leftrightarrow \mathbf{A}_{:,i_0}$;
    % More details on the next two steps can be found in [11];
    Update the $QR$ factorization of $\overline{\mathbf{H}}_k$ ;
    Apply previous rotations to $\overline{\mathbf{H}}_k$ and $\beta\mathbf{e}_1$;
end
Solve $\mathbf{d} = \beta\mathbf{e}_1$; % ($\mathbf{H} = \mathbf{H}_k = TriU(\mathbf{A}_{1:k,1:k})$);
Update  $\mathbf{x} = \mathbf{x} + \mathbf{Ld}$; %($\mathbf{L} = \mathbf{L}_k = Diag(ones(k,1)) + TriL(\mathbf{A}_{:,1:k}, -1)$);
% Reorder the components of $\mathbf{x}$;
for i=1,...,N do
    $b_{p_i} = x_i$;
end
```

Note that we have used the symbol ↔ to express swapping contents in the CMRH algorithm. For more details on over-storage and pivoting strategy in the CMRH solver, see [30, 11, 7, 1, 8].

4 Numerical results

In this section we examine the accuracy and performance of the considered linear solvers for the generalized finite element solution of two transient boundary-value problems. The first problem was studied previously in [21] but here we consider higher numbers of enrichment functions. The aim is to investigate the severely ill-conditioned systems or even singular systems resulting from the GFEM.

In the presented results the errors of the GFEM solutions obtained using the SVD method are considered as the baseline errors. For the comparison purpose, these errors are assumed to be caused by the finite element and the time integration approximations rather than the solution of the linear system. This assumption is based on the wide usage in the literature of the SVD solver to deal with highly ill-conditioned linear systems similar to the ones considered in the current study. For the considered iterative solvers and in all the following problems the iterations are terminated when the residual is less than 10^{-10}, which is small enough to guarantee that the algorithm truncation error dominated the total numerical error.

In the first example the GFEM is used to solve a transient diffusion problem given by the equations (1) with a known analytical solution [21]. The error in the boundary-value solution is evaluated using the analytical solution. Other direct solvers based on canonical Gaussian elimination are also considered. The aim in this

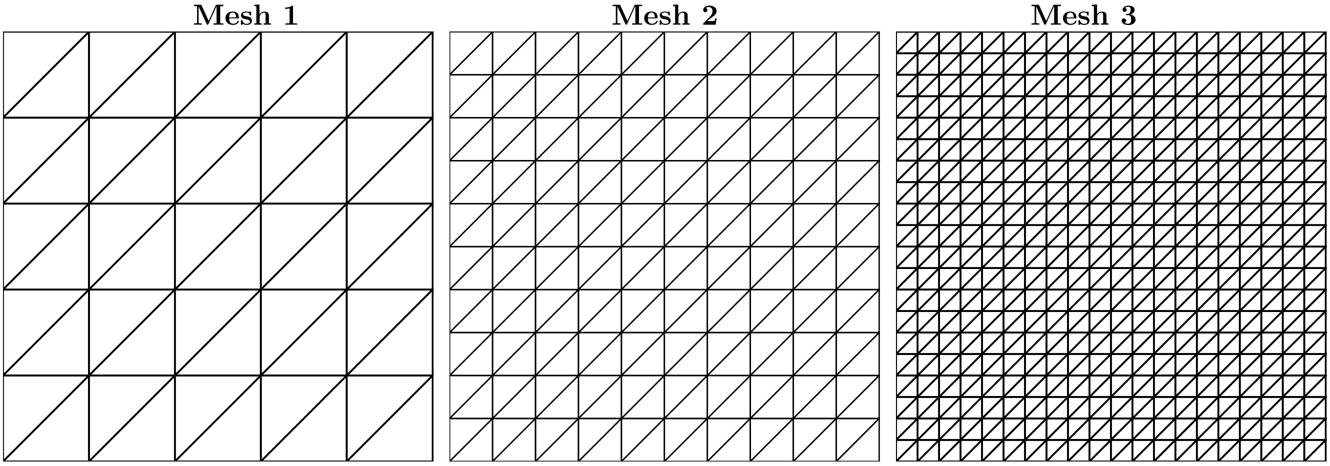


Figure 3: Structured meshes with different element densities used for Example 1.

example is to assess the limits of such dense direct solvers as well as the convergence in the iterative solvers and the resulting accuracy in the generalized finite element solution. With respect to iterative solvers we also show the number of iterations required to achieve the requested residual and how this error is reflected in the solution accuracy of the boundary-value problem. In the second example, we consider the problem of heat transfer in a composite material with an internal heat source. This is a complex physical problem which can be used to test the limits of the considered iterative solvers. The performance of both the CMRH and GMRES for the generalized finite element solution of time-dependent problems using high number of enrichments is computationally assessed. In both examples a set of exponential functions similar to (7) is used for the enrichment.

4.1 Example 1

As a first test example we consider a transient diffusion problem in a squared domain. This example has also been considered in [21, 14] as a benchmark problem for validating the generalized finite element methods. We solve the equations (1) in $\Omega = [0, 2] \times [0, 2]$ with the reaction term $f(t, \xi)$, the boundary function $g(t, \xi)$ and the initial condition $u_0(\xi)$ all being explicitly calculated such that the exact solution of the problem (1) is given by

$$U(t, x, y) = (2x - x^2)^{\frac{1}{5D}} (2y - y^2)^{\frac{1}{5D}} \left(1 - e^{-Dt}\right), \tag{12}$$

where the diffusion coefficient $D = 0.01$. To quantify the errors in this test example we consider the L^2-error defined as

$$\epsilon_2(t) = \left\| u_h^n - U(t_n, x_h, y_h) \right\|_{L^2(\Omega)}, \tag{13}$$

where $\|\cdot\|_{L^2(\Omega)}$ is the L^2 norm, u and U are, respectively, the computed and exact solutions. We consider the three meshes shown in Figure 3 to examine the grid dependence of the results for the considered linear solvers in the GFEM. The meshes consist of 50 elements and 36 nodes in Mesh 1, 200 elements and 121 nodes in Mesh 2, and 800 elements and 441 nodes in Mesh 3. It is worth mentioning that the numbers of enrichment functions used here are deliberately higher than the numbers considered in the original example in [21]. The higher numbers of enrichment leads to severely ill-conditioned systems. This behavior with the GFEM is known to have serious consequences on the choice of the linear solver. In this work we concern ourselves with studying this choice rather than studying the convergence of the GFEM which was previously studied in different places including [21].

In Table 1 we summarize the L^2-error obtained for different numbers of enrichment using Mesh 1 and a fixed timestep $\Delta t = 0.1$. The iterative solvers are compared to direct solvers, namely, the SVD, the

Table 1: L^2-errors in the generalized finite element solution of Example 1 on Mesh 1 with $\Delta t = 0.1$ using different linear solvers and different number of enrichments Q. Here, the errors are measured after running the simulation for 1, 5 and 10 timesteps.

		Direct solver				Iterative solver	
Q	# timesteps	SVD	LDL$^\top$	GaussP	Gauss	GMRES	CMRH
14	1	2.26E-06	2.62E-06	2.27E-06	2.31E-06	2.68E-06	2.70E-06
	5	1.23E-05	1.26E-05	1.25E-05	1.24E-05	1.24E-05	1.42E-05
	10	2.57E-05	2.64E-05	2.64E-05	2.59E-05	2.58E-05	2.62E-05
16	1	2.28E-06	3.04E-06	2.31E-06	—	2.50E-06	2.83E-06
	5	1.24E-05	1.65E-04	1.28E-05	—	1.26E-05	1.35E-05
	10	2.59E-05	1.04E-02	2.69E-05	—	2.59E-05	2.72E-05
18	1	2.31E-06	—	—	—	2.52E-06	2.59E-06
	5	1.25E-05	—	—	—	1.26E-05	1.35E-05
	10	2.61E-05	—	—	—	2.60E-05	2.61E-05

Gauss elimination (Gauss), the Gauss elimination with pivoting and scaling (GaussP) [16] and Cholesky decomposition (LDL$^\top$) [5]. The results of direct and iterative solvers are presented for $Q = 14, 16$ and 18 at three different instants. For the lowest number of enrichment functions $Q = 14$, it is clear that the direct solvers Gauss, GaussP and LDL$^\top$ start with slightly smaller errors at the first timestep compared to the iterative solvers. Thereafter, these direct solvers seem to accumulate errors faster at subsequent timesteps compared to the SVD solver.

By increasing the number of enrichments to $Q = 16$ the direct solver Gauss becomes unstable (— in Table 1 corresponds to runs where the Gauss, GaussP and LDL$^\top$ solvers produce indefinite numbers). The solution error with LDL$^\top$ at the first timestep is similar to those with other solvers. However, this error becomes one order of magnitude larger after 5 timesteps and three orders of magnitude larger after 10 timesteps. At this number of enrichment the GaussP solver seems to still produce reasonable results. It can also be noted in Table 1 that, at the last set of results corresponding to $Q = 18$, the three direct solvers Gauss, GaussP and LDL$^\top$ cease to produce numerical results. On the other hand the iterative solvers GMRES and CMRH show consistent results to those obtained using the direct solver SVD for all considered numbers of enrichments. In general the errors obtained using the CMRH solver are slightly higher than those obtained using the GMRES method while the latter errors are slightly higher than the ones obtained using the SVD method. The number of iteration with both GMRES and CMRH increases as we add more enrichment functions. At the first time step GMRES converges into the considered tolerance after 352 iterations for $Q = 14$ while at the same time step GMRES requires 393 iterations for $Q = 18$. The corresponding numbers of iterations with CMRH are 336 and 384, which again show an increase.

The iterative solvers alongside with the SVD method are capable of producing useful results for all the considered enrichment numbers in Table 1 even when the direct solvers Gauss, GaussP and LDL$^\top$ are not. This is found to be the case despite the fact that only moderately high enrichment numbers with a relatively coarse mesh are considered in Table 1. Refining the mesh and/or increasing the number of enrichment functions leads to even worse conditioned linear systems. This suggests that the canonical Gauss elimination-based solvers have serious limitations in dealing with such systems. This is a particularly important observation as the usage of solvers based on canonical Gaussian elimination are widely used in the

Table 2: L^2-errors in the generalized finite element solution of Example 1 on Mesh 2 using SVD, GMRES and CMRH solvers for high numbers of enrichments Q and different timesteps Δt. Here, the errors are measured after running the simulation for 1, 5 and 10 timesteps.

				CMRH		GMRES		SVD
Q	Δt	# sv	# timesteps	# itr	L^2-error	# itr	L^2-error	L^2-error
	0.5	11	1	836	3.62E-06	943	3.35E-06	3.25E-06
20	0.1	11	5	856	2.06E-06	950	1.67E-06	1.47E-06
	0.05	11	10	856	1.97E-06	957	1.57E-06	1.36E-06
	0.5	25	1	928	3.70E-06	1061	3.35E-06	3.25E-06
30	0.1	25	5	934	2.29E-06	1066	1.67E-06	1.46E-06
	0.05	25	10	936	2.19E-06	1068	1.58E-06	1.37E-06
	0.5	61	1	999	3.61E-06	1130	3.35E-06	3.25E-06
40	0.1	61	5	998	2.27E-06	1138	1.66E-06	1.45E-06
	0.05	60	10	1021	2.22E-06	1144	1.54E-06	1.36E-06

literature [33, 32, 13, 2] to deal with linear systems generated by the GFEM. The usage of SVD is limited to few examples due to the high computational cost needed for its implementation. This computational cost becomes prohibitive for very large linear systems or when solving problems in the time domain. Notice that a time-dependent linear system has to be solved repeatedly and usually for a large number of timesteps. Hence, in such cases it is necessary to find an alternative method to the direct SVD method. All the remaining computations in this paper are performed only using GMRES and CMRH solvers. The solutions obtained with the iterative solvers are compared to those obtained using the SVD method.

To quantify the accuracy of the iterative solvers with respect to the size of the linear system of algebraic equations we consider a first mesh refinement as well as higher number of enrichment functions. Table 2 summarizes the L^2-errors for the considered linear solvers using Mesh 2 along with Q = 20, 30 and 40. The table also includes the number of singular values (# sv) resulting from the SVD solver of each of the matrices that are associated with different timesteps Δt used in the simulations. It is clear from Table 2 that all matrices in the linear system are singular for the considered values of Q and Δt. Hence, the value # sv obtained with SVD, is used to indicate the quality of the linear systems instead of the condition number which is infinitely large for such systems. It should be stressed that the considered iterative solvers are based on minimizing the residual. Such a strategy can still be effective even when the linear system is singular.

As in the previous comparison a consistent accuracy is achieved using CMRH, GMRES and SVD solvers. Again the errors obtained using CMRH and GMRES solvers are slightly higher than those obtained using the SVD method for all the considered values of Q and Δt. It is also evident that increasing the number of enrichment Q results in an increase in # sv, thus a higher order singularity is encountered. This also results in an increase in the number of iterations (# itr) required in GMRES and CMRH solvers to achieve the fixed tolerance of 10^{-10}. However, for any given Q and Δt, the CMRH solver requires fewer iterations than the GMRES solver to reach the same tolerance. Hence, for a relatively similar accuracy in the generalized finite element solution, the convergence in the CMRH solver is attained with at least 10% less iterations than those required for the convergence in the GMRES solver. Needless to mention that this faster convergence with the CMRH solver ensures a proportional saving in the computational cost for the CMRH solver compared

Table 3: L^2-errors in the generalized finite element solution of Example 1 on Mesh 2 at different instants using SVD, GMRES and CMRH methods.

		CMRH		GMRES		SVD
Time	Δt	# itr	L^2-error	# itr	L^2-error	L^2-error
t = 1	0.5	872	6.81E-06	980	6.34E-06	6.22E-06
	0.1	869	4.06E-06	983	3.21E-06	3.05E-06
	0.05	871	3.93E-06	988	3.07E-06	2.93E-06
t = 2.5	0.5	869	1.56E-05	1041	1.42E-05	1.42E-05
	0.1	948	9.68E-06	1042	8.14E-06	8.14E-06
	0.05	948	9.65E-06	1043	7.80E-06	7.90E-06
t = 5	0.5	992	2.61E-05	1078	2.55E-05	2.56E-05
	0.1	982	1.77E-05	1080	1.66E-05	1.64E-05
	0.05	986	1.71E-05	1081	1.61E-05	1.66E-05

to the GMRES solver. The computational cost in the direct SVD solver is far too large to be compared to the GMRES and CMRH solvers where the number of iterations is still much smaller than the size of the linear system.

As mentioned above, increasing the number of enrichment functions clearly affects the size and the structure of the associated linear system. A higher values of Q leads to a higher # sv and hence, a more sensitive linear system. To check the stability of the linear solvers, Table 2 also shows the results obtained at the same instant $t = 0.5$ but achieved with different timesteps Δt. For each value of Q, refining the timestep Δt leads to better L^2-errors with all solvers. For example using the CMRH method, refining Δt from 0.1 to 0.05 at $Q = 30$, reduces the L^2-error from 2.29×10^{-6} to 2.19×10^{-6}. However, when increasing Q and for a given timestep the L^2-error does not improve or even slightly increases with all the linear solvers. For example using the CMRH solver, increasing Q from 30 to 40 at $\Delta t = 0.05$, leads to a slight increase in the L^2-error from 2.19×10^{-6} to 2.22×10^{-6}. This suggests that the temporal error is dominating the accuracy rather than the spatial one. Note that adding more enrichment functions in the finite element solution improves the spatial resolution rather than the temporal error. Hence, in this case adding more enrichment functions does not improve the L^2-error but it only accumulates round-off errors.

Next, we investigate the stability of the considered iterative solvers to the accumulation of L^2-errors over longer time spans. The study is performed using Mesh 2 and a number of enrichment functions $Q = 20$ for the same three timesteps $\Delta t = 0.5, 0.1$ and 0.05 considered previously in Table 2. However, now the problem is considered over longer time spans where the results are shown in Table 3. Again consistent accuracy is observed for all the solvers with the SVD method leading to the most accurate results in comparison to GMRES and CMRH solvers. When using CMRH and GMRES solvers for all considered instants, a smaller Δt consistently improves the L^2-error. This suggests that the iterative solvers do not display instability over large number of steps even for the longer time spans considered. Table 3 also lists the number of iterations # itr required in CMRH and GMRES solvers to achieve the fixed tolerance. It is also evident that the CMRH solver requires less number of iterations compared to the GMRES solver while the later achieves better errors than the former. For the considered boundary-value problem, the CMRH solver leads to about 10% reduction in # itr compared to the GMRES solver for the same required tolerance to stop the iterations.

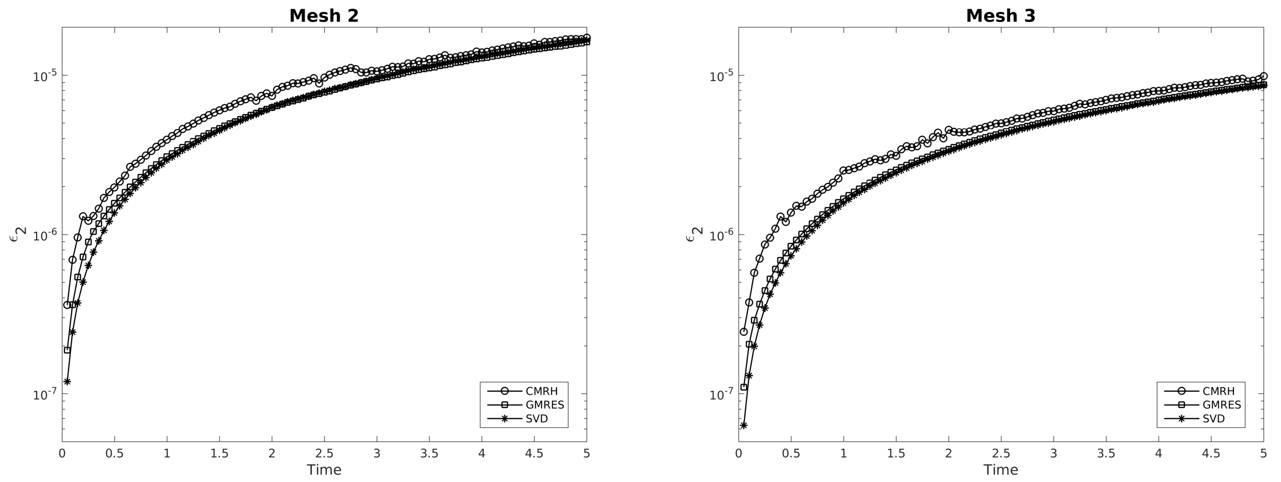


Figure 4: Time evolution of the L^2-error ϵ_2 for Example 1 on Mesh 2 (left plot) and Mesh 3 (right plot) with $\Delta t = 0.05$ and $Q = 20$ using SVD, GMRES and CMRH methods.

Our final concern with this example is to test the sensitivity of the iterative solvers to the mesh refinements. To this end structured Mesh 2 is considered along side with a further refinement namely, Mesh 3 shown in Figure 3. For both meshes, the number of enrichment functions is set to $Q = 20$. To minimize the temporal errors in the computed solutions, the finest timestep $\Delta t = 0.05$ is considered for this set of results. Figure 4 shows the time evolution of the L^2-errors for the three solvers CMRH, GMRES and SVD on the two meshes. As in the previous simulations, the obtained errors exhibit a consistent behavior for all considered linear solvers. The SVD method produces the minimum errors with slightly higher errors obtained with GMRES and CMRH solvers. As expected large L^2-errors are obtained on the coarse mesh Mesh 2 while refining the mesh improves the L^2-errors on Mesh 3. This behavior is observed in all the three solvers included in Figure 4. The time evolution of the L^2-errors shows a consistent accumulation of the solution error independently from the linear solver.

To further understand the effect of mesh refinement on the convergence of the considered iterative solvers, Figure 5 reports the number of iterations in CMRH and GMRES solvers relevant to the L^2-errors shown in Figure 4. The figure clearly confirms that the CMRH solver requires fewer iterations than the GMRES solver for all the three meshes used in our simulations. In fact refining the mesh seems to increase the gap between the two iterative solvers. On average the CMRH solver converges with 10% less iterations than the GMRES solver on Mesh 2. This number of iterations required for convergence increases to about 15% for the same simulations on Mesh 3. In addition, to have a better look into the convergence rate of CMRH and GMRES solvers at different instants, Figure 6 shows plots of the residual as it evolves iteratively with CMRH and GMRES solvers. The plots are presented at three different instants again using $\Delta t = 0.05$, $Q = 20$ and for Mesh 2 and Mesh 3. It is clear that there is a difference between the convergence plots for the GMRES solver and the CMRH solver. The former converges faster than the GMRES solver for all considered instants. The convergence in the CMRH solver becomes faster than in the GMRES solver as the mesh becomes fine. This suggests better efficiency in the CMRH solver with larger linear systems. Obviously, refining the mesh in the generalized finite element method results into larger linear systems which would then increase the number of iterations required for convergence in the iterative solvers. These features can be clearly observed in Figure 6.

4.2 Example 2

In this example we solve a heat equation of the form (1) with discontinuities in the conductivity coefficient D. Similar test example has been considered in [6]. The circular domain is centered at $(1, 1)^\top$ with unit

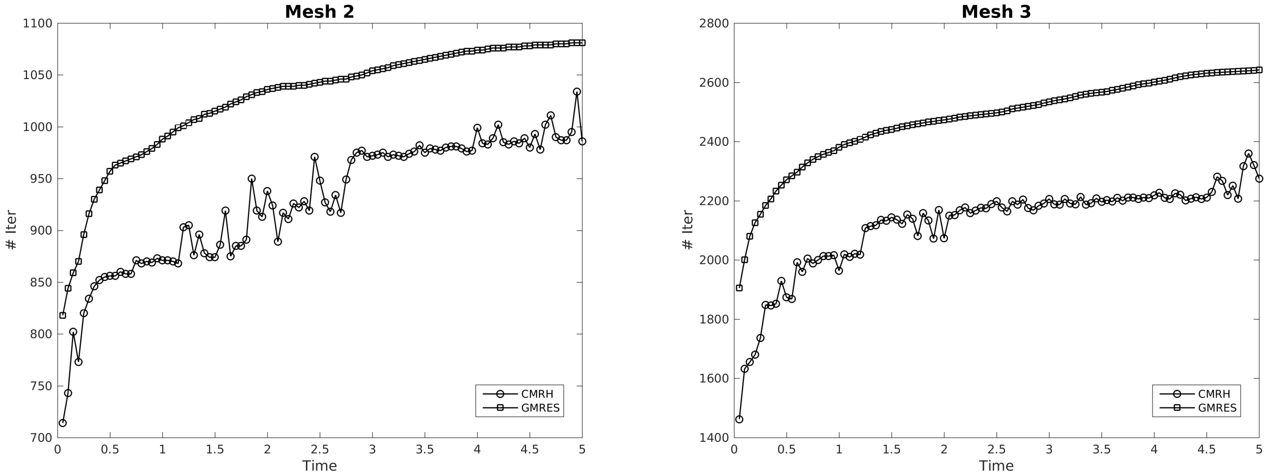


Figure 5: Time evolution of the number of iterations in GMRES and CMRH methods for Example 1 on Mesh 2 (left plot) and Mesh 3 (right plot) with $\Delta t = 0.05$ and $Q = 20$.

radius formed by two materials with different heat conduction properties. Here, the solution u in the boundary-value problem (1) refers to the temperature media, D the conduction coefficient, $f(t, \mathbf{x})$ the thermal source, $g(t, \mathbf{x})$ the ambient temperature, and $u_0(\mathbf{x})$ the initial temperature. In our simulations a constant ambient temperature 300 is considered which is also assumed to be the domain temperature at the start of the simulation. The conduction coefficient of the composite enclosure is given as

$$D(\mathbf{x}) = \begin{cases} 100, & \text{if} \quad \mathbf{x} \in \Omega_s, \\ \\ 0.1, & \text{elsewhere}, \end{cases}$$

where Ω_s is a circular sub-domain centered at $(1.25, 1.25)^\top$ with radius 0.25 as shown in Figure 7. Heat energy is introduced into the domain with two different releases where the heat source is defined as

$$f(t, \mathbf{x}) = \begin{cases} 1500, & \text{if} \quad \mathbf{x} \in \Omega_s, \\ \\ 300, & \text{elsewhere}. \end{cases}$$

In this test example we extend the range of problems considered to include more complex components compared to the first example. The physical model includes discontinuous coefficients in the boundary-value problem due to simulating isotropic composite materials. Having steep discontinuities in the coefficients of the boundary value problem, which is the case here, can have an effect on the conditioning of the associated linear systems. Furthermore, 6-noded quadratic elements are used to represent the geometry compared to the 3-noded linear elements used in the previous example. Using quadratic elements is usually preferred in the GFEM where the field solution is recovered with the enrichment. Hence, the element size is only limited to representing the geometry. Quadratic elements are much more flexible than linear elements when meshing curved geometries. This flexibility in many cases leads to oddly-shaped elements or combinations of large and small elements. This can be seen for example in the GFEM mesh used for this problem shown in Figure 7. Such meshes also affect the conditioning of the resulting linear system in the GFEM. The example is used to test the sensitivity of iterative solvers to these challenges. It should also be noted that the problem geometry can only be meshed with an unstructured mesh. Hence, the system matrix to be solved now is of a non-uniform sparsity pattern and the iterative solvers can be tested for unstructured mesh computations. To investigate the limits of the iterative solvers when dealing with severely ill-conditioned systems of a non-uniform sparsity pattern, the problem will be solved for a high number of enrichment

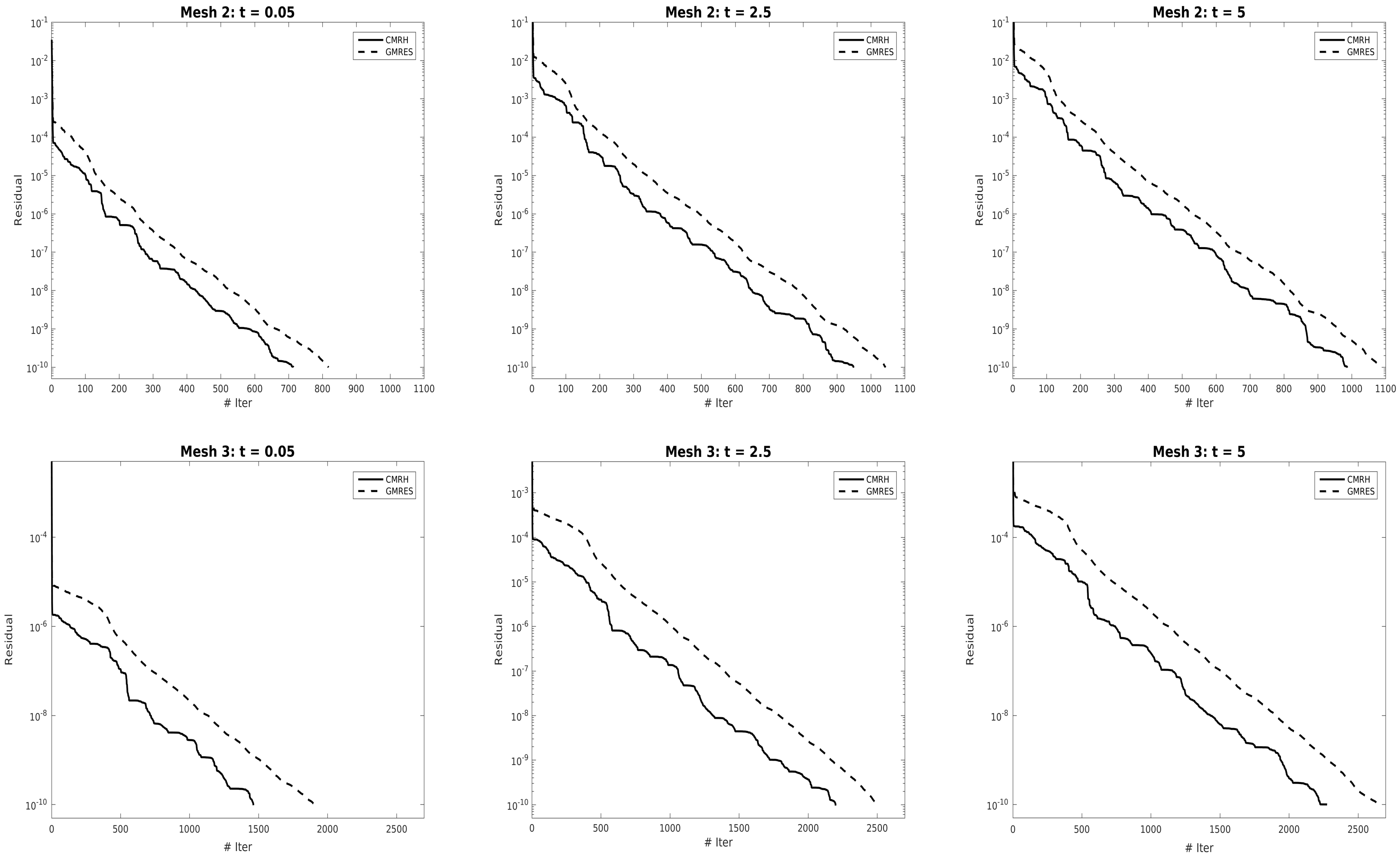
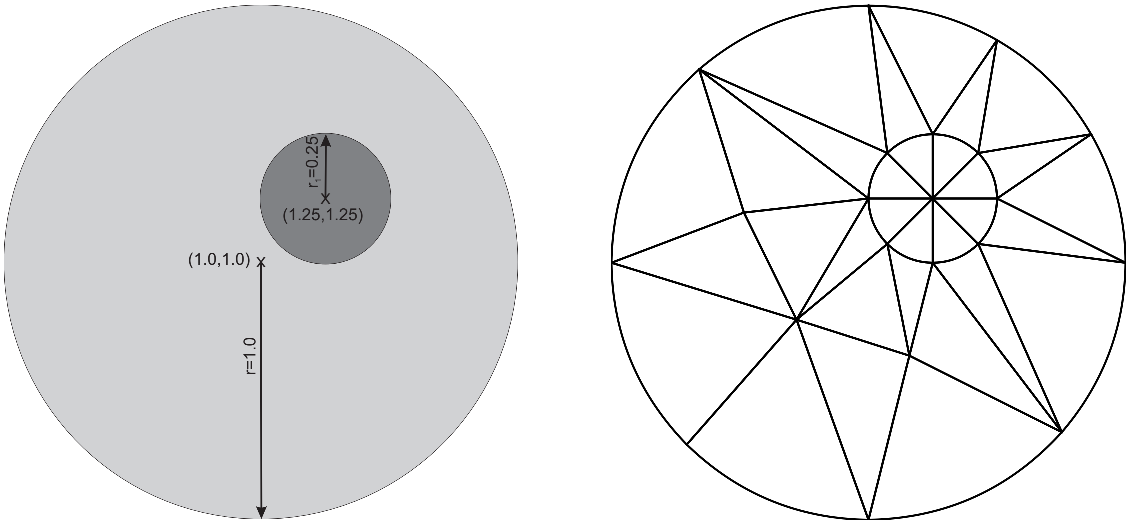


Figure 6: Convergence results in the residual for Example 1 on Mesh 2 (first row) and Mesh 3 (second row) with $\Delta t = 0.05$ and $Q = 20$ using GMRES and CMRH methods at $t = 0.05$ (first column), $t = 2.5$ (second column) and $t = 5$ (third column).



Geometry

31 elements, 72 nodes

Figure 7: Problem layout and the GFEM mesh used for Example 2.

functions. In addition to check the effect of the time integration scheme on the stability of the solution, a relatively large timestep $\Delta t = 0.1$ is chosen.

Because the problem cannot be solved exactly and in order to ensure the convergence of the GFEM, the problem is first solved for $Q = 5, 10, 15$ and 20 enrichment functions where the SVD method is used to solve the associated linear systems. It was confirmed that the GFEM has converged where the same solution is obtained with $Q = 10, 15$ and 20. Hence, the converged solution with $Q = 20$ is taken as a reference for further GFEM computations of this test example. The problem is then solved again for $Q = 50$ and 70. The GFEM solution for $Q = 50$ and 70 is verified in two ways. First it is compared to the reference solution. Then the GFEM results are critically evaluated based on how realistic and consistent the obtained heat diffusion patterns are. This is an important evaluation as different linear solvers may converge to different solutions due to the conditioning issues in the linear system. However, it is not possible that the iterative solver will converge to the wrong solution that is consistent with the physical problem.

It was observed for $Q = 50$ and in the first timestep that both GMRES and CMRH solvers have converged into the target tolerance of 10^{-10}. The solution obtained using the GMRES solver shows irregular heat patterns which are not consistent with the reference solution. However, the heat minimum and maximum magnitudes remain comparable to those obtained with the reference solution. On the other hand the CMRH and SVD solutions lead to similar solutions that match the reference solution. As the thermal fronts progress in time and beginning from the second timestep the GMRES solution which is still irregular starts to show heat magnitudes of few orders larger than those of the reference solution. On the other hand, the CMRH and SVD solution remains consistent with the reference solution up to $t = 2$ where the CMRH solution shows relatively small irregularities in the heat patterns and then by $t = 2.5$ the CMRH solution generates irregular patterns but still of a maximum and minimum magnitudes of the same order as the reference solution. The SVD solution stays in general consistent with the reference solution but at $t = 2.5$ starts to show small irregularities.

To further investigate this problem the number of enrichment functions is increased to $Q = 70$. Although the GMRES solver converges to the prescribed tolerance but the computed GFEM solution is clearly nonphysical and does not match the reference solution. The GMRES solution and starting from the first timestep exhibits non-physical oscillations with irregular heat patterns. The heat magnitudes are also unrealistically high. The solutions obtained using CMRH and SVD solvers remain consistent with the reference solution. The temperature distributions obtained using CMRH, GMRES and SVD solvers are depicted in Figure 8 for three different instants t = 1, 1.5 and 2. It can clearly be seen the irregular patterns of the solution obtained with GMRES solver. It is clear from the figure that the temperature

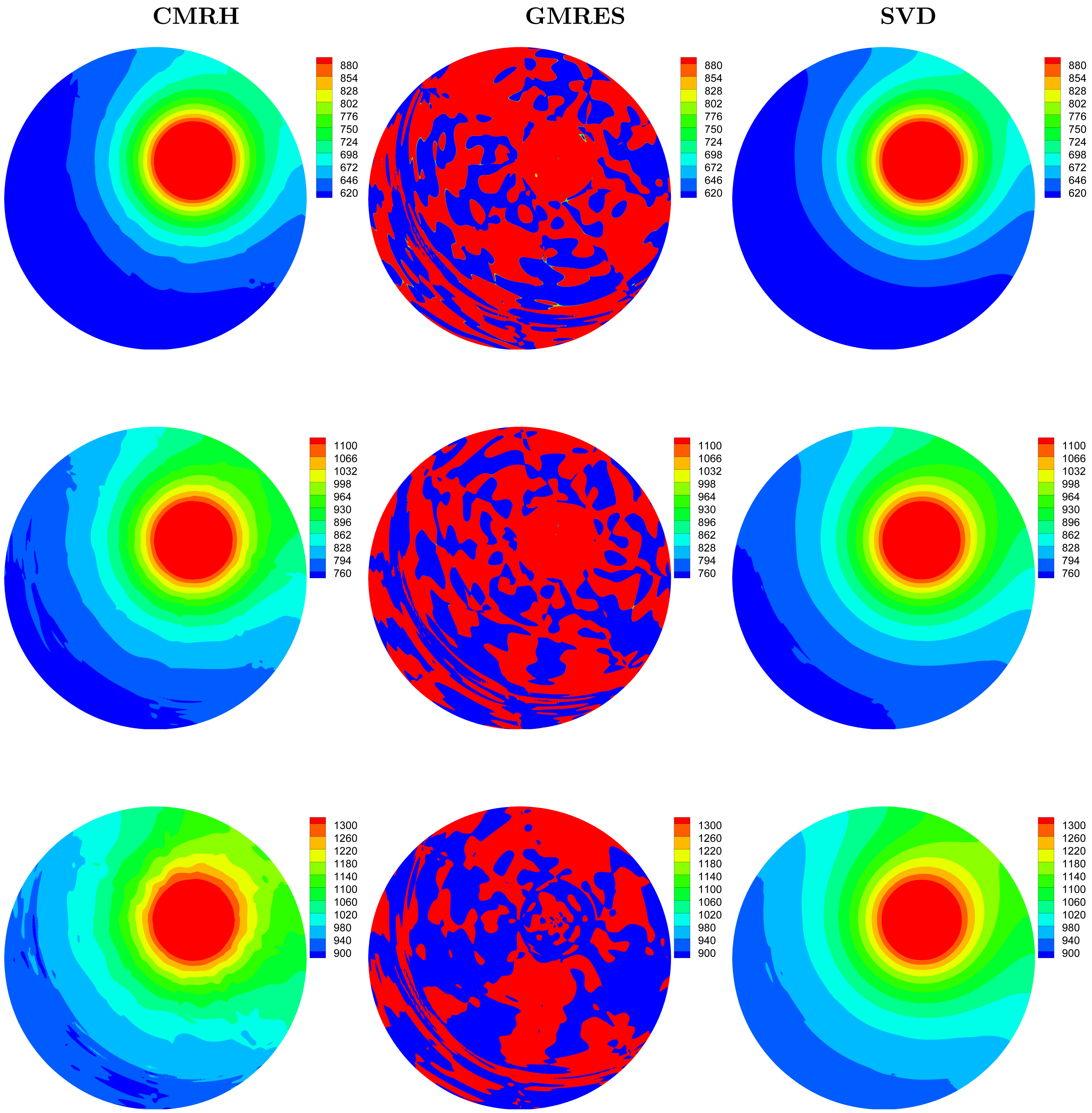


Figure 8: Temperature distributions obtained for Example 2 using the CMRH solver (first column), the GMRES solver (second column) and the SVD solver (third colum) for three different instants $t = 1$ (first row), $t = 1.5$ (second row) and $t = 2$ (third row).

computed using the CMRH solver covers all the heat features captured by the direct SVD solver. Indeed, the localized heat gradients and the moving thermal fronts are well resolved using the CMRH solver for the three selected instants. The figure also shows that CMRH produces some irregularities in the solution which builds up with time. To a lesser extent such irregularities are also observed with the SVD solution. The irregularities observed with different solvers can be attributed to the fact that for the considered number of enrichments $Q = 70$, the linear systems to be solved in the GFEM solver are relatively dense and severely ill-conditioned. However, this deterioration in the accuracy of the computed solution is not only attributed to the linear solver but mainly because the system matrix does not change within the time and only the right-hand side changes at each time step. Therefore, the deterioration in the different linear solvers is not due to the worsening of the matrix conditioning but it is rather due to falling into cases which get closer to the worst behaviour *i.e.*, cases which expose the ill-conditioning.Moreover, because the considered timestep is relatively large, the error introduced by the time integration scheme is also significant. This can also be seen in the figure where the results show a build-up in the error at the different time instances with CMRH as well as with SVD. Obviously, the chosen timestep contribute to the instability of the GMRES solution in this case.

When using iterative solvers in the GFEM and as was observed in this example it is possible that the iterative solver converges to a wrong solution. In this case, both CMRH and GMRES solvers have converged to the same tolerance but the GMRES solution is clearly physically wrong as it shows a random heat distribution which is not consistent with the heat source. On the other hand the results obtained using CMRH and SVD methods show similar heat patterns which are consistent with the heat source and the discontinuity in the composite enclosure. This suggests that the CMRH solver is more stable in this case than the GMRES method. For the considered thermal conditions, the computational work needed to perform one timestep in the generalized finite element using the direct SVD solver is several times longer than using its CMRH counterpart. Finally, it is also important to note that despite the ill-conditioning of linear systems, the convergence of the GFEM solution is still ensured for $Q = 50$ and 70. A similar heat conduction problem has been solved in [6] using the partition of unity method but using a low number of enrichments compared to the ones considered here. The results reported in [6] have also revealed that the standard finite element method requires far large number of degrees of freedom for a comparable accuracy in the GFEM for similar heat conduction problem.

5 Concluding remarks

The performance of a class of iterative solvers based on Krylov sub-spaces has been assessed for the solution of linear systems of algebraic equations in the generalized finite element solution of time-dependent boundary-value problems. The enrichment consists of embedding a hierarchy of exponential functions in the finite element shape functions which tend to accurately approximate internal boundary layers in the problem under study. The numerical advantage of this approach is related to the selection of enrichment functions which mimic the spatial behaviour of the solution at different time phases such that each function represent a different phase. However, these functions are time independent and the integration in time is still achieved following the conventional methods. These approximation properties can lead to a significant saving in the computational costs compared to only spatial approximation enrichments that change at each timestep. In general, the generalized finite element method shows higher accuracy than the conventional finite element method for a fixed number of degrees of freedom. The results obtained showed that the generalized finite element method has the advantage of requiring less computational resources for the time-dependent diffusion problems than a conventional finite element method, typical of those widely used in the finite element solution of elliptic partial differential equations. This fact, as well as its favorable stability properties, make it an attractive alternative for diffusion solvers based on finite element techniques. In the current study, special attention has been given to an iterative solver using the changing minimal residual method based on the Hessenberg reduction and its performance has been examined for the generalized finite element solution of two test examples for transient diffusion problems. Comparisons to results obtained with the well-established iterative solver using the generalized minimal residual method and other dense direct solvers widely used in generalized finite element methods have also been presented. The aim is to evaluate the performance of the linear solvers for solving the systems resulting form the generalized finite element discretizations. The comparison is carried out mainly in terms of errors in the problem solution rather than the residual of the linear solver used in the solution process. To identify the contribution of the linear solver in the solution error we assume that the results obtained using the direct singular value decomposition solver are a reference base for the other linear solvers.

For the considered problems, the changing minimal residual method based on the Hessenberg reduction has been shown to be superior to all considered methods. The obtained results have also shown that the direct solvers based on canonical Gaussian eliminations have serious limitations when dealing with the ill-conditioned matrices associated with the linear systems in the generalized finite element method. This may encourage other researchers to adopt iterative solvers when using generalised finite element methods despite the inherited ill-conditioning property in their linear systems. In addition, compared to the GMRES solver the CMRH solver has consistently achieved the same set tolerance as the GMRES solver but with about 10% fewer iterations than the GMRES solver. For heat conduction problems in composite materials, the CMRH solver has also demonstrated better stability than the GMRES solver. Although we have restricted our numerical simulations to the case of two-dimensional problems, the more important implications of our research concern the use of effective iterative methods for three-dimensional problems in radiation-conduction applications. We believe that for these problems, the changing minimal residual method based on the Hessenberg reduction might be very effective since it reduces the number of iterations needed for convergence and it is relatively inexpensive to implement.

Acknowledgments. The authors wish to thank the anonymous reviewers for their helpful comments on an earlier draft of the manuscript.

References

[1] A. Alia, H. Sadok, and M. Souli. CMRH method as iterative solver for boundary element acoustic systems. *Eng. Anal. Bound. Elem.*, 36:346–350, 2012.

[2] I. Babuška, X. Huang, and R. Lipton. Machine computation using the exponentially convergent multiscale spectral generalized finite element method. *ESAIM: Mathematical Modelling and Numerical Analysis*, 48(02):493–515, 2014.

[3] E Béchet, H Minnebo, Nicolas Moës, and B Burgardt. Improved implementation and robustness study of the X-FEM for stress analysis around cracks. *International Journal for Numerical Methods in Engineering*, 64(8):1033–1056, 2005.

[4] T. Belytschko, R. Gracie, and G. Ventura. A review of extended/generalized finite element methods for material modeling. *Modelling and Simulation in Materials Science and Engineering*, 17(4):043001, 2009.

[5] P. Bettess and J.A. Bettess. A profile matrix solver with built-in constraint facility. *Engineering Computations*, 3(3):209–216, 1986.

[6] G.C. Diwan, M.S. Mohamed, M. Seaid, J. Trevelyan, and O. Laghrouche. Mixed enrichment for the finite element method in heterogeneous media. *International Journal for Numerical Methods in Engineering*, 101(1):54–78, 2015.

[7] S. Duminil. A parallel implementation of the CMRH method for dense linear systems. *Numer. Algor*, 63:127–142, 2013.

[8] S. Duminil, M. Heyouni, P. Marion, and H. Sadok. Algorithms for the CMRH method for dense linear systems. *Numer. Algor*, 71:383–394, 2016.

[9] A. El Kacimi and O. Laghrouche. Wavelet based ILU preconditioners for the numerical solution by PUFEM of high frequency elastic wave scattering. *Journal of Computational Physics*, 230(8):3119–3134, 2011.

[10] M.R. Hematiyan, M. Mohammadi, L. Marin, and A. Khosravifard. Boundary element analysis of uncoupled transient thermo-elastic problems with time- and space-dependent heat sources. *Applied Mathematics and Computation*, 218:1862–1882, 2011.

[11] M. Heyouni and H. Sadok. A new implementation of the CMRH method for solving dense linear systems. *J. Comput. Appl. Math.*, 213:387–399, 2008.

[12] R. Hiptmair, A. Moiola, and I. Perugia. A survey of Trefftz methods for the Helmholtz equation. *arXiv preprint arXiv:1506.04521*, 2015.

[13] T. Huttunen, P. Gamallo, and R.J. Astley. Comparison of two wave element methods for the Helmholtz problem. *Communications in Numerical Methods in Engineering*, 25(1):35–52, 2009.

[14] Muhammad Iqbal, Heiko Gimperlein, M Shadi Mohamed, and Omar Laghrouche. An a posteriori error estimate for the generalized finite element method for transient heat diffusion problems. *International Journal for Numerical Methods in Engineering*, 110(12):1103–1118, 2017.

[15] D.J. Kim, S.G. Hong, and C.A. Duarte. Generalized finite element analysis using the preconditioned conjugate gradient method. *Applied Mathematical Modelling*, 2015.

[16] A. Klein and A. Godunov. *Introductory Computational Physics.* Cambridge University Press, 2006. Cambridge Books Online.

[17] O. Laghrouche, P. Bettess, and R.J. Astley. Modelling of short wave diffraction problems using approximating systems of plane waves. *Int. J. Numer. Meth. Engng.*, 54:1501–1533, 2002.

[18] T Luostari, T Huttunen, and P Monk. Improvements for the ultra weak variational formulation. *International Journal for Numerical Methods in Engineering*, 94(6):598–624, 2013.

[19] J.M. Melenk and I. Babuška. The partition of unity finite element method: basic theory and applications. *Comput. Methods Appl. Mech. Engrg*, 139:289–314, 1996.

[20] A. Menk and S. Bordas. A robust preconditioning technique for the extended finite element method. *International Journal for Numerical Methods in Engineering*, 85(13):1609–1632, 2011.

[21] M.S. Mohamed, M. Seaid, J. Trevelyan, and O. Laghrouche. A partition of unity FEM for time-dependent diffusion problems using multiple enrichment functions. *Int. J. Numer. Meth. Engng.*, 93:245–265, 2013.

[22] M.S. Mohamed, M. Seaid, J. Trevelyan, and O. Laghrouche. Time-independent hybrid enrichment for finite element solution of transient conduction-radiation in diffusive grey media. *J. Comp. Phys.*, 251:81–101, 2013.

[23] M.S. Mohamed, M. Seaid, J. Trevelyan, and O. Laghrouche. An enriched finite element model with *q*-refinement for radiative boundary layers in glass cooling. *Journal of Computational Physics*, 258:718–737, 2014.

[24] E.A. Munts, S.J. Hulsho, and R. de Borst. The partition-of-unity method for linear diffusion and convection problems: accuracy, stabilization and multiscale interpretation. *Int. J. Numer. Meth. Engng.*, 43:199–213, 2003.

[25] P. O'Hara, C.A. Duarte, and T. Eason. Transient analysis of sharp thermal gradients using coarse finite element meshes. *Comput. Methods Appl. Mech. Engrg.*, 200:812–829, 2011.

[26] M.J. Peake, J. Trevelyan, and G. Coates. Extended isogeometric boundary element method (XIBEM) for three-dimensional medium-wave acoustic scattering problems. *Computer Methods in Applied Mechanics and Engineering*, 284:762–780, 2015.

[27] E. Perrey-Debain, O. Laghrouche, P. Bettess, and J. Trevelyan. Plane-wave basis finite elements and boundary elements for three-dimensional wave scattering. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 362(1816):561–577, 2004.

[28] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statis. Comput.*, 7:856–869, 1986.

[29] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[30] H. Sadok. CMRH : A new method for solving nonsymmetric linear systems based on the Hessenberg reduction algorithm. *Numer. Algorithms*, 20(4):303–321, 1999.

[31] H. Sadok and D. B. Szyld. A new look at CMRH and its relation to GMRES. *BIT Numer. Math.*, 52:485–501, 2012.

[32] T. Strouboulis, I. Babuška, and K. Copps. The design and analysis of the generalized finite element method. *Computer methods in applied mechanics and engineering*, 181(1):43–69, 2000.

[33] T. Strouboulis, I. Babuška, and R. Hidajat. The generalized finite element method for Helmholtz equation: theory, computation, and open problems. *Computer Methods in Applied Mechanics and Engineering*, 195(37):4711–4731, 2006.

[34] N. Sukumar, D.L. Chopp, E. Béchet, and N. Moës. Three-dimensional non-planar crack growth by a coupled extended finite element and fast marching method. *International Journal for Numerical Methods in Engineering*, 76(5):727, 2008.