

An Improved Non-Intrusive Global-Local Approach for Sharp Thermal Gradients in a Standard FEA Platform

J. Plews¹, C.A. Duarte^{1*}, and T. Eason²

¹*Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Newmark Laboratory, 205 North Mathews Avenue, Urbana, Illinois 61801, USA*

²*U.S. Air Force Research Laboratory, Air Vehicles Directorate, Wright-Patterson AFB, Ohio, USA*

SUMMARY

Several classes of important engineering problems – in this case, problems exhibiting sharp thermal gradients – have solution features spanning multiple spatial scales and, therefore, necessitate advanced *hp* finite element discretizations. Although *hp*-FEM is unavailable off-the-shelf in many predominant commercial analysis software packages, the authors herein propose a novel method to introduce these capabilities via a generalized finite element method non-intrusively in a standard FEA platform. The methodology is demonstrated on two verification problems as well as a representative, industrial-scale problem. Numerical results show that the techniques utilized allow for accurate resolution of localized thermal features on structural-scale meshes without *hp*-adaptivity or the ability to account for complex and very localized loads in the FEA code itself. This methodology enables the user to take advantage of all the benefits of both *hp*-FEM discretizations and the appealing features of many available CAE/FEA software packages in order to obtain optimal convergence for challenging multiscale problems.

Copyright © 2011 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: generalized finite element method; *hp*-finite element method; multiscale problems; partition of unity method; X-FEM

1. INTRODUCTION

1.1. Background

A growing number of problems encountered in engineering practice today require consideration of phenomena encompassing multiple spatial scales of interest. One example of particular interest – and which is part of the motivation behind this work – lies in the structural analysis of hypersonic flight vehicles. At very high airspeeds, rapid variations in the density and temperature of the compressible flow lead to shock impingements on the skin of the vehicle. Interactions between shock

*Correspondence to: C.A. Duarte, Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Newmark Laboratory, 205 North Mathews Avenue, Urbana, Illinois 61801, USA. E-mail: caduarte@illinois.edu.

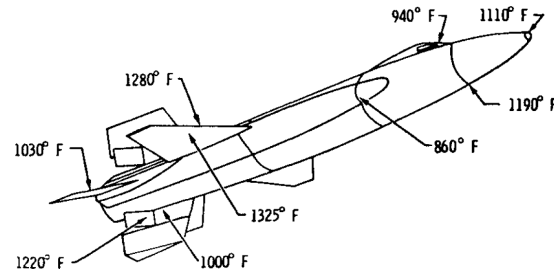


Figure 1. Measured temperatures on the skin of a North American X-15 spaceplane from an actual hypersonic flight at Mach 5.0 (cf. [1]).

waves, typically occurring most severely on the leading edge of the aircraft wing, have been shown to lead to very intense, localized thermomechanical loads, and characterization of these complicated effects itself has been an active research topic [2–8]. A comprehensive, historical overview of the challenges posed by aero-thermal-mechanical effects in hypersonic structures and their importance is given in [1]. In the scope of this paper, however, only the thermal component of loadings is considered.

Based on experimental investigations, intense, localized heat fluxes can be concentrated on an area just microns in width, or many orders of magnitude smaller than the structure. However, these loadings may have a drastic effect on the overall behavior of the structure as well as the neighborhood of the loading itself, and thus the two disparate scales of interest may not in general be considered separately.

1.2. Present Approaches & Limitations

In current design practice, commercial finite element analysis (FEA) software packages are commonly used to predict the response of the hypersonic vehicles of interest, as well as other complex structures exhibiting multiscale behavior. Optimal finite element discretizations for this class of problems require state-of-the-art localized, adaptive mesh refinement combined with high-order polynomial approximations [9, 10]. Unfortunately, performing such *hp*-adaptivity in commercially available FEA software is often prohibitively difficult or, in many cases, even impossible. Additionally, treatment of sharp, localized thermal loads requires special attention and cannot in general be computed automatically in conventional FEA software. More importantly, if these loadings are applied on meshes designed to capture only the global response of the structure, the error of the finite element solution may be large even far away from the localized features due to so-called pollution error [11, 12].

Problems involving multiple spacial scales of interest are traditionally handled in engineering practice by global-local or submodeling analysis [13, 14]. The global-local FEM procedure involves two steps. First, the solution of the problem is computed on a coarse, global, quasi-uniform mesh. Next, small subdomains containing local features of interest like cracks or other stress raisers are extracted from the global domain and analyzed using the global solution as boundary conditions. The use of the crude global solution as boundary conditions for local problems is a key point in the procedure, as it may lead to local solutions with large errors [15].

Several authors have proposed improvements to the global-local procedure aiming at addressing the issue of local error [16–20]. In the Generalized Finite Element Method (GFEM) proposed in

[21,22], this issue is addressed by going one step further in a global-local analysis; the local solution is used as an enrichment function for the global problem, rather than the final solution. This enriched global problem is solved again using the factorization computed in the first step of the global-local analysis [23]. This methodology is denoted a GFEM with global-local enrichment functions (GFEM^{gl}). Numerical and mathematical analysis of the GFEM^{gl} have shown [12, 15, 22–24] that the solution of the enriched global problem has an accuracy comparable to a direct simulation of the problem with fine-scale features discretized in the global problem.

1.3. Contributions of this Paper

In a standard global-local analysis different softwares or even different methodologies can be used to solve the global and local problems. This flexibility, however, is lost when solving the enriched global problem in the GFEM^{gl} approach discussed above, which requires special enrichment functions and thus prevents the use of available commercial FEA software to solve the usually large and complex global problem and specialized *hp*-adaptive software to solve local problems. To address this issue, the authors propose in this paper a new algorithm enabling the solution of the enriched global problem in the GFEM^{gl}. The resulting methodology effectively introduces *hp*-adaptive capabilities to an existing FEA platform without any code modifications. The methodology is demonstrated here using the commercial software Abaqus, which was chosen due to its popularity in the engineering community, its heat transfer capabilities, and its robust scripting interface. The method used, however, is quite general and can be applied to a variety of FEA software.

This non-intrusive GFEM^{gl} approach offers many benefits, circumventing the need for *hp*-adaptivity in standard FEA codes and enabling accurate computation of sharp thermal loads on structural-scale meshes. As a result, it will be shown to yield optimal convergence for this class of problems. Additionally, an example will demonstrate that the methodology is able to provide a great deal of flexibility in handling a variety of multiscale analysis cases for the same structural model of interest. While this work focuses on intense, localized heat sources, the methodology is relevant to a much broader range of problems exhibiting multiscale phenomena without modifications to the overall approach.

2. PROBLEM DEFINITION

Consider a domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega = \Gamma^u \cup \Gamma^f \cup \Gamma^c$, where $\Gamma^u \cap \Gamma^f = \emptyset$, $\Gamma^u \cap \Gamma^c = \emptyset$, and $\Gamma^c \cap \Gamma^f = \emptyset$. The strong form of the governing partial differential equation is given by Poisson's equation,

$$\nabla \cdot (\boldsymbol{\kappa} \nabla u) = -Q(\mathbf{x}) \quad \text{in } \Omega, \quad (1)$$

given the physical interpretation of heat transfer, where $u(\mathbf{x}) \equiv u(x, y, z)$ is the temperature field, $\boldsymbol{\kappa}$ is the thermal conductivity tensor, and $Q(\mathbf{x}) \equiv Q(x, y, z)$ is the internal heat source. Boundary

conditions prescribed on $\partial\Omega$ are given by

$$\begin{aligned} u &= \bar{u} \quad \text{on } \Gamma^u \\ -\boldsymbol{\kappa}\nabla u \cdot \mathbf{n} &= \bar{f} \quad \text{on } \Gamma^f \\ -\boldsymbol{\kappa}\nabla u \cdot \mathbf{n} &= \alpha(u - u_\infty) \quad \text{on } \Gamma^c \end{aligned} \quad (2)$$

where \mathbf{n} is the outward unit normal vector to Γ^f and Γ^c , and \bar{f} and \bar{u} are prescribed normal heat flux and temperature, respectively. Parameter α is the convection coefficient, and u_∞ is the free-stream temperature for convective conditions.

3. GENERALIZED FEM AND THE GFEM^{gl}

3.1. GFEM Approximations

The generalized finite element method (GFEM) [25–29] is a Galerkin method based on discretization spaces defined using the concept of a partition of unity (*PoU*). Partition of unity methods originated from the work of Babuška et al. [25, 26, 30] as well as Duarte and Oden [28, 31–34]. The extended finite element method, or X-FEM, is another example of a method based upon *PoU* concepts with many similarities to the GFEM [35, 36]. The usefulness of the GFEM hinges on the idea that the partition of unity can be enriched, or combined with local function approximation spaces built around *a priori* knowledge about the solution of a given problem.

In the GFEM, standard finite element shape functions φ_α are chosen as the partition of unity, since φ_α , $\alpha = 1, \dots, N$, in a mesh covering a domain Ω with N nodes are such that $\sum_{\alpha=1}^N \varphi_\alpha(\mathbf{x}) = 1$ for all \mathbf{x} in Ω . A GFEM shape function $\phi_{\alpha i}$ is then computed as the product of the FEM *PoU* φ_α and an enrichment function $L_{\alpha i}$,

$$\phi_{\alpha i}(\mathbf{x}) = \varphi_\alpha(\mathbf{x})L_{\alpha i}(\mathbf{x}) \quad (\text{no summation on } \alpha), \quad (3)$$

where α is a node in the FE mesh. Figure 2 illustrates shape function construction for various types of enrichments.

3.2. GFEM^{gl}

Although special GFEM enrichment functions may be designed to accommodate analytical solution characteristics, for instance, in the case of fracture mechanics [37] or modeling of polycrystalline structures [38], in many cases *a priori* knowledge of the solution behavior is limited. Moreover, especially in large problems, performing *hp*-adaptivity on a structural-scale mesh in order to account for localized solution features may add many degrees of freedom to the problem and, thus, may prove much too computationally expensive.

The GFEM with global-local enrichment functions (GFEM^{gl}) [12, 22], however, allows for on-the-fly, numerical creation of custom enrichments via the solution of smaller, *hp*-adapted local problems which enclose features of interest in the structural-scale (global) domain. Thus, expensive mesh refinements and localized, high-order polynomial enrichments need not be done in the global domain itself, and only a few degrees of freedom are added to the global problem as a result of the

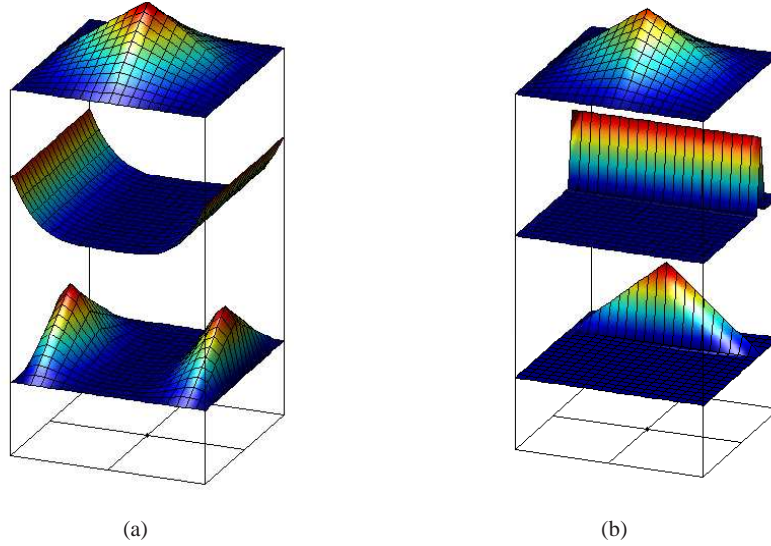


Figure 2. Construction of a GFEM shape function – from top to bottom, ϕ_α , the FEM partition of unity, $L_{\alpha i}$, an enrichment function, and $\phi_{\alpha i}$, the resulting shape function for (a) a polynomial enrichment function, and (b) a custom, non-polynomial enrichment.

numerically-built enrichment functions. The most basic GFEM^{gl} solution procedure is comprised of three primary steps – a coarse-scale initial solution, extraction and solution of local problems, and enrichment and reanalysis [22, 23] of the global problem based on local solutions.

Initial Global (IG) Problem An initial coarse-scale analysis is first performed on the global problem on $\bar{\Omega}_G = \Omega_G \cup \partial\Omega_G$, yielding initial solution u^0 . The initial global problem is formulated as, for all $v^0 \in X_G(\Omega_G)$, find $u^0 \in X_G(\Omega_G)$ such that

$$\begin{aligned} & \int_{\Omega_G} \nabla u^0 \mathbf{\kappa} \nabla v^0 d\Omega + \eta \int_{\Gamma_G^u} u^0 v^0 d\Gamma + \int_{\Gamma_G^c} \alpha u^0 v^0 d\Gamma \\ &= \int_{\Omega_G} q v^0 d\Omega + \int_{\Gamma_G^f} \bar{f} v^0 d\Gamma + \eta \int_{\Gamma_G^u} \bar{u} v^0 d\Gamma + \int_{\Gamma_G^c} \alpha u_\infty v^0 d\Gamma, \end{aligned} \quad (4)$$

where η is a predetermined penalty parameter for enforcement of Dirichlet boundary conditions, and $X_G(\Omega_G)$ is a GFEM discretization of $H^1(\Omega_G)$.

Local Problem(s) The initial global solution resulting from the linear system of equations implied by (4) is then used directly as a Dirichlet boundary condition in a local problem. Convective boundary conditions can also be used [23]. The local domain is comprised of $\bar{\Omega}_L = \Omega_L \cup \partial\Omega_L$, a subdomain of Ω_G , which in practice is taken to be a user-specified, small subset of finite elements extracted (copied) from the global problem upon which *hp*-adaptivity may be performed. However, the local domain may be selected and mesh refinements done automatically using *a posteriori* error estimates on the initial global solution, u^0 . The local problem is formulated as follows: for

all $v_L \in X_L(\Omega_L)$, find $u_L \in X_L(\Omega_L)$ such that

$$\begin{aligned} & \int_{\Omega_L} \nabla u_L \mathbf{\kappa} \nabla v_L d\Omega + \eta \int_{\partial\Omega_L \setminus (\partial\Omega_L \cap \Gamma_G^f)} u_L v_L d\Gamma + \int_{\partial\Omega_L \cap \Gamma_G^c} \alpha u_L v_L d\Gamma \\ &= \eta \int_{\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)} u^0 v_L d\Gamma + \eta \int_{\partial\Omega_L \cap \Gamma_G^u} \bar{u} v_L d\Gamma + \int_{\Omega_L} q v_L d\Omega + \int_{\partial\Omega_L \cap \Gamma_G^f} \bar{f} v_L d\Gamma \\ &+ \int_{\partial\Omega_L \cap \Gamma_G^c} \alpha u_\infty v_L d\Gamma, \end{aligned} \quad (5)$$

where $X_L(\Omega_L)$ is again a GFEM discretization of $H^1(\Omega_L)$. The initial global and local problem steps comprise a procedure akin to the familiar global-local FEM [13, 14, 39].

Enriched Global (EG) Problem In general, the local problem may not provide a satisfactory estimate of the actual global solution, due in large part to poor boundary conditions on local problems from the initial, coarse-scale analysis. Thus, taking the global-local analysis one step further, the solution u_L yielded by the local problem is used in the GFEM^{gl} to build so-called global-local shape functions

$$\phi_\alpha^{gl}(\mathbf{x}) = \phi_\alpha(\mathbf{x}) u_L(\mathbf{x}) \quad (6)$$

numerically, which are in turn added to the global approximation space and utilized to enrich and re-solve the global problem as defined in (4). This step is termed the enriched global (EG) problem, the solution of which is denoted hereafter u^E .

Extensive numerical and mathematical analysis presented in [12, 22–24] demonstrates that the accuracy of u^E is close to that provided by a direct simulation of the problem with fine scale features discretized in the global problem. Since in the GFEM^{gl} the local solution u_L is used as an enrichment function for the EG problem, it only needs to capture the exact solution up to a constant. This is the case if the local boundary conditions are provided by the FEM, since the FEM solution oscillates around the exact solution. Further details on this analysis can be found in [24]. The convergence study presented in Section 5.2.1 shows that the proposed algorithm enjoys this same property (cf. Figure 11).

An *a priori* error estimate for the GFEM^{gl} accounting for the effect of inexact boundary conditions at local problem is presented in [24]. Two strategies to control this effect are also presented in that study, the first of which is based on the use of a buffer or over-sampling zone in the local problems. This strategy is adopted in this paper and described in more detail in Section 5.1. The second strategy analyzed in [24] and originally proposed in [12], is based on multiple global-local iterations.

4. A NON-INTRUSIVE GFEM^{gl} ALGORITHM FOR USE IN A FE SOLVER

As previously discussed, the extension of this global-local solution strategy to a legacy FEA platform is nontrivial due to the enriched global step, which introduces custom enrichment functions into the global approximation space. In recent years, non-intrusive methodologies such as the one proposed here have been actively investigated in order to enable a smooth transition of advanced GFEM/X-FEM discretizations to application in popular and venerable commercial FEA codes. For

instance, Giner et al. [40], Shi et al. [41, 42], and Xu and Yuan [43, 44] propose implementing X-FEM for fracture mechanics applications in a commercial FEA code, Abaqus, by using customized elements. However, prior to each analysis, the user must manually select elements in the global model which will possess X-FEM enrichments, which is a prominent disadvantage of this strategy, since the global model must then be altered for each separate analysis case or crack configuration. This complication could prove extremely inconvenient when dealing with very large global models.

Gendre et al. [45, 46] propose a non-intrusive algorithm implementing nonlinear FEM in a standard FE solver which is somewhat similar to the GFEM methodology introduced here, where a patch of elements containing a localized plastic zone of interest is “exactly” extracted from the global problem by a Schur complement method. However, in this case, the local patch boundary must exactly match the global problem mesh where the local patch is to be inserted, so that no refinement can be performed along the local boundary. Because of the partition of unity approach, this is not an issue in the GFEM^{gl}, which is demonstrated on a sample mesh in Figure 3. Also, the Schur complement of the *local* degrees of freedom – as opposed to the global degrees of freedom, the approach adopted here – is computed, which in general comprises many degrees of freedom and thus proves very expensive for large global problems (so to avoid this issue, a procedure to approximate the Schur complement is adopted).

The procedure which will be described here takes a different approach from the aforementioned philosophies. In this case, the proposed algorithm uses two stand-alone codes – a standard FE solver, and another stand-alone, in-house GFEM code. The GFEM^{gl} solution is partitioned, and portions are solved in each code separately (explained further in Section 4.1). Thus, the only communication which takes place between the two is controlled by a converter code (detailed in Appendix A). The approach proposed herein is perhaps most similar to Bordas and Moran [47] and is in a sense the “inverse” procedure to the Schur complement method for scale-bridging described in Gendre et al. [45, 46], which will be explained much further in the following section.

It is important to note, also, that when using the proposed approach, any *hp*-FEM code could theoretically be used instead of a GFEM code in the local problem, since the only unique aspect of the GFEM which is used here is the partition of unity to bridge scales between global and local solutions. This is because only polynomial enrichments – no special, analytical GFEM/X-FEM enrichment functions – are present. This is in stark contrast to fracture mechanics problems, where special GFEM step and branch function enrichments are used to represent the behavior of cracks in the domain. In this case, using an *hp*-FEM solver in the local problem would not be possible, since the local mesh would need to be designed to fit the crack, and thus the local mesh could not in general be nested inside the global mesh. This would make accurate numerical integration of special enrichment functions around, for example, a crack front extremely difficult.

4.1. Partitioned System of Equations

The GFEM^{gl} by its very nature is readily extensible to an approach using multiple-solvers; one standard FE solver (referred to here as **FE-S**) is made to handle the coarse-scale, global problem, and another GFEM solver (**GFE-S**) orchestrates the analysis by handling the local and enriched global problem aspects. The essence of the proposed algorithm is as follows: The GFEM solution

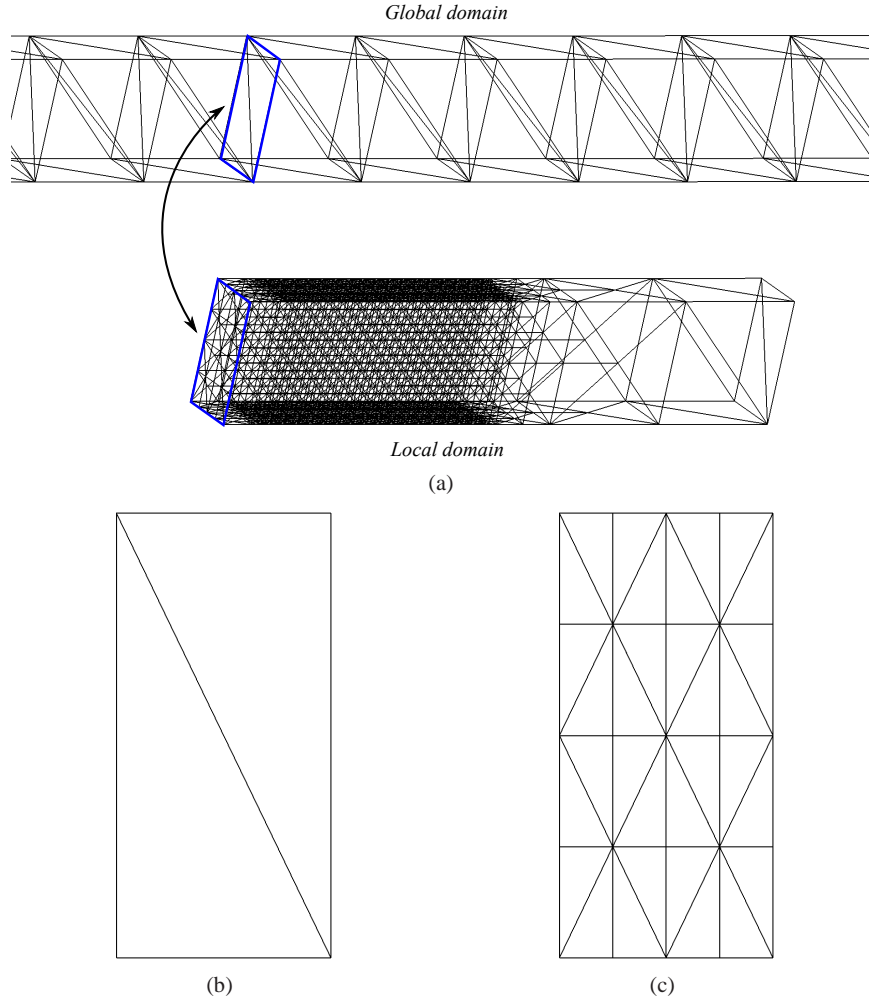


Figure 3. A demonstration of the possibility for arbitrarily refined meshes at the local domain boundary, irrespective of the matching global problem mesh. The use of the local solution as an enrichment function, sewn together with the global problem approximation by a partition of unity, allows for much more flexibility than, for example, “exact” extraction and solution of a localized subset of the domain. Figure 3(a) shows the corresponding mesh faces in the global and the local domain, respectively, while (b) and (c) show a zoom-in on the difference in mesh refinement between the two.

of the enriched global problem, u^E , can be partitioned as

$$u^E = \tilde{u}^0 + u^{gl} = \begin{bmatrix} \mathbf{N}^0 & \mathbf{N}^{gl} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}^0 \\ \mathbf{u}^{gl} \end{bmatrix}, \quad (7)$$

where \mathbf{N}^0 has standard FEM shape functions and \mathbf{N}^{gl} has the global-local shape functions defined in (6). Vectors $\tilde{\mathbf{u}}^0$ and \mathbf{u}^{gl} have global and global-local enrichment degrees of freedom, respectively. Then, the gradient of the temperature field is defined as

$$\nabla u^E = \begin{bmatrix} \mathbf{B}^0 & \mathbf{B}^{gl} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}^0 \\ \mathbf{u}^{gl} \end{bmatrix}. \quad (8)$$

The resulting system of equations in the enriched global problem, as formulated in and implied by (4), $\mathbf{K}\underline{\mathbf{u}}^E = \mathbf{f}$ may then also be partitioned as

$$\begin{bmatrix} \mathbf{K}^0 & \mathbf{K}^{0,gl} \\ \mathbf{K}^{gl,0} & \mathbf{K}^{gl} \end{bmatrix} \begin{bmatrix} \underline{\mathbf{u}}^0 \\ \underline{\mathbf{u}}^{gl} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^0 \\ \mathbf{f}^{gl} \end{bmatrix}, \quad (9)$$

where

$$\mathbf{K}^0 := \int_{\Omega_G} (\mathbf{B}^0)^T \boldsymbol{\kappa} \mathbf{B}^0 d\Omega$$

is computed by **FE-S**, and

$$\begin{aligned} \mathbf{K}^{0,gl} &:= \int_{\Omega_L} (\mathbf{B}^0)^T \boldsymbol{\kappa} \mathbf{B}^{gl} d\Omega \\ \mathbf{K}^{gl} &:= \int_{\Omega_L} (\mathbf{B}^{gl})^T \boldsymbol{\kappa} \mathbf{B}^{gl} d\Omega \end{aligned}$$

are computed in **GFE-S**. The solution to System (9) can then be found by static condensation on $\underline{\mathbf{u}}^{gl}$, since, in general, $\dim(\underline{\mathbf{u}}^0) \gg \dim(\underline{\mathbf{u}}^{gl})$. From the first equation in System (9),

$$\begin{aligned} \underline{\mathbf{u}}^0 &= (\mathbf{K}^0)^{-1} \mathbf{f}^0 - (\mathbf{K}^0)^{-1} \mathbf{K}^{0,gl} \underline{\mathbf{u}}^{gl} \\ &= \underline{\mathbf{u}}^0 - \mathbf{S}^{0,gl} \underline{\mathbf{u}}^{gl}, \end{aligned} \quad (10)$$

where $\mathbf{K}^{0,gl}$ are known as “pseudo-loads,”

$$\mathbf{S}^{0,gl} := (\mathbf{K}^0)^{-1} \mathbf{K}^{0,gl}$$

are known as “pseudo-solutions” corresponding to the pseudo-loads, and

$$\underline{\mathbf{u}}^0 = (\mathbf{K}^0)^{-1} \mathbf{f}^0$$

is the initial, coarse-scale solution. Both $\mathbf{S}^{0,gl}$ and $\underline{\mathbf{u}}^0$ may be computed by **FE-S** using forward and backward substitution on a factorization of the coarse-scale global stiffness matrix, \mathbf{K}^0 . Because \mathbf{K}^0 does not change between initial global and enriched global problems, the factorization of \mathbf{K}^0 may also, if possible, be stored in **FE-S** after the initial global problem step and reused in the enriched global problem in order to reduce computational cost as well as solution time[†]. Substituting Equation (10) in the second equation in System (9) and rearranging,

$$\underbrace{(\mathbf{K}^{gl} - \mathbf{K}^{gl,0} \mathbf{S}^{0,gl})}_{\hat{\mathbf{K}}_{gl}} \underline{\mathbf{u}}^{gl} = \underbrace{\mathbf{f}^{gl} - \mathbf{K}^{gl,0} \underline{\mathbf{u}}^0}_{\hat{\mathbf{f}}_{gl}}, \quad (11)$$

[†]The FE solver used for numerical examples in this paper, Abaqus, has limitations on reuse of the factorized global matrix. Thus, the factorization must be computed multiple times per analysis. This limitation has also been addressed in [45].

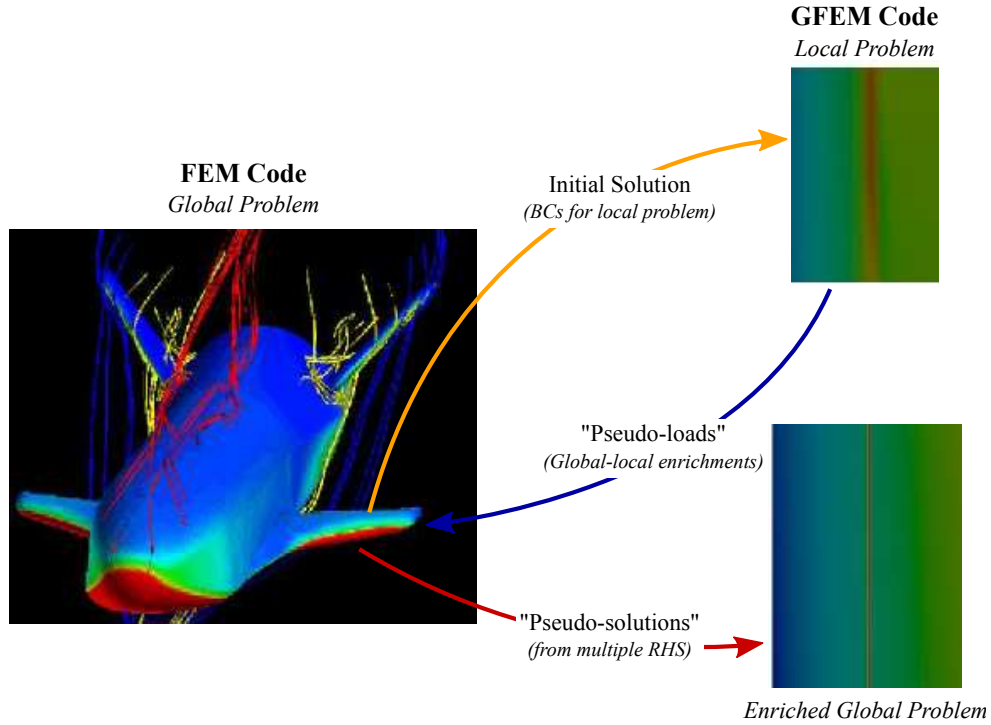


Figure 4. Illustration of non-intrusive algorithm – exchange of “pseudo-loads” and “pseudo-solutions” between FEM and GFEM codes.

so the solution corresponding to global-local degrees of freedom comes directly from the solution of

$$\hat{\mathbf{K}}^{gl} \underline{\mathbf{u}}^{gl} = \hat{\mathbf{f}}^{gl},$$

where $\hat{\mathbf{K}}^{gl}$ can be interpreted as the Schur complement of \mathbf{K}^0 .

Thus, the static condensation algorithm requires only the exchange of pseudo-loads and pseudo-solutions between **FE-S** and **GFE-S**, making this approach extensible to almost any FEA software package. The algorithm described here is also illustrated graphically in Figure 4. A detailed, step-by-step description of the procedure used to implement this algorithm in the FEA code chosen for this paper, Abaqus, is given in Appendix A.

4.2. Treatment of Rough Loads

Standard FEM solvers cannot typically handle sharp heat fluxes applied on meshes designed to capture only the coarse-scale component of the solution. Furthermore, application of such sharp loadings on coarse meshes may cause error in the finite element solution to propagate even far from the localized feature (so-called “error pollution”) [11, 12]. Thus, the authors also propose an improved strategy for handling sharp loadings within the non-intrusive framework.

The global load vector, \mathbf{f}^0 above, is decomposed as

$$\mathbf{f}^0 = \mathbf{f}_R^0 + \mathbf{f}_S^0,$$

where \mathbf{f}_R^0 is the “rough,” sharp, localized portion of the load, and \mathbf{f}_S^0 is the remaining “smooth” portion. Although **FE-S** is able to compute \mathbf{f}_S^0 without difficulty, the additional sharp load \mathbf{f}_R^0 must be numerically integrated using **GFE-S**.

As was examined in [12], it will be shown that applying *only* the smooth loading, \mathbf{f}_S^0 , in the initial global problem not only gives a satisfactory estimate of the solution to develop local problem boundary conditions but also eliminates aforesaid error pollution effects.

However, the *total* global load, including sharp features, must somehow be computed accurately and applied in the enriched global problem. Taking advantage of information available from the local problem step, the highly refined, *hp*-adapted local domain elements may be used as so-called integration elements in the global problem, coupled with a high-order numerical quadrature rule, to recompute the total load, \mathbf{f}^0 , in **GFE-S** just prior to the enriched global problem. The new global load vector is then passed to **FE-S** alongside the pseudo-loads, and this right hand side is solved during the enriched global problem phase (further explanation of this procedure can be found in Appendix A). The results of this “improved” global analysis effectively supersede the solution obtained from the coarse-scale, initial global step and are used in place of the initial global solution in the static condensation algorithm for the enriched global solution.

This so-called rebuilding of the global load vector serves to provide a more accurate estimate of the coarse-scale, global degrees of freedom \mathbf{u}^0 , which is critical to obtaining optimal convergence in the enriched global solution. The effects of these strategies on numerical results will be shown for a sample problem in Section 5.2.

In typical FEA software, even finding a way to apply the proper sharp loading – analytically defined or otherwise – can be a nontrivial task. It will be demonstrated further in Section 5.2 that standard, commercial finite element codes may not necessarily have any built-in protocol for handling sharp, user-defined loadings on coarse meshes, providing additional benefit to the proposed approach.

4.3. Adoption of a Quadratic Tetrahedron Partition of Unity

Even when using global-local enrichments, on a coarse, global mesh, a linear approximation of the global solution to many classes of problems often may not yield satisfactory error levels. In the GFEM, it is common practice to choose a linear partition of unity and enrich these “hat” functions with higher-order polynomials in order to improve global approximations [27, 28]. Similarly, in the state-of-the-art *hp*-version of the finite element method, arbitrarily high-order shape functions may be hierarchically added to elements in a FE mesh [9].

However, in many available 3D commercial FEA codes, user options for the order of the approximation are often quite limited[‡]. Moreover, higher-order – in this case, quadratic – elements are generally implemented through standard Lagrangian finite element shape functions, by adding additional nodes to the tetrahedral element. Thus, the partition of unity used in the GFEM^{gl} enriched global model must similarly accommodate the ten-noded quadratic tetrahedral element. In GFEM^{gl} local problems, however, ten-noded quadratic elements from the global problem may be directly converted to equivalent linear tetrahedral elements upon which *hp*-adaptivity may be performed just

[‡]In the commercial code Abaqus, used for numerical examples in this paper, only linear “TET4” or quadratic “TET10” tetrahedral elements are implemented for heat transfer simulations.

as before. Some examples of the improvement in enriched global solutions and convergence results thanks to the use of a quadratic global approximation in a standard FE solver will be shown in Section 5.

5. NUMERICAL EXAMPLES

5.1. L-shaped Domain

The non-intrusive GFEM^{gl} algorithm, here implemented using the commercial code Abaqus as the FE solver (henceforth Abaqus+GFEM^{gl}), was verified by solving a small, three-dimensional L-shaped domain of overall dimensions $100 \times 100 \times 10$ mm, illustrated in Figure 5. The global domain was meshed with linear (TET4) and quadratic (TET10) tetrahedral elements 10 mm in size. Inhomogeneous temperature boundary conditions of 150°C and -150°C were applied to the top and right faces of the domain, respectively, with the rest of the boundary remaining insulated. Thus, the problem exhibits a sharp heat flux singularity at the reentrant corner without the need to compute a sharp load vector, making it an ideal verification problem.

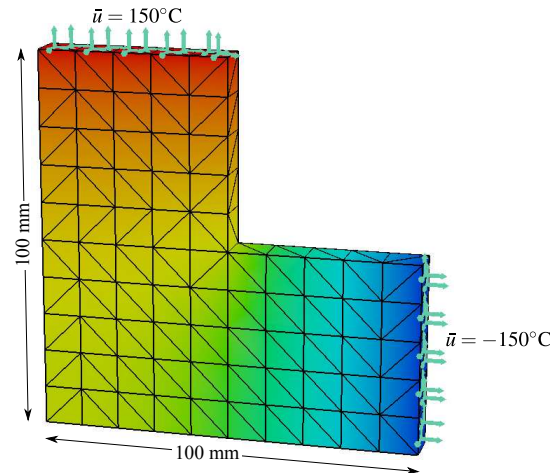


Figure 5. L-shaped domain verification problem. The enriched global temperature field is illustrated here on the structured tetrahedral mesh used in Abaqus.

Using the GFEM^{gl} methodology, the local problem was chosen as a neighborhood around the interior corner, $\Omega_L = \{\mathbf{x} | 20 < x < 80; 20 < y < 80; 0 < z < 10\}$, while the corresponding global-local enrichment zone on the global domain was chosen to be slightly smaller, enclosing only the first layer of nodes α around the corner, $\{\alpha | 40 \leq x_\alpha \leq 60; 40 \leq y_\alpha \leq 60; 0 \leq z_\alpha \leq 10\}$. The difference in size between the local domain and corresponding global-local enrichment region in this case serves to ameliorate the effect of inaccurate boundary conditions on the local domain [24]. Global and local domains for this problem are shown in Figure 6.

A reference solution u_{ref} was also generated using hp -GFEM (the GFEM analog of hp -FEM) with 7 levels of global mesh refinement, overall polynomial order $p = 3$, and 25 levels of geometric mesh refinement about the reentrant corner in the domain. Solutions from Abaqus+GFEM^{gl} were compared against the hp -GFEM reference solution. Table I shows the relative error levels in the

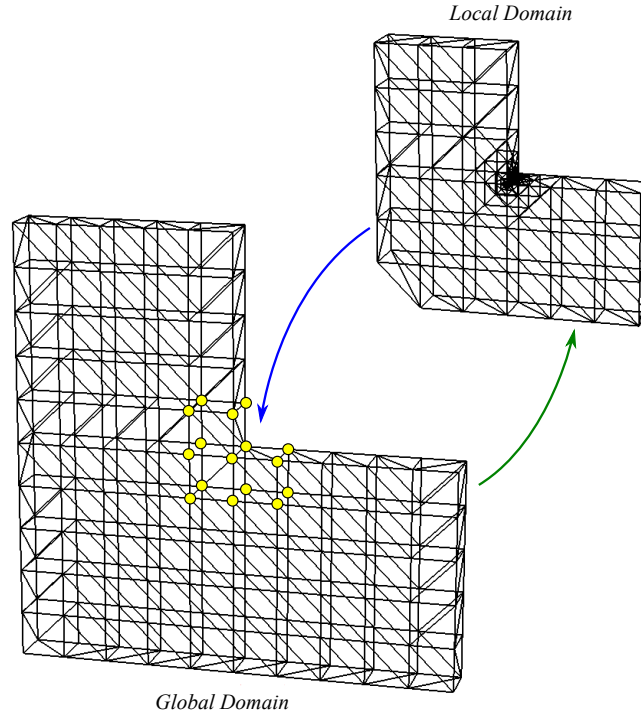


Figure 6. The global mesh (left) and local domain extracted from the global mesh (right) are illustrated here. Geometric mesh refinement is demonstrated in the local domain. Dots on the global domain represent the chosen global-local enrichment zone in the enriched global problem. Note that the local domain is larger than the corresponding enrichment zone – this serves to reduce error due to inaccurate boundary conditions from the coarse-scale global solution [24].

solution resulting from simulations using both TET4 and TET10 global meshes. Here, relative error in the energy norm of each finite element solution u_h is computed as

$$e_U^r = \sqrt{\frac{|B(u_{ref}, u_{ref}) - B(u_h, u_h)|}{|B(u_{ref}, u_{ref})|}}. \quad (12)$$

While linear TET4 results are poor, the solution can be drastically improved by taking advantage of quadratic TET10 elements in the Abaqus global problem. Based on these results, the corner

Table I. L-shaped domain results for TET4 and TET10 global meshes, 22 levels of *local domain* mesh refinement about the reentrant corner. Energy norm error, e_U^r , is computed with respect to the *hp*-GFEM reference solution. In both cases, the number of additional global-local enrichment dofs is small relative to the global problem size.

	dofs			e_U^r	
	<i>IG</i>	Local	<i>EG</i>	<i>IG</i>	<i>EG</i>
TET4	192	42,560	192 + 16	11.95%	5.46%
TET10	1,023	42,560	1,023 + 63	4.95%	0.44%

singularity is resolved very effectively by using Abaqus+GFEM^{gl} with *only* local domain mesh refinement, while adding just a few additional degrees of freedom to the global problem. For qualitative comparison, Figure 7 shows the heat flux fields corresponding to initial global and enriched global solutions, respectively, on the TET10 mesh.

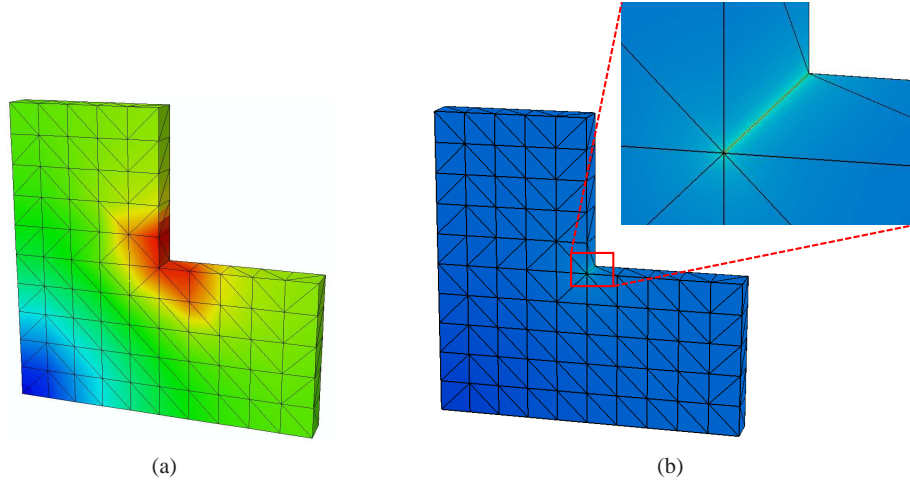


Figure 7. Heat flux field on an L-shaped domain (1,023 global dofs) corresponding to (a) Abaqus coarse-scale initial global solution and (b) Abaqus+GFEM^{gl} enriched global solution. The sharp flux resolution possible on a coarse mesh in Abaqus+GFEM^{gl} is also demonstrated here.

5.2. Beam Subjected to Localized Laser Heating

In this example, a sharp, steady-state Gaussian laser flux was applied to the front surface of a small aluminum beam of dimensions $12 \times 0.5 \times 0.24$ inches, illustrated in Figure 8.

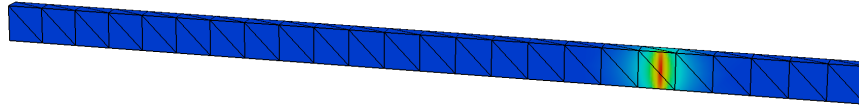


Figure 8. Temperature field on a beam subjected to a sharp laser heating. The coarse-scale tetrahedral mesh used in Abaqus is also shown. Note that the sharp heating is *not* aligned with an element edge in the global mesh.

The expression for the flux is given by

$$\bar{f}(\mathbf{x}) = I_0 * \frac{1}{2\pi a^2} * G(\mathbf{x}, b, a), \quad 8.0 \leq x \leq 10.0, \quad (13)$$

with

$$G(\mathbf{x}, b, a) = \exp\left(\frac{-(x-b)^2}{2a^2}\right). \quad (14)$$

Here, parameter $I_0 = 295 \frac{\text{ft-lbf}}{\text{s}}$ is the laser flux intensity, $a = 0.025\text{in}$ is the laser focus, or width, and $b = 9.3\text{in}$ dictates the x -coordinate of the center of the flux. The analytical sharp flux function is shown in Figure 9. Convective conditions were applied on the remainder of the boundary, with convection coefficient $\alpha = 11 \frac{\text{lbf}}{\text{ft-s}^\circ\text{C}}$ and free-stream temperature $u_\infty = 0^\circ\text{C}$.

5.2.1. Convergence Study Convergence behavior of the proposed non-intrusive GFEM^{gl} approach was investigated using this sample problem. A reference solution was developed using *hp*-GFEM with heavy mesh refinement and uniform polynomial order $p = 3$ resulting in 736,990 total degrees of freedom. Error was computed as relative error in the energy norm just as in Equation (12) using

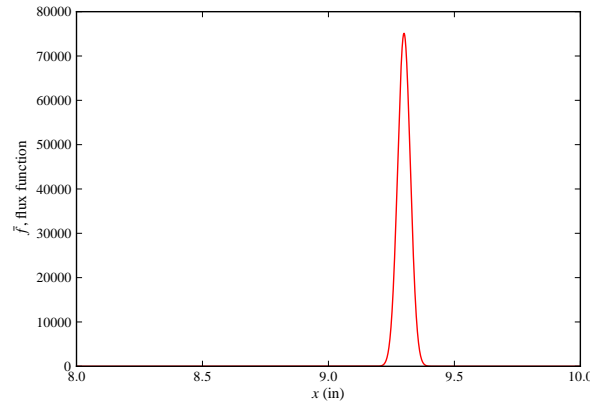


Figure 9. Analytical function representing the sharp, Gaussian laser heating applied to the front surface of the beam plotted over a small interval on the domain.

the internal energies of the Abaqus+GFEM^{gl} enriched global and *hp*-GFEM reference solutions, respectively.

The global domain in Abaqus was meshed with TET4 and, subsequently, TET10 elements. In each case, the global mesh remained constant, with uniform tetrahedral elements 0.5 inches in size (shown in Figure 8), corresponding to 100 dofs in the TET4 global mesh, and 441 dofs in the TET10 case. The local domain and corresponding enrichment zone also remained a constant size of $2 \times 0.5 \times 0.24$ inches surrounding the sharp flux, where uniform local polynomial order $p = 3$ was used. A series of mesh refinements was performed in the local problem, and global-local enrichments in all cases added a mere 20 dofs to the TET4 global problem and 91 dofs to the TET10 global problem; thus, only 20 (TET4) or 91 (TET10) pseudo-loads and pseudo-solutions were exchanged between FEM and GFEM solvers for each mesh refinement level.

For comparison, convergence results were also obtained using the optimal methodology, *hp*-GFEM. Each “equivalent” *hp*-GFEM solution resulted from successively refining the mesh near the localized laser heating, analogously to what is done in the GFEM^{gl} local problem, with a uniform polynomial order $p = 3$ throughout. Figure 10 illustrates the coarse global mesh used in Abaqus+GFEM^{gl} simulations compared to the very refined global mesh necessitated by *hp*-GFEM solutions.

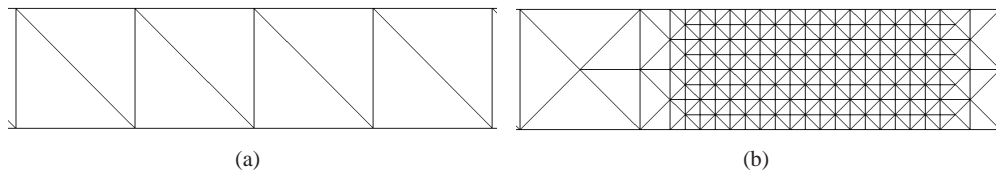


Figure 10. Global meshes corresponding to (a) Abaqus+GFEM^{gl} and (b) *hp*-GFEM. The enriched global problem in Abaqus+GFEM^{gl} consists of just 120 dofs for the TET4 mesh, or 532 dofs in the TET10 case, whereas the *hp*-GFEM global mesh with $p = 3$ and heavy localized mesh refinement results in a total of 189,290 dofs.

Figure 11 compares convergence of Abaqus+GFEM^{gl} against the *hp*-GFEM. Relative error in the energy norm, e_U^r , is plotted here against local domain mesh refinement (in Abaqus+GFEM^{gl}) or

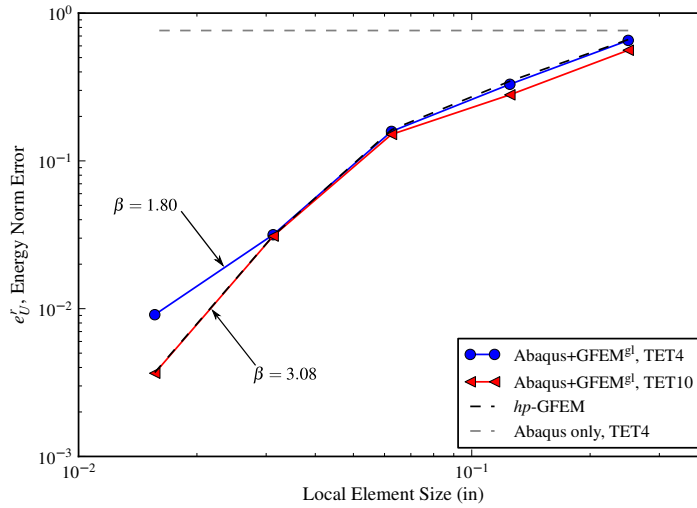


Figure 11. Convergence of non-intrusive GFEM^{gl} algorithm using FE solver Abaqus vs. *hp*-GFEM with respect to minimum local domain mesh size. Error is computed w.r.t. a reference solution with 736,990 dofs. Asymptotic convergence rates and error levels are nearly identical for Abaqus+GFEM^{gl} (TET10) and *hp*-GFEM. The dashed line shows relative error in the initial global solution, i.e., without any global-local enrichment, on a coarse TET4 mesh using coarse-scale numerical integration of the sharp laser flux (76% error).

localized global mesh refinement (in the *hp*-GFEM). It should be noted, however, that refinement was *only* performed in the local domain in the Abaqus+GFEM^{gl} case, and that the size of the enriched global problem remained exactly the same for all mesh refinement levels – 100+20 and 441+91 dofs for TET4 and TET10 meshes, respectively. Results from the non-intrusive GFEM^{gl} methodology are very similar to those obtained using the *hp*-GFEM itself. The convergence rate of the Abaqus+GFEM^{gl} TET10 case is quite near optimal, around the polynomial order of the approximation, $p = 3$. To illustrate the dramatic benefit of using the GFEM^{gl}, the figure also shows the error in the solution obtained from Abaqus using only the coarse initial global TET4 mesh with no global-local enrichment.

Point-wise error of the temperature field over the front surface of the beam with respect to the reference solution is shown in Figure 12. It is important to note that error of the local problem solution is large at the local domain boundary ($x = 8.5$ in). However, reanalysis of the global problem in Abaqus using this solution as an enrichment function improves the quality of the global solution even at locations where the local solution is not accurate. This is a key difference between this method and traditional global-local analysis [12, 24].

5.2.2. Effect of Special Treatment of the Sharp Loading Figure 13 illustrates the importance of using the GFEM^{gl} local problem information and high-order numerical quadrature rules available only in **GFE-S** to improve accuracy of the global load vector. Convergence results here are given for analyses run in which only coarse-scale computation of the sharp loading on the global mesh was used, and also for cases in which the recomputation procedure (cf. Section 4.2) was used. In both cases, the sharp loading was applied in the global domain. Thus, optimal convergence would

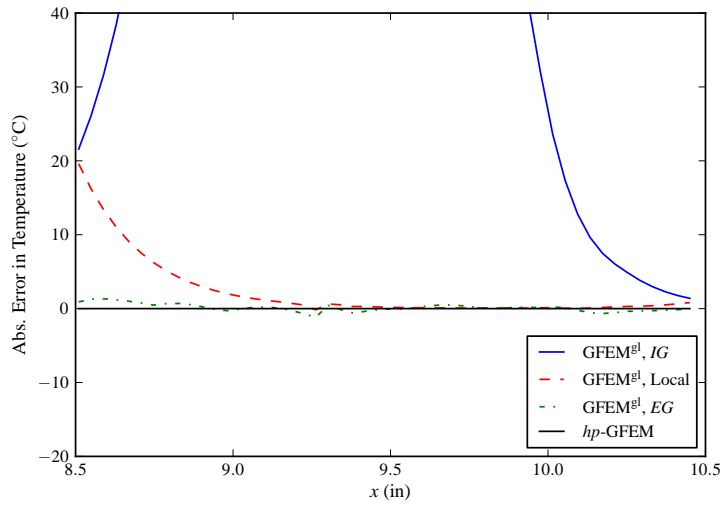


Figure 12. Point-wise error in temperature of GFEM^{gl} solutions with respect to the hp -GFEM reference solution. The local problem boundary is located at $x = 8.5$. Although error of the local problem solution is large at the boundary, the enriched global solution is of good quality.

in fact not nearly be possible without using this scheme for accurately recomputing the global load vector.

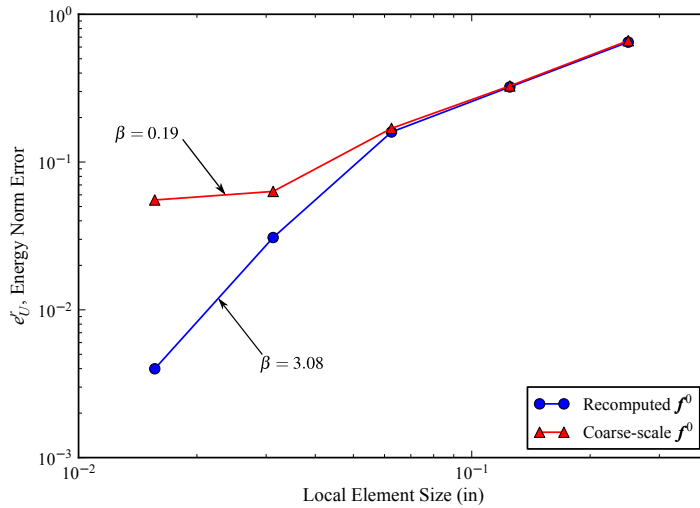


Figure 13. Effect of rebuilding the global load vector using GFEM solver with fine-scale computation of the sharp load on convergence behavior. Convergence in the energy norm is brought to a halt as the mesh is refined when inaccurate, coarse-scale computation of the sharp loading is performed because representation of the global load is poor, and thus the solution corresponding to global degrees of freedom is similarly unsatisfactory.

Furthermore, an investigation was performed to compare the accuracy of computation of the sharp Gaussian laser flux on a coarse mesh in Abaqus, versus coarse-scale computation of the load using a high-order numerical quadrature rule in the GFEM code, versus the recomputation

approach described in Section 4.2, where a high-order quadrature rule is paired with the use of highly-refined local domain elements for integration. The resulting internal energies and error levels are summarized in Table II. Results are shown for both initial and enriched global problems. Here, internal energy is computed as $U_h = 1/2 * B(u_h, u_h)$.

Table II. Error levels resulting from various approaches to computing the global sharp load vector on the coarse, global mesh. Abaqus and GFEM coarse-scale computations were performed on the initial global problem, using the initial, coarse mesh. A higher-order quadrature rule was used in the GFEM computation. Recomputation using highly-refined local elements was carried out in the last *IG* case. Enriched global results corresponding to 13 levels of local domain mesh refinement are also shown for each load computation case.

Method used	TET4 mesh		TET10 mesh	
	Int. Energy	e_U^r	Int. Energy	e_U^r
Abaqus, <i>IG</i> , coarse-scale	4.036×10^5	92.67%	3.539×10^6	48.84%
GFEM, <i>IG</i> , coarse-scale	5.672×10^5	89.53%	1.430×10^6	70.68%
GFEM, <i>IG</i> , fine-scale recomputation	5.568×10^5	89.73%	1.439×10^6	70.47%
Abaqus+GFEM ^{gl} , <i>EG</i> , coarse-scale	2.850×10^6	5.25%	2.860×10^6	2.95%
Abaqus+GFEM ^{gl} , <i>EG</i> , fine-scale recomputation	2.857×10^6	0.91%	2.857×10^6	0.37%

In the case of the initial global problem in Abaqus, while the TET4 mesh results seem reasonable compared to the **GFE-S** results, the energy of the TET10 analysis case is vastly overestimated, since the global load vector is also overestimated, and from this it is evident that computation of the sharp loading on a coarse mesh in Abaqus is unreliable. In enriched global results, on the other hand, it is shown that an accurate estimate of the actual global load vector is necessary to better approximate the solution corresponding to global dofs and obtain acceptable error levels.

Finally, Figure 14 illustrates the effects on the enriched global solution of applying versus *not* applying the coarse-scale, sharp load at all in the initial global problem. Here, the difference in

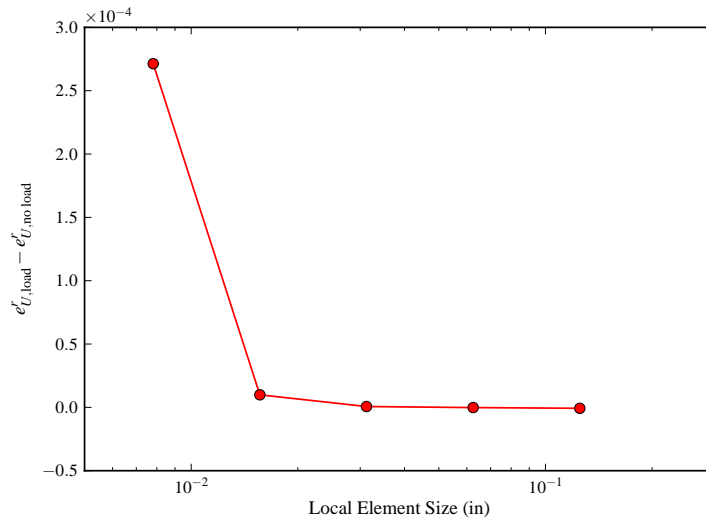


Figure 14. The difference in energy norm error of the enriched global problem solution between the case where a sharp loading is ($e_{U,load}^r$) versus is not ($e_{U,no load}^r$) applied on the coarse, initial global TET10 beam mesh is shown. As the local mesh reaches high levels of refinement, pollution error due to the sharp *IG* problem flux dominates the case where the initial sharp load is applied.

relative error in the energy norm of the enriched global problem solution between the case where the sharp loading is and is not applied is plotted against local domain mesh refinement, with the enriched global problem remaining exactly the same size. It can be observed that as discretization error is reduced as a result of high levels of local mesh refinement, the effect of pollution error (and some integration error) from applying the sharp flux on the coarse, initial mesh dominates. As shown in Table II, however, the initial solution on a coarse mesh may not improve as a result of decreased integration error.

5.3. Large Stiffened Panel

In the final sample problem presented, an attempt is made to demonstrate the adaptability of the proposed non-intrusive methodology in handling a variety of different multiscale analysis cases for the same global model of interest. This nice feature of the non-intrusive implementation allows the user to insert localized problem information anywhere within the global model, using capabilities of the GFEM^{gl} alone to handle these localized features, and not requiring any changes to the model itself.

The problem of interest is taken to be a representative, computationally large stiffened panel section of dimensions $600 \times 600 \times 3$ cm with stiffener beams of cross-section 20×20 cm attached underneath the panel near each edge. The panel was subjected to a constant, steady-state surface heat flux as well as intense, localized Gaussian laser heatings at various locations. The sharp, analytical flux function used to represent the localized heating is plotted in Figure 15.

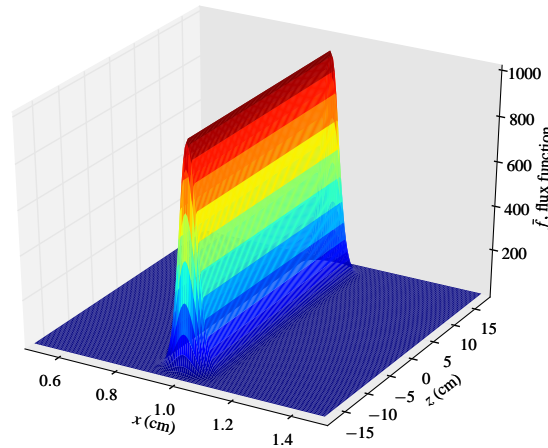


Figure 15. A sample depiction of the intense Gaussian laser flux applied to the top surface of the stiffened panel, plotted over a very small interval on the domain. The laser heating covers an area of approximately 0.1×30 cm on the global domain in all cases.

Adopting the partitioning strategy for \tilde{f}^0 discussed in Section 4.2, localized laser heatings were introduced *only* in local and enriched global problems, and in all cases, the very same Abaqus global problem was used, upon which only the smooth (constant) portion of the surface heating was applied. Thus, **GFE-S** alone was tasked with handling all computations involving localized effects. In each case, the sharp flux was placed slightly to the left or right, or above or below an element edge, but never directly in line with one, so that the quality of the GFEM^{gl} solution is dependent mostly upon the quality of the global-local enrichment functions. This was done to represent the

most general possible analysis case, where the mesh may not necessarily be designed to account for the location and configuration of the sharp loading.

The global domain was meshed with linear TET4 elements, resulting in 45,534 initial global degrees of freedom. Temperature boundary conditions of 0°C were applied to the left and right faces of the panel with the boundary insulated elsewhere. The three chosen locations for the laser heating are illustrated in Figure 16, along with the general panel geometry. As usual, in each case,

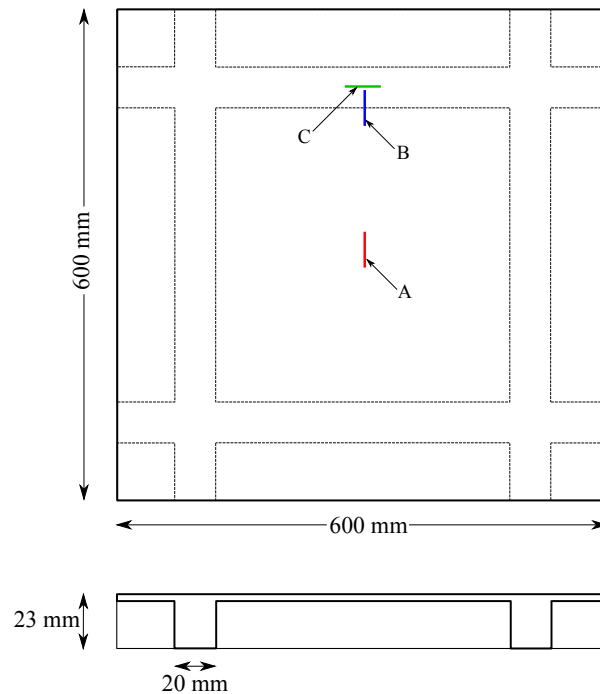


Figure 16. Geometry of the stiffened panel and locations of applied intense, localized surface laser heating: location A, around the center of the panel; location B, over a panel-stiffener junction; and location C, along the axis of an edge stiffener. *Note: drawing not to scale.*

a local domain was chosen to cover a small neighborhood of the locally-applied, sharp laser flux. Necessary mesh refinements and polynomial enrichments were taken care of by **GFE-S** only in the local problem.

The initial global temperature field corresponding to a smooth, constant flux over the top surface of the panel is shown in Figure 17. Note that because the global model remains unchanged for each sharp load case (only the constant surface flux is applied each time), this same initial global solution may be utilized repeatedly for multiple sharp flux cases as needed, saving some computational cost when many analysis cases are required. Computational data for each sharp flux case is listed in Table III. The corresponding enriched global solutions exhibiting the sharp solution characteristics are shown graphically in Figures 18, 19, and 20.

While the argument may be made that meshes consisting of two-dimensional plate and one-dimensional beam elements may be used in the elastic analysis of structures of a similar nature to this plate in order to reduce computational cost and simplify challenging three-dimensional meshing requirements, in general these types of meshes may *not* be used in heat transfer analyses. As illustrated by Figure 19, the sharp flux of interest applied near the edge of the panel exhibits

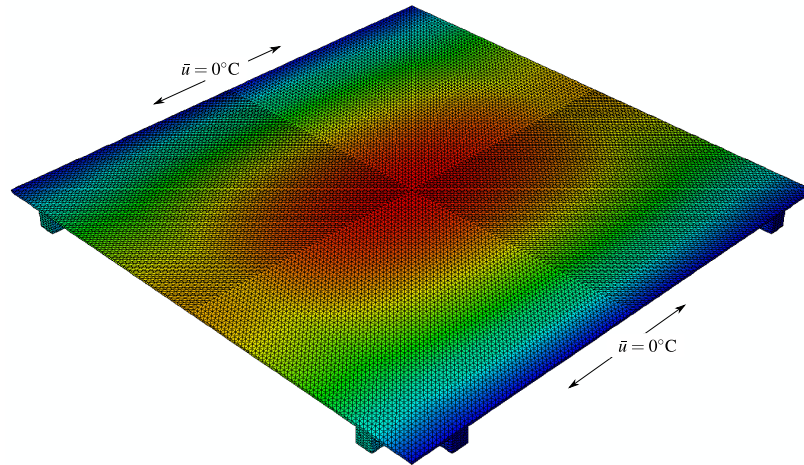


Figure 17. Temperature field on a stiffened panel from the Abaqus initial global solution (only a constant surface flux applied). The Abaqus TET4 mesh and temperature boundary conditions are also illustrated here.

Table III. Abaqus+GFEM^{gl} numerical results for the stiffened panel problem. The tremendous savings stemming from use of the GFEM^{gl} methodology with respect to additional enriched global problem degrees of freedom is evident here. While *IG* and local solutions underestimate the true solution (drastically in some cases), the enriched global problem is much better able to capture true, sharp solution characteristics.

Flux Loc.	dofs			Internal Energy		
	<i>IG</i>	Local	<i>EG</i>	<i>IG</i>	Local	<i>EG</i>
A	45,534	46,680	45,534 + 55	4.389×10^6	8.912×10^4	5.575×10^6
B	45,534	197,000	45,534 + 85	4.389×10^6	2.455×10^7	1.188×10^8
C	45,534	342,520	45,534 + 85	4.389×10^6	1.850×10^7	1.145×10^8

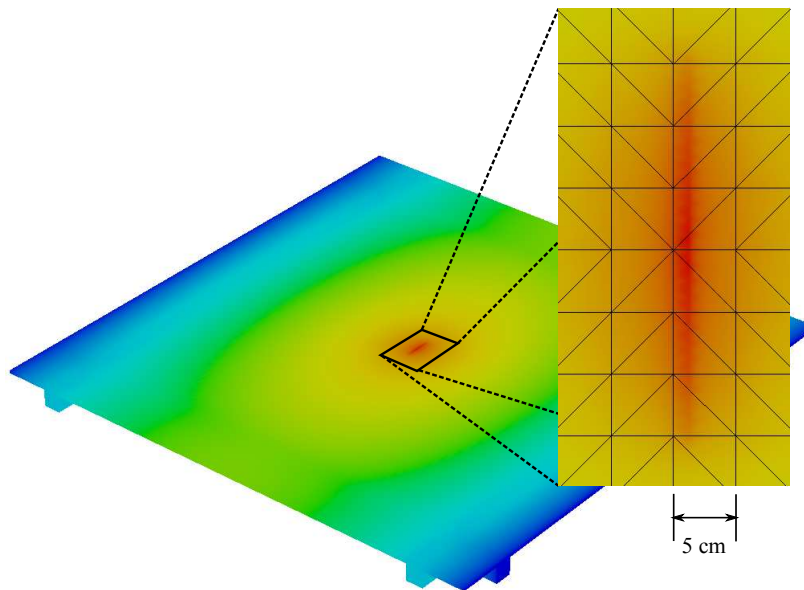


Figure 18. Enriched global problem temperature field on a stiffened panel from the Abaqus+GFEM^{gl} solution, sharp flux location A, on the center of the panel. A zoom-in on the sharp feature of interest is also shown. The sharp flux is slightly skewed to the right of the line of a global mesh edge, so that the quality of the sharp global solution features relies strongly on the quality of global-local enrichments.

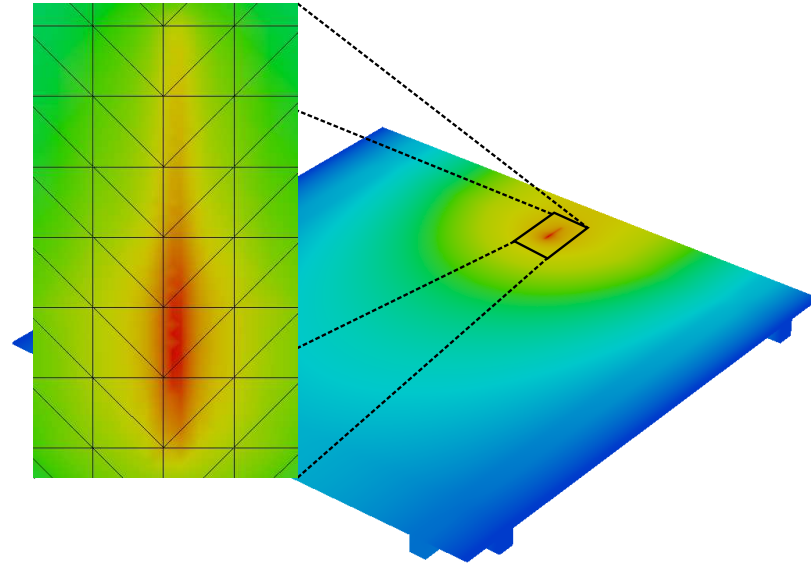


Figure 19. Temperature field on a stiffened panel for sharp flux location B. Here, the laser heating intersects a stiffener beam near the edge of the panel. The effectiveness of the 3D structural-scale model in accounting for through-the-thickness effects due to the stiffener is evident here. The sharp flux is slightly skewed to the right of the line of a global mesh edge, so that the quality of the sharp global solution features relies strongly on the quality of global-local enrichments.

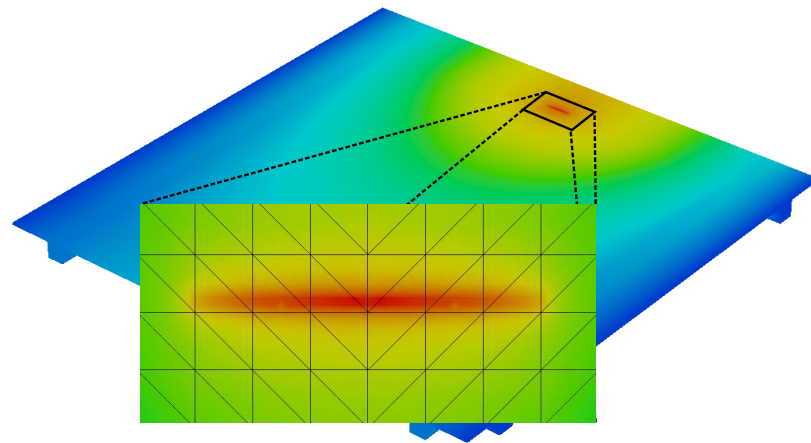


Figure 20. Temperature field on a stiffened panel corresponding to sharp flux location C, in which case the flux is concentrated entirely over a stiffener beam. The sharp flux is applied slightly above the line of a global mesh edge, so that the quality of the sharp global solution features relies strongly on the quality of global-local enrichments.

significant through-the-thickness effects due to the presence of a stiffener beam underneath. Thus, a full three-dimensional analysis is crucial for predicting actual thermal behavior.

It should be noted that in problems of this nature – large, representative, structural-scale problems which necessitate fine global meshes in order to accurately represent complicated geometry – it would be computationally infeasible to perform *hp*-adaptivity on the global mesh to capture local solution features. The GFEM^{gl} circumvents this issue entirely, since *hp*-adaptivity need only be performed on a comparatively small subset of the global domain. Without the capability for such *hp*-adaptivity, coarse scale computation of the sharp flux on the global mesh might provide limited

prediction of the intense temperature field surrounding the sharp flux on the panel, but the localized sharp solution characteristics could not be captured satisfactorily.

The immense flexibility of the methodology is also demonstrated by the fact that a localized feature can be placed arbitrarily within the global problem without making any changes to the global model itself, which, as aforementioned, is a significant limitation of approaches using single element-based enrichments. Not only can different localized features be inserted into the global problem of interest, but any combination of localized effects can similarly be considered simultaneously via the extraction and solution of multiple local problems from the global domain.

Finally, this problem was used to study the computational effort required for the proposed methodology in terms of CPU time. Table IV shows total factorization and solution times for each panel flux case solved using *hp*-GFEM discretizations. Table V gives corresponding factorization and solution times using Abaqus+GFEM^{gl}. The non-intrusive GFEM^{gl} framework has a distinct advantage over *hp*-GFEM, since the initial global problem phase needs to be solved only once for all three load cases. Using *hp*-GFEM, on the other hand, the entire global stiffness matrix must be reassembled and re-factorized for each analysis case. CPU times required for factorization

Table IV. *hp*-GFEM CPU times for factorization and solution of the stiffened panel problem.

Flux Loc.	CPU Time (s)	Internal Energy
A	15.86	5.577×10^6
B	354.99	1.195×10^8
C	999.02	1.149×10^8
<i>Total:</i>	<i>1369.87</i>	

Table V. Abaqus+GFEM^{gl} CPU times for factorization and solution of the stiffened panel problem.

Flux Loc.	<i>IG</i>	CPU Time (s)			Total	Internal Energy
		Local	Abaqus nRHS	<i>EG</i> GFEM static cond.		
A		15.14	34.70	0.10	50.51	5.570×10^6
B	0.67	320.51	54.02	0.18	375.2	1.188×10^8
C		947.77	51.25	0.19	999.69	1.145×10^8
<i>Total:</i>					<i>1425.40</i>	

and solution are comparable for both cases (4% difference for this problem). Due to a limitation of Abaqus heat transfer analysis, however, the stiffness matrix of the coarse-scale global problem was re-factorized for each pseudo-load in the enriched global problem solution phase. Thus, computation of pseudo-solutions in Abaqus (Abaqus nRHS in the above table) would be substantially faster were this limitation of heat transfer analysis resolved – instead of re-factorizing the global matrix, a forward/backward substitution would be sufficient. For reference, each global factorization in Abaqus takes about 0.6 seconds, while each forward/backward substitution takes only 0.1 seconds, though the difference between the two grows quickly with problem size. This study shows that despite these limitations, the proposed methodology brings the benefits of *hp*-adaptivity to existing software like Abaqus at only a modest CPU overhead.

The data in Table V shows that the main overhead of the proposed methodology with respect to a standard global-local analysis comes from pseudo-solution computations in Abaqus, while the

CPU time required for static condensation of enrichment degrees of freedom is minimal. The total CPU time for a standard global-local analysis of this problem is about 1284.76 seconds. While this is about a 10% reduction in CPU time compared to the proposed algorithm, the error of the local solutions is, in general, significantly greater than in the method proposed here as discussed earlier and demonstrated in, e.g., [15]. In contrast, the internal energy computed with Abaqus+GFEM^{gl} differs by less than 1% from the *hp*-GFEM values.

6. CONCLUSIONS

The methods demonstrated have been shown to provide *hp* finite element discretizations to a standard, commercial FEA code without the need for code modifications, via the generalized finite element method with global-local enrichment functions (GFEM^{gl}). Moreover, the proposed non-intrusive approach has been successfully applied to problems with localized features of interest, subjected to very intense, localized thermal loadings, and it demonstrates several characteristics which are very effective in handling problems of this nature. For instance,

- (i) the GFEM^{gl} introduces adaptive mesh refinement and high-order polynomial approximations on only a small, local subset of the structural-scale, global domain, resulting in only a few extra degrees of freedom, which is especially beneficial when the global model of interest is extremely large, as in the presence of complicated geometry;
- (ii) the method improves upon traditional global-local and submodeling techniques for multiscale problems already available in current FEA platforms by using local solutions as enrichment functions on the global domain, rather than the final solution, greatly reducing the effect of important local error due to poor boundary conditions;
- (iii) even with existing limitations on reuse of the global stiffness matrix, the CPU overhead of the proposed methodology is modest. The benefits of *hp*-adaptivity includes reduced user time in model preparation which in practice is much more costly than CPU time;
- (iv) extraction of local problems and adaptive mesh refinement may be automated by *a posteriori* error estimates on initial coarse-scale, global solutions, minimizing the need for user intervention in the GFEM code;
- (v) adaptive mesh refinement can be performed in a local domain even when a higher-order partition of unity (higher-order element) is used in the global model, which has been demonstrated here using a quadratic tetrahedral global mesh, but a similar methodology has also been successfully applied to 8-, 20-, and 27-noded hexahedral elements [48];
- (vi) the methodology provides a means of computing extremely sharp, localized loadings, even on a coarse-scale mesh – and it has been shown that attempting to compute sharp loadings similarly in a standard FE solver alone may not be possible without incurring very large errors in the solution.

Therefore, the non-intrusive implementation introduced here features several capabilities which have not been identified by the authors as being available in other comparable methodologies:

- (i) the method delivers *optimal convergence* as well as error levels competitive with standard *hp*-FEM approximations, which have been recognized as optimal for the class of problems investigated in this paper [9, 10];
- (ii) numerical experiments demonstrate that the method delivers high accuracy on local quantities like point-wise temperature fields at regions with large thermal gradients. This has also been demonstrated for elasticity [23] and transient heat transfer problems [49];
- (iii) the implementation allows for arbitrarily many localized features of interest to be introduced to the very same global structural model via the extraction and solution of multiple local problems, providing excellent flexibility when many analysis cases are necessary.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the contributions of the Midwest Structural Sciences Center (MSSC) at the University of Illinois at Urbana-Champaign. The Center is supported by the U.S. Air Force Research Laboratory, Air Vehicles Directorate, under contract number FA8650-06-2-3620.

A. IMPLEMENTATION DETAILS OF THE NON-INTRUSIVE ALGORITHM

The following gives particulars of the procedure used to non-intrusively implement the GFEM^{gl} in a standard FE solver – in this case, Abaqus, which was used for all numerical examples given – which is described in general in Section 4.

Each analysis using the non-intrusive implementation of the GFEM^{gl} in Abaqus (Abaqus+GFEM^{gl}) is orchestrated by **GFE-S**, which executes continuously throughout the process. The exchange of the initial global solution, pseudo-loads, and pseudo-solutions between Abaqus and **GFE-S** is facilitated by a converter code written in a combination of Python (the scripting language of Abaqus/CAE [50, 51]) and C++. Because the two codes are otherwise isolated from each other, identical global models, or job files, must be present for both Abaqus and **GFE-S**. Two model files must be written for Abaqus: one for the initial global analysis (henceforth `abaqus.inp`), and another for the enriched global problem, which will contain pseudo-loads (`abaqus_nrhs.inp`). Abaqus/CAE is utilized here for its `.odb` binary output database format, which contains all user-requested output in a very conveniently organized, hierarchical data structure and can be read and converted directly by an Abaqus Python script (referred to here as the Python converter code).

A.1. Initial Global Problem

The procedure for the initial global problem step is as follows:

- (i) Call Abaqus from **GFE-S** to execute model file `abaqus.inp` via Python script.
- (ii) When problem is done executing, execute Python converter code to write initial global results \underline{u}^0 from the `.odb` file to output file readable by **GFE-S**.
- (iii) Read initial solution in **GFE-S**.

A.2. Local Problem(s)

Once the initial global problem solution is read in **GFE-S**, execution continues, and local problems can subsequently be extracted and solved, as specified. The procedure for a local problem is as follows:

- (i) Extract a local domain Ω_L (selected by the user or by *a posteriori* error estimates on u^0), and apply initial solution u^0 from Abaqus as a Dirichlet boundary condition.
- (ii) Perform *hp*-adaptivity as necessary and solve local problem.
- (iii) Using local solution u_L , compute global-local shape functions ϕ_α^{gl} as in Equation (6), and use these to assemble \mathbf{K}^{gl} and \mathbf{f}^{gl} corresponding to global-local dofs, and $\mathbf{K}^{0,gl}$, pseudo-loads to be passed to Abaqus, as defined in Section 4.1.
- (iv) Recompute total global load vector \mathbf{f}^0 using refined local elements for integration.
- (v) Write out pseudo-loads and recomputed global load vector (including the sharp load features) to a file.

A.3. Enriched Global Problem

The enriched global problem requires multiple communications between **GFE-S** and Abaqus. In this step, because in general multiple pseudo-loads $\mathbf{K}^{0,gl}$ will be present, Abaqus's built-in capability to solve multiple right hand sides as part of the same job is utilized. The enriched global procedure is done as follows:

- (i) Read pseudo-loads and global load vector from **GFE-S** in converter code, and write each right hand side therein to a separate analysis step in an input file stub readable by Abaqus.
- (ii) Call Abaqus and execute enriched global model `abaqus_nRHS.inp` (written prior to the analysis), including file containing multiple right hand sides (*not* known prior to the analysis) written above.
- (iii) When problem is done executing, execute Python converter code to write resulting pseudo-solutions $\mathbf{S}^{0,gl}$ and recomputed global solution $\underline{\mathbf{u}}^0$ from the `.odb` file to output file readable by **GFE-S**.
- (iv) Read output file in **GFE-S** and compute the global degrees of freedom, $\underline{\mathbf{u}}^0$ as in Equation (10), the Schur complement of the global stiffness matrix, $\hat{\mathbf{K}}^{gl}$ as in Equation (11), as well as $\hat{\mathbf{f}}^{gl}$.
- (v) Compute the solution for global-local degrees of freedom $\underline{\mathbf{u}}^{gl}$ as in Equation (11) and total solution u^E as in Equation (7).

After the final step, any post-processed quantities which were requested by the user in **GFE-S** are computed, and Abaqus initial global and **GFE-S** local and enriched global solutions may also be visualized. Execution of the program completes, and all requested output from the enriched global solution is available to the user from **GFE-S**.

REFERENCES

1. Thornton EA. Thermal structures – four decades of progress. *Journal of Aircraft* 1992; **29**(3):485–498.
2. Wieting AR. Experimental study of shock wave interference heating on a cylindrical leading edge 1987. NASA Technical Memorandum 100484.
3. Moselle JR, Wieting AR, Holden M, Glass C. Studies of aerothermal loads generated in regions of shock/shock interaction in hypersonic flow 1988. AIAA-88-0477.
4. Friedmann PP, Powell KG, Mcnamara JJ, Thuruthimattam BJ, Bartels R. Hypersonic aerothermoelastic studies for reusable launch vehicles 2004. AIAA 2004-1590.
5. Friedmann PP, Powell KG, Mcnamara JJ, Thuruthimattam BJ, Bartels R. Three-dimensional aeroelastic and aerothermoelastic behavior in hypersonic flow 2005. AIAA 2005-2175.
6. D'Ambrosio D. Numerical prediction of laminar shock/shock interactions in hypersonic flow. *Journal of Spacecraft and Rockets* 2003; **40**(2):153–161.
7. Thornton EA, Wieting AR, Morgan K. Application of integrated fluid-thermal-structural analysis methods. *Journal of Thin-Walled Structures* 1991; **11**:1–23.
8. Turner TL, Ash RL. Analysis of the thermal environment and thermal response associated with thermal-acoustic testing 1990. AIAA-90-0975-CP.

9. Szabo B, Babuška I. *Finite Element Analysis*. John Wiley and Sons: New York, 1991.
10. Rachowicz W, Oden JT, Demkowicz L. Toward a universal h - p adaptive finite element strategy, Part 3. Design of h - p meshes. *Computer Methods in Applied Mechanics and Engineering* 1989; **77**:181–212.
11. Babuška I, Strouboulis T. *The Finite Element Method and its Reliability*. Numerical Mathematics and Scientific Computation, Oxford Science Publications: New York, USA, 2001.
12. O'Hara P, Duarte C, Eason T. Generalized finite element analysis of three-dimensional heat transfer problems exhibiting sharp thermal gradients. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**(21–26):1857–1871, doi:10.1016/j.cma.2008.12.024.
13. Felippa C. Introduction to finite element methods. 2004. Course Notes. Department of Aerospace Engineering Sciences, University of Colorado at Boulder. Available at <http://www.colorado.edu/engineering/Aerospace/CAS/courses.d/IFEM.d>.
14. Noor A. Global-local methodologies and their applications to nonlinear analysis. *Finite Elements in Analysis and Design* 1986; **2**:333–346.
15. Kim DJ, Duarte C, Pereira J. Analysis of interacting cracks using the generalized finite element method with global-local enrichment functions. *Journal of Applied Mechanics* 2008; **75**(5):051107, doi:10.1115/1.2936240.
16. Whitcomb J. Iterative global/local finite element analysis. *Computers and Structures* 1991; **40**:1027–1031.
17. Jara-Almonte CC, Knight CE. The specified boundary stiffness/force sbsf method for finite element subregion analysis. *International Journal for Numerical Methods in Engineering* 1988; **26**(7):1567–1578, doi:10.1002/nme.1620260708.
18. Hirai I, Uchiyama Y, Mizuta Y, Pilkey W. An exact zooming method. *Finite Elements in Analysis and Design* 1985; **1**:61–69.
19. Hirai I, Wang B, Pilkey W. An efficient zooming method for finite element analysis. *International Journal for Numerical Methods in Engineering* 1984; **20**:1671–1683.
20. Mao K, Sun C. A refined global-local finite element analysis method. *International Journal for Numerical Methods in Engineering* 1991; **32**:29–43.
21. Duarte C, Kim DJ, Babuška I. Chapter: A global-local approach for the construction of enrichment functions for the generalized fem and its application to three-dimensional cracks. *Advances in Meshfree Techniques, Computational Methods in Applied Sciences*, vol. 5, Leitão V, Alves C, Duarte C (eds.), Springer: The Netherlands, 2007. ISBN 978-1-4020-6094-6.
22. Duarte C, Kim DJ. Analysis and applications of a generalized finite element method with global-local enrichment functions. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(6–8):487–504, doi:10.1016/j.cma.2007.08.017.
23. Kim DJ, Pereira J, Duarte C. Analysis of three-dimensional fracture mechanics problems: A two-scale approach using coarse generalized FEM meshes. *International Journal for Numerical Methods in Engineering* 2010; **81**(3):335–365, doi:10.1002/nme.2690.
24. Gupta V, Kim DJ, Duarte C. Analysis and improvements of the global-local enrichment functions for the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2011; To be submitted.
25. Babuška I, Caloz G, Osborn J. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM Journal on Numerical Analysis* 1994; **31**(4):945–981.
26. Babuška I, Melenk J. The partition of unity method. *International Journal for Numerical Methods in Engineering* 1997; **40**:727–758.
27. Duarte C, Babuška I, Oden J. Generalized finite element methods for three dimensional structural mechanics problems. *Computers and Structures* 2000; **77**:215–232.
28. Oden J, Duarte C, Zienkiewicz O. A new cloud-based hp finite element method. *Computer Methods in Applied Mechanics and Engineering* 1998; **153**:117–126.
29. Strouboulis T, Capps K, Babuška I. The generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:4081–4193.
30. Melenk J, Babuška I. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:289–314.
31. Duarte C, Oden J. Hp clouds—A meshless method to solve boundary-value problems. *Technical Report 95-05*, TICAM, The University of Texas at Austin May 1995.
32. Duarte C, Oden J. An hp adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:237–262.
33. Duarte C, Oden J. Hp clouds – An hp meshless method. *Numerical Methods for Partial Differential Equations* 1996; **12**:673–705.
34. Duarte C. *The hp Cloud Method*. PhD dissertation, The University of Texas at Austin December 1996. Austin, TX, USA.

35. Moës N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering* 1999; **46**:131–150.
36. Belytschko T, Gracie R, Ventura G. A review of extended/generalized finite element methods for material modeling. *Modelling and Simulations in Materials Science and Engineering* 2009; **17**:24pp. [Http://dx.doi.org/10.1088/0965-0393/17/4/043001](http://dx.doi.org/10.1088/0965-0393/17/4/043001).
37. Duarte C, Hamzeh O, Liszka T, Tworzydło W. A generalized finite element method for the simulation of three-dimensional dynamic crack propagation. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**(15–17):2227–2262, doi:10.1016/S0045-7825(00)00233-4.
38. Simone A, Duarte C, van der Giessen E. A generalized finite element method for polycrystals with discontinuous grain boundaries. *International Journal for Numerical Methods in Engineering* 2006; **67**(8):1122–1145, doi:10.1002/nme.1658.
39. Diamantoudis A, Labeas G. Stress intensity factors of semi-elliptical surface cracks in pressure vessels by global-local finite element methodology. *Engineering Fracture Mechanics* 2005; **72**:1299–1312.
40. Giner E, Sukumar N, Tarancón JE, Fuenmayor FJ. An Abaqus implementation of the extended finite element method. *Engineering Fracture Mechanics* Feb 2009; **76**(3):347–368, doi:10.1016/j.engfracmech.2008.10.015.
41. Shi J, Lua J, Waisman H, Liu P, Belytschko T, Sukumar N, Liang Y. X-FEM toolkit for automated crack onset and growth prediction. *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Schaumburg, IL, USA, 2008; 1–22. AIAA 2008-1763.
42. Shi J, Chopp D, Lua J, Sukumar N, Belytschko T. Abaqus implementation of extended finite element method using a level set representation for three-dimensional fatigue crack growth and life predictions. *Engineering Fracture Mechanics* Sep 2010; **77**(14):2840–2863, doi:10.1016/j.engfracmech.2010.06.009.
43. Xu Y, Yuan H. Computational modeling of mixed-mode fatigue crack growth using extended finite element methods. *International Journal of Fracture* Aug 2009; **159**(2):151–165, doi:10.1007/s10704-009-9391-y.
44. Xu Y, Yuan H. Computational analysis of mixed-mode fatigue crack growth in quasi-brittle materials using extended finite element methods. *Engineering Fracture Mechanics* Jan 2009; **76**(2):165–181, doi:10.1016/j.engfracmech.2008.08.011.
45. Gendre L, Allix O, Gosselet P. A two-scale approximation of the Schur complement and its use for non-intrusive coupling. *International Journal for Numerical Methods in Engineering* 2011; doi:10.1002/nme.
46. Gendre L, Allix O, Gosselet P, Comte F. Non-intrusive and exact global/local techniques for structural problems with local plasticity. *Computational Mechanics* Feb 2009; **44**(2):233–245, doi:10.1007/s00466-009-0372-9.
47. Bordas S, Moran B. Enriched finite elements and level sets for damage tolerance assessment of complex structures. *Engineering Fracture Mechanics* 2006; **73**:1176–1201.
48. O'Hara P, Duarte C, Eason T, Garzon J. Efficient analysis of transient heat transfer problems exhibiting sharp thermal gradients 2011. Submitted for publication.
49. O'Hara P, Duarte C, Eason T. Transient analysis of sharp thermal gradients using coarse finite element meshes. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**(5–8):812–829, doi:10.1016/j.cma.2010.10.005.
50. Dassault Systèmes Simulia Corporation, Providence, RI, USA. *Abaqus Scripting User's Manual* 2010.
51. Dassault Systèmes Simulia Corporation, Providence, RI, USA. *Abaqus Scripting Reference Manual* 2010.