

Comparison of multimesh hp -FEM to interpolation and projection methods for spatial coupling of reactor thermal and neutron diffusion calculations

Lenka Dubcova^a, Pavel Solin^{a,b,*}, Glen Hansen^c, HyeonKae Park^c

^a*Institute of Thermomechanics, Dolejskova 5, Prague, CZ 18200.*

^b*Department of Mathematics and Statistics, University of Nevada - Reno, 1664 N. Virginia St., Reno, NV 89557-0208.*

^c*Multiphysics Methods Group, Idaho National Laboratory, P.O. Box 1625, Idaho Falls, ID 83415-3840.*

Abstract

Multiphysics solution challenges are legion within the field of nuclear reactor design and analysis. One major issue concerns the coupling between heat and neutron flow (neutronics) within the reactor assembly. These phenomena are usually very tightly interdependent, as large amounts of heat are quickly produced with an increase in fission events within the fuel, which raises the temperature that affects the neutron cross section of the fuel. Furthermore, there typically is a large diversity of time and spatial scales between mathematical models of heat and neutronics. Indeed, the different spatial resolution requirements often lead to the use of very different meshes for the two phenomena. As the equations are coupled, one must take care in exchanging solution data between them, or significant error can be introduced into the coupled problem. We propose a novel approach to the discretization of the coupled problem on different meshes based on an adaptive multimesh higher-order finite element method (hp -FEM), and compare it to popular interpolation and projection methods. We show that the multimesh hp -FEM method is significantly more accurate than the interpolation and projection approaches considered in this study.

Keywords: reactor multiphysics simulation, coupled diffusion, multiphysics error
2000 MSC: 65Z05, 65N30, 65N22, 80M10

1. Introduction

Nuclear reactor modeling and simulation is filled with multiphysics modeling challenges. Inside of a light water reactor, coolant flows through fuel bundles to remove heat that is used to generate electric power. This coolant flow can cause vibration and wear of the fuel rods which reduce their life. Further, the coolant moderates the speed of neutrons in the reactor, which affects the fission efficiency within the fuel, which changes the amount of heat produced that needs to be removed by the coolant. As such, calculations that describe the flow of neutrons (neutronics), coolant (fluid mechanics), and heat are coupled within the reactor. Indeed, the coupling may be tight, which requires that care be taken in the solution of the equation systems that describe the phenomena. Further, one must be cognizant of numerical error that may be introduced into the solution of the coupled system, especially if a large number of time steps are needed for the solution.

Generally, each of the phenomena to be modeled inside of the reactor have different spatial discretization requirements [7]. For example, the coolant mesh requires a refined anisotropic mesh near the fuel to resolve the turbulent boundary layer to correctly compute the heat flow from the fuel into the coolant, as seen in the gas-cooled reactor shown in Fig. 1. Likewise, the neutronics equations require resolution to capture effects at material interfaces and discontinuities. If a single mesh spanning the entire domain and all physics phenomena were used, it would need to be fine enough at every location to meet the discretization needs of

*Corresponding author

Email addresses: `dubcova@centrum.cz` (Lenka Dubcova), `solin@unr.edu` (Pavel Solin), `Glen.Hansen@inl.gov` (Glen Hansen), `Ryosuke.Park@inl.gov` (HyeonKae Park)

the most demanding physics at that location. Further, as the length and time scales of the various physics components may vary greatly during a transient simulation, this mesh would only be appropriate for a certain period in time. This leads one to conclude that to capture individual behavior of the solution components more efficiently, different physics should be discretized on their individual meshes, and an adaptive method should be used to refine each one based on the needs of the physics it supports.

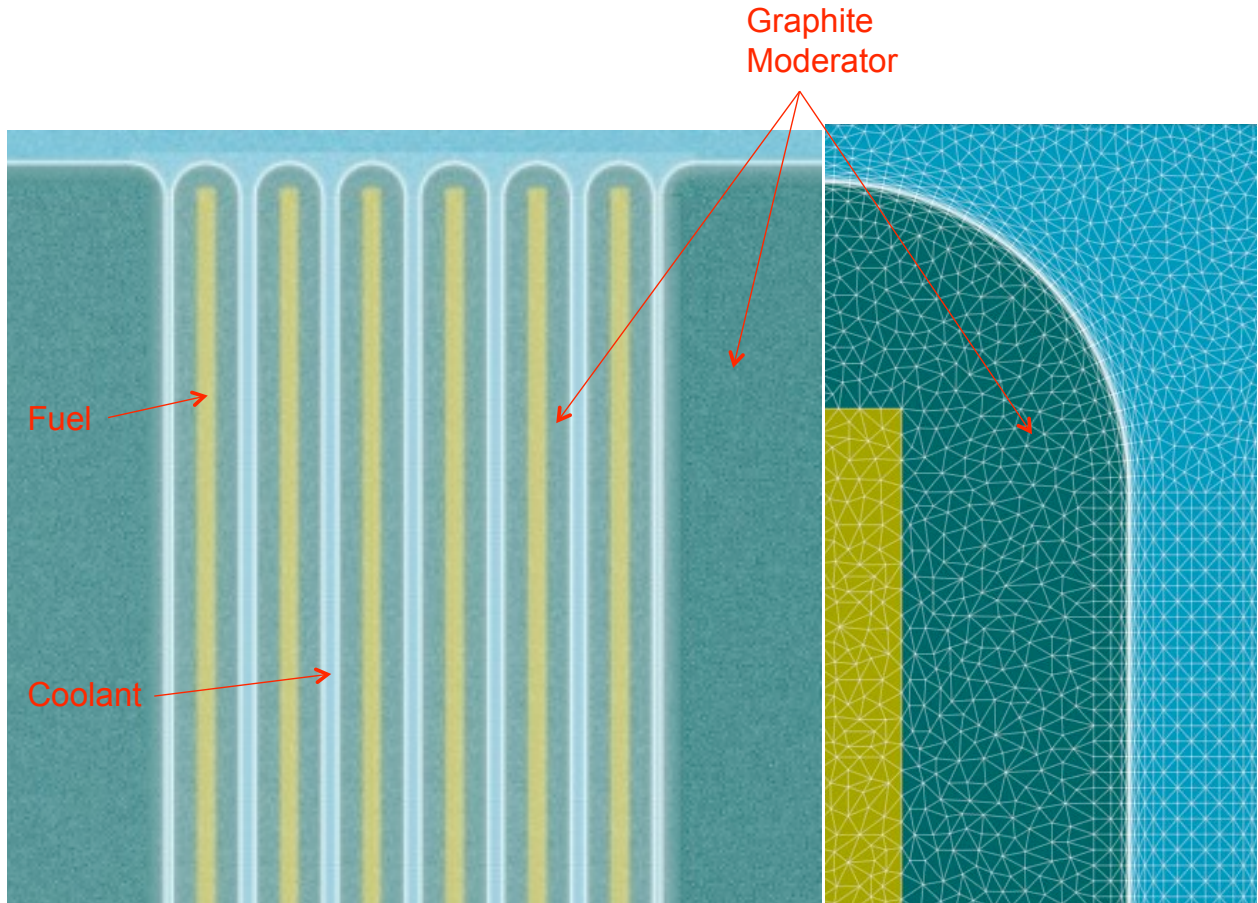


Figure 1: Triangular mesh generated on a cross-section of a gas-cooled reactor core. The gas coolant flows between the sections of the graphite moderator that contain the reactor fuel. Note that graded anisotropic mesh is used where the coolant meets the surface of the moderator. Substantial refinement is needed in this area as the correct resolution of the coolant flow in the boundary layer is crucial for the accurate calculation of both the heat flux and forces between the coolant and the moderator.

Traditionally, reactor multiphysics problems have been analyzed by dividing them into several distinct processes, one for each physical phenomena of interest, and addressing each phenomena independently using existing monophysics codes. In reality, each component of the solution is often strongly coupled to the others, necessitating interaction between each of the models (and codes). Traditional coupling paradigms rely on solving the different physics in a loosely coupled fashion, a technique mathematically described as operator splitting (OS). This is often implemented in a simple black-box fashion, where the output of one code serves as input of another code, using various methods of data transfer between the non-matching meshes. Iteration is used between the codes to converge the overall problem. This approach, while attractive because it employs existing codes and user experience, may exhibit accuracy and stability issues. The different codes employed typically use different meshes that necessitate a data exchange at each iteration. These data exchanges may be quite challenging to perform accurately, as it is often necessary to conduct an expensive geometric

intersection of the meshes to host the solution of one component on the other’s mesh, and vice versa [12]. For three-dimensional problems on complex reactor geometry, the generation of individual meshes can be very time consuming and expensive [7]. Secondly, it is often challenging to perform the data transfer so that conservation of important quantities (mass, momentum, energy) of the solution is achieved, originating from the fact that coupling terms between the various physics components are dealt with in an inconsistent fashion. These issues are discussed in some detail in the study by Jiao and Heath [8].

This paper proposes a multimesh *hp*-FEM method that employs an assembly process to discretize multi-physics coupled problems in a monolithic way without employing operator splitting or data transfer between meshes. Each solution component is approximated on a different mesh that is obtained adaptively to best fit the specific physics requirements. The multimesh *hp*-FEM has been previously employed to coupled thermo-mechanical models [6], coupled moisture and heat transfer models [14], and coupled electromagnetics-thermal models [15].

This study explores the use of adaptive multimesh *hp*-FEM for computational nuclear engineering problems. It begins with Section 2 that describes the algorithm for automatic adaptation of *hp*-meshes with arbitrary-level hanging nodes and demonstrates advantages of *hp*-FEM over low-order FEM using an IAEA benchmark problem and an axisymmetric problem involving the multigroup neutron diffusion equation. Section 3 describes the multimesh discretization technology and compares it to selected interpolation and projection data transfer methods. Conclusions are formulated in Section 6.

2. The adaptive higher-order finite element method (*hp*-FEM)

The *hp*-FEM is a modern version of the finite element method (FEM) that attains very fast (exponential) convergence rates by combining optimally finite elements of variable size (h) and polynomial degree (p) [16]. The main principles of exponential convergence are that (a) very smooth, polynomial-like functions are approximated using large high-order elements and (b) non-analytic (non-polynomial-like) functions such as singularities are approximated via small low-order ones. The superiority of the *hp*-FEM over standard (low-order) FEM has been demonstrated by many independent researchers [1, 2, 3, 10, 9, 16]. Although the implementation of the *hp*-FEM is involved, the method is becoming increasingly popular among practitioners.

The *hp*-FEM differs from standard FEM in a number of aspects, and in particular the complexity of spatial adaptation due to the large number of ways an element can be refined. One can increase the polynomial degree of an element without subdividing it in space or one can split the element in space and distribute the polynomial degrees in the subelements in multiple ways. The number of refinement candidates considered here is approximately 100 in 2D and several hundred in 3D. An example for cubic triangles and cubic quadrilaterals is shown in Fig. 2.

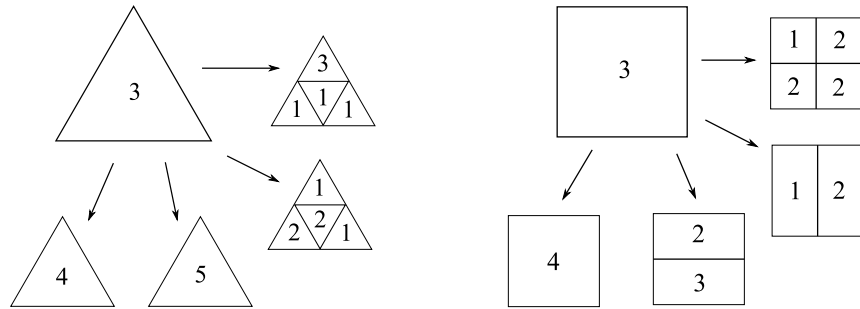


Figure 2: Examples of different *hp* refinement options for a triangular and quadrilateral element. Note that the ability to refine elements using subdivision and elevate the order of the element leads to many different possibilities. The number shown at the center of the element is the polynomial degree of the element.

In low-order FEM, the number of refinement candidates is small and thus it is sufficient to guide adaptation using element-wise constant error estimators. However, such error estimators do not provide enough

information to guide *hp*-adaptation. For this, one needs to have much better information about the *shape* of the error function $e_{h,p} = u - u_{h,p}$ in addition to its magnitude. In principle, it is possible to obtain this information from estimates of higher derivatives of the solution, but this approach is not usually practical. One way to circumvent this problem is to employ a *reference solution*; an approximation u_{ref} that is at least one order more accurate than the coarse solution $u_{h,p}$. The *hp*-adaptation algorithm is then guided by an *a-posteriori* error estimate of the form $e_{h,p} = u_{\text{ref}} - u_{h,p}$. More details on automatic *hp*-adaptation on meshes with arbitrary level hanging nodes may be found in [13].

2.1. Example: IAEA benchmark EIR-2

This example was chosen due to the excellent work on this problem by Wang and Ragusa ([17], p. 46). This paper poses a multiple mesh *hp*-FEM method and demonstrates it on several examples of multigroup diffusion in one dimension, using a modified version of the 1Dhp90 code [4]. They then present a single mesh, single group neutron diffusion solution of EIR-2, the *Saphir* benchmark.

This classical nuclear engineering benchmark [11] describing an external-force-driven configuration without fissile materials present will be used to compare the performance of adaptive *hp*-FEM and adaptive *h*-FEM with linear and quadratic quadrilateral elements¹. The domain is a 96×86 cm rectangle comprising five regions as illustrated in Fig. 3.

Solved is the one-group neutron diffusion equation

$$-\nabla \cdot (D(x, y) \nabla \phi) + \Sigma_a(x, y) \phi = Q_{\text{ext}}(x, y)$$

for the unknown neutron flux $\phi(x, y)$. Zero flux boundary conditions are assumed on the entire outer boundary. The values of the diffusion coefficient $D(x, y)$, absorption cross-section $\Sigma_a(x, y)$ and the source term $Q_{\text{ext}}(x, y)$ are constant in the subdomains. The total cross-section Σ_t is 0.60 in Ω_1 , 0.48 in Ω_2 , 0.70 in Ω_3 , 0.85 in Ω_4 , and 0.90 cm^{-1} in Ω_5 . The scattering cross-section Σ_s is 0.53 in Ω_1 , 0.20 in Ω_2 , 0.66 in Ω_3 , 0.50 in Ω_4 , and 0.89 cm^{-1} in Ω_5 . The absorption cross-section $\Sigma_a = \Sigma_t - \Sigma_s$. The diffusion coefficient $D(x, y)$ is calculated from Σ_t using the standard relation $D = 1/(3\Sigma_t)$. The source $Q_{\text{ext}} = 1 \text{ cm}^{-3} \text{ t}^{-1}$ in areas Ω_1 and Ω_3 and zero elsewhere.

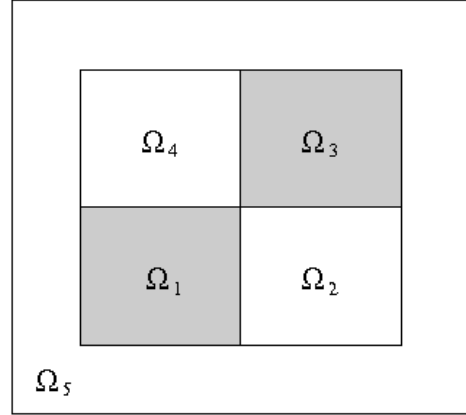


Figure 3: Geometry of IAEA benchmark EIR-2.

2.1.1. Results

The results presented below were obtained using Hermes2D², an open-source modular C++ library for solving various types of partial differential equations (PDE) as well as multiphysics PDE systems using adaptive higher-order finite element methods (*hp*-FEM) and Discontinuous Galerkin methods (*hp*-DG). It provides various capabilities for automatic adaptation of *hp*-meshes, including dynamical meshes for time-dependent problems, and supports the solution of coupled multiphysics problems in a monolithic way without operator splitting. The solution $\phi(x, y)$ is shown in Fig. 4.

Figures 5 and 6 show the resulting meshes (adaptive *h*-FEM with linear and quadratic elements and adaptive *hp*-FEM, respectively).

¹Note: it would be more correct to call these elements *bilinear* and *biquadratic*, respectively, but we will stay with the shorter notation.

²<http://hpfem.org/hermes2d/>

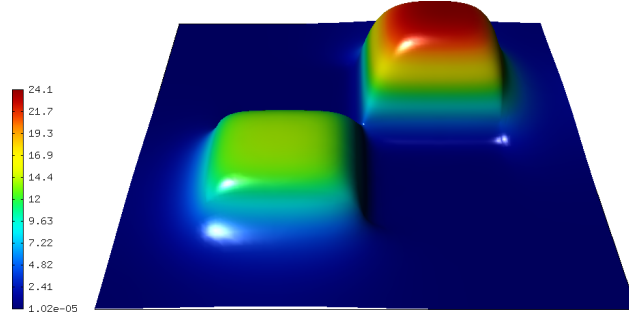


Figure 4: Solution $\Phi(x, y)$.

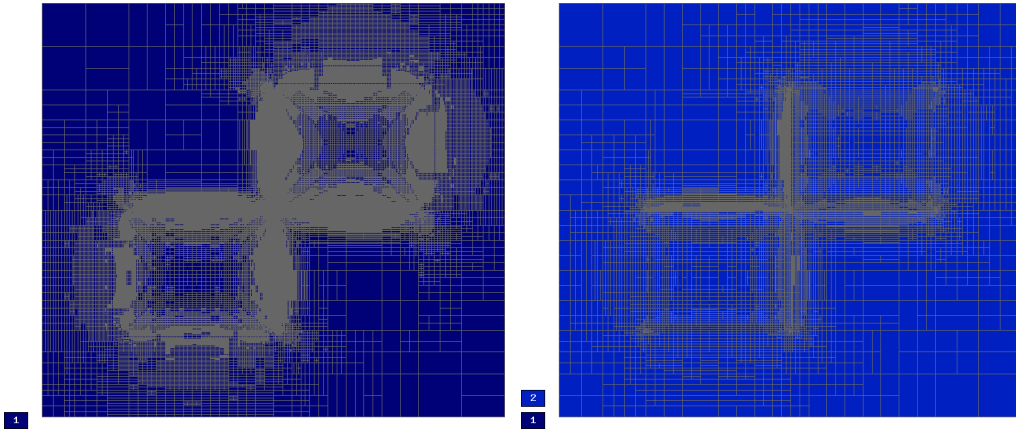


Figure 5: Resulting meshes for adaptive h -FEM with linear and quadratic elements.

In Fig. 6, colors are used to distinguish between different polynomial degrees of elements. In the hp -FEM, edges and element interiors can have different polynomial degrees. Polynomial degrees of edges are depicted using thin color belts along them. If the interior of an element is split diagonally and contains two colors, different directional (anisotropic) polynomial degrees are used.

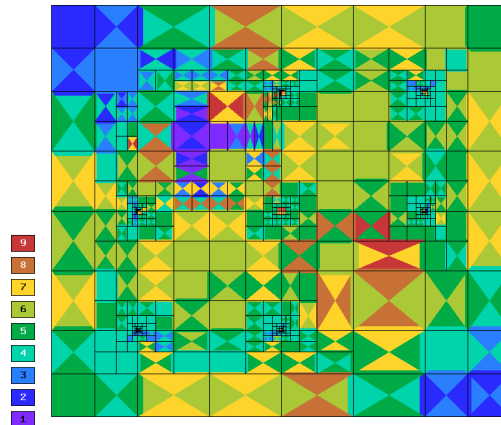


Figure 6: Resulting mesh for adaptive hp -FEM.

A comparison of the corresponding convergence rates in the H^1 -norm, both in terms of the discrete problem size (degrees of freedom, DOF) and CPU time, is shown in Figs. 7 and 8.

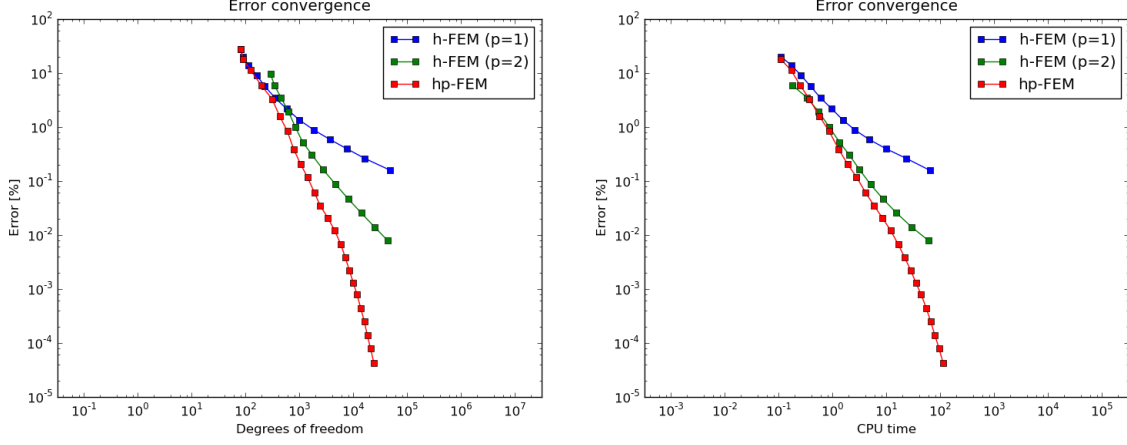


Figure 7: Convergence comparison in terms of discrete problem size (DOF) and CPU time.

In these results, adaptive hp -FEM is superior to both adaptive h -FEM with linear elements and adaptive h -FEM with quadratic elements. One may also see that the theoretically-predicted asymptotic slope $-1/2$ for h -FEM with linear elements as well as the slope -1 for h -FEM with quadratic elements is attained with approximately 5000 degrees of freedom.

This section closes by comparing the performance of the adaptive hp -FEM algorithm in Hermes2D with the performance of the hp -FEM algorithm presented by Wang and Ragusa for the same benchmark problem (see Fig. 25 in [17]). The code used by Wang and Ragusa is based on an adaptive hp -FEM code developed by Demkowicz [5]. Both codes use the same type of reference solution and measure the error in the same norm.

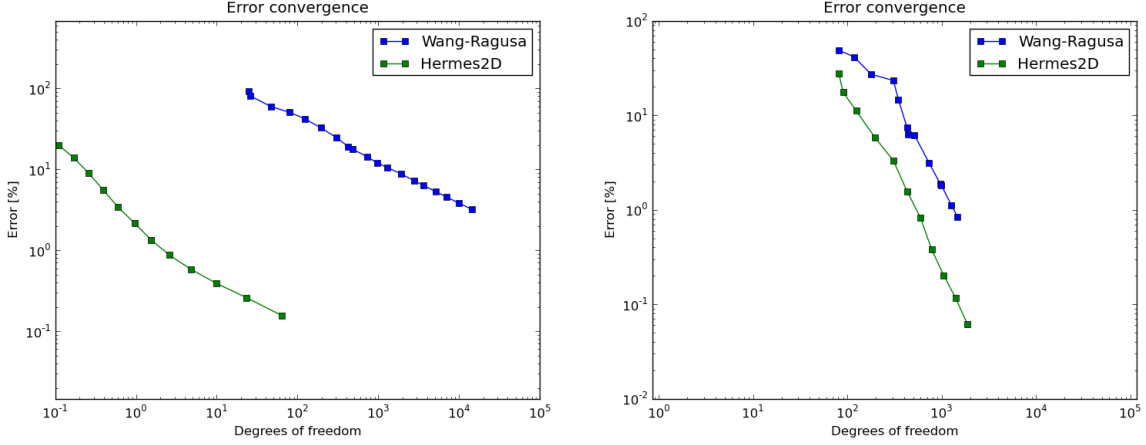


Figure 8: Convergence comparison of adaptive h -FEM with linear elements (left) and adaptive hp -FEM (right) performed using Hermes2D in comparison with the results of Wang and Ragusa [17].

Figure 8 shows that Hermes2D provides substantially better convergence for both h -FEM (with linear elements) and hp -FEM. To explore this result, consider the hp -FEM results. Using Wang's notation, the theoretical convergence rate of hp -FEM is proportional to $\exp(\gamma N^{1/3})$, where $\gamma < 0$. Note that the slope of

the data is very similar, but the constant of proportionality is smaller with Hermes2D, resulting in a constant advantage of requiring roughly half the DOFs for the same level of accuracy. The advantage is amplified when the results for h -FEM are considered. The better performance of Hermes2D may be related to the fact that it employs a different adaptivity algorithm, both spatially *and polynomially* anisotropic hp -refinements, and arbitrary-level hanging nodes.

3. Multimesh hp -FEM

Many of the physical phenomena occurring in nuclear reactors are strongly coupled. The length scales of the various physics components typically vary significantly in time and space, and the discretization requirements of the components are often quite different. To address these requirements, different meshes are typically used to host the solution of the different components, and these meshes may evolve independently over time. This study proposes a novel monolithic approach that preserves the coupling structure of multiphysics problems from the continuous to the discrete level. The method is based on an adaptive multimesh hp -FEM discretization technique [14, 6, 15], which preserves the consistent discretization of the problem even as components of the solution are discretized on different meshes.

In standard hp -FEM, the domain Ω is covered with a finite element mesh \mathcal{T} consisting of non-overlapping convex elements K_1, K_2, \dots, K_M (typically triangles or quadrilatera) hosting polynomial bases $1 \leq p_1, p_2, \dots, p_M$. In general, the N solution components exhibit different behavior requiring that they be approximated on different meshes $\mathcal{T}_n = \{K_1^n, K_2^n, \dots, K_M^n\}, n = 1, \dots, N$. The elements in these meshes possess nonuniform polynomial degrees $p(K_i^n) = p_i^n$. In order to limit the algorithm to a reasonable level of complexity, this approach establishes a geometric relationship between the meshes. This relationship begins with the definition of a coarse *master mesh* \mathcal{T}_m , that serves as the common basis of refinement for the coupled solution. Given this basis \mathcal{T}_m , a set of refinement operations are applied to develop an appropriate mesh \mathcal{T}_n for the n^{th} solution component of interest. The refinement procedure applied for each \mathcal{T}_n may be mutually independent for each component.

Use of the master mesh as a common basis of refinement significantly simplifies the approach, as it eliminates the need to perform geometric intersection operations to relate the solution between meshes. Further, the hierarchical relationship between the discrete spaces provides an effective mechanism for the representation and assembly of the finite element problem. In this case, the stiffness matrix is assembled on a *virtual union mesh* \mathcal{T}_u , which is the geometrical union of the meshes \mathcal{T}_n . To visualize this union of mesh representations, imagine printing the meshes on transparencies and overlaying them one above the other, as illustrated in Figure 3).

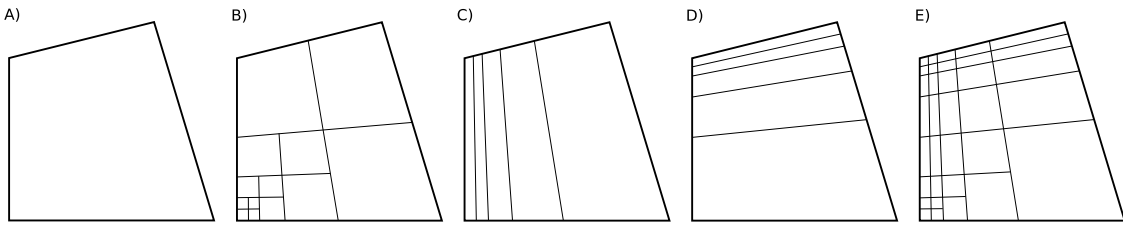


Figure 9: Diagram A) shows the master mesh \mathcal{T}_m that is the common basis of refinement of the multiple multiphysics meshes. B) through D) are meshes $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ obtained by refinement of the master mesh to meet the requirements of physics packages hosted upon them. The multiphysics problem is actually assembled on the union mesh \mathcal{T}_u , that is the intersection of B) through D).

The union mesh \mathcal{T}_u is not explicitly constructed; its virtual elements are traversed by the element-by-element assembling procedure employed in standard hp -FEM [16]. This approach does not incur interpolation or projection error, as information is not actually transferred between the meshes; each element on each mesh is used to integrate the solution that the element hosts. In other words, all integrals in the weak formulation of each coupled problem are evaluated exactly (up to the error in numerical quadrature).

4. Comparison to interpolation- and projection-based intermesh data transfer methods

In this section we use a problem with known exact solution to compare the multimesh discretization approach with two standard techniques to transfer solution data between different meshes: linear interpolation of vertex values and L_2 projection. We will solve a heat transfer equation with a spatially-dependent thermal conductivity $k = k(x, y)$,

$$\begin{aligned} -\nabla \cdot (k \nabla u) &= f & \text{in } \Omega = (0.01, 1.0)^2, \\ u &= u_D & \text{on } \partial\Omega. \end{aligned} \quad (1)$$

The weak problem is to find $u \in H_1(\Omega)$ satisfying $u = u_D$ on $\partial\Omega$ such that

$$\int_{\Omega} k \nabla u \cdot \nabla v \, dx dy = \int_{\Omega} f v \, dx dy, \quad \text{for all } v \in H_0^1(\Omega). \quad (2)$$

With

$$k(x, y) = x^3 + y^3, \quad f(x, y) = 2 - 2 \left(\frac{y^3}{x^3} + \frac{x^3}{y^3} \right) \quad \text{and} \quad u_D(x, y) = \frac{1}{x} + \frac{1}{y},$$

the exact solution is given by

$$u(x, y) = \frac{1}{x} + \frac{1}{y}.$$

For comparison purposes, two different meshes \mathcal{T}_A and \mathcal{T}_B were generated as follows: A piecewise-linear mesh \mathcal{T}_A is obtained by solving equation (2) using adaptive h -FEM. This mesh contains 189 DOF and it is virtually optimal to represent the exact solution u (see left part of Fig. 10). Another mesh \mathcal{T}_B is a uniform piecewise-linear 256×256 subdivision of the computational domain containing 65025 DOF, as shown in the right part of Fig. 10. The thermal conductivity $k(x, y)$ is represented on the mesh \mathcal{T}_B via its continuous, piecewise-linear vertex interpolant $k_h(x, y)$.

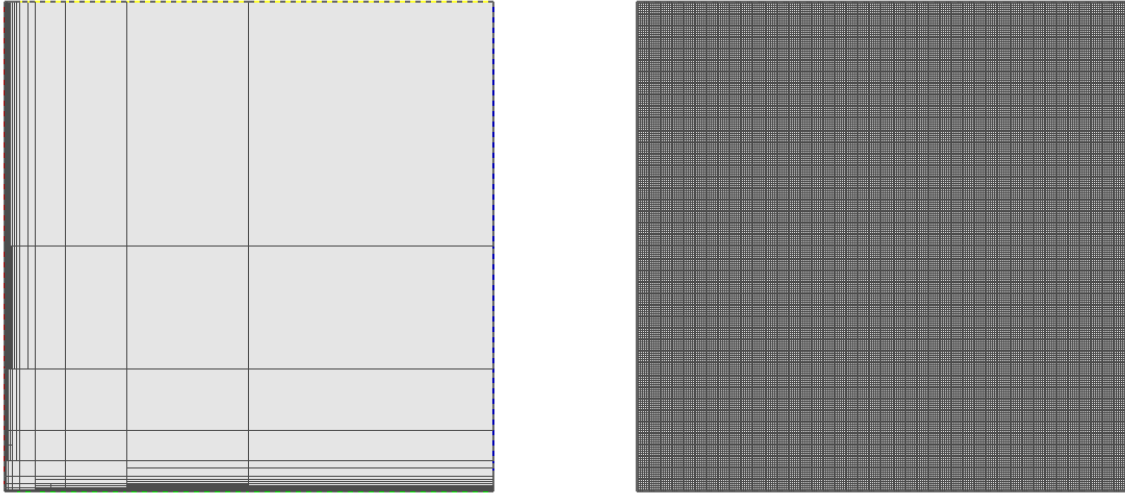


Figure 10: Left: non-uniform piecewise-linear mesh \mathcal{T}_A with 189 DOF where equation (2) is solved. Right: uniform piecewise-linear mesh \mathcal{T}_B with 65025 DOF where the thermal conductivity $k(x, y)$ is represented via its continuous, piecewise-linear vertex interpolant $k_h(x, y)$.

In the following we solve equation (2) in four different ways and compare the results to the exact solution u :

(1) *Transferring $k_h(x, y)$ from \mathcal{T}_B to \mathcal{T}_A via interpolation and solving on \mathcal{T}_A*

Let us begin with linear interpolation of vertex values, which probably is the most widely used method for transferring solution data between different meshes. The piecewise-linear function $k_h(x, y)$ defined on the mesh \mathcal{T}_B is approximated on the mesh \mathcal{T}_A by means of a continuous function $k_h^{(1)}(x, y)$ that matches $k_h(x, y)$ exactly at all grid vertices of the mesh \mathcal{T}_A and is bilinear in each element of \mathcal{T}_A .

In other words, $k_h^{(1)} = \sum_i c_i v_i(x, y)$ where $v_i \in V_h^A$ are the vertex basis functions on the mesh \mathcal{T}_A , and the coefficients c_i are the values of the function $k(x, y)$ at the corresponding grid points of \mathcal{T}_A . (For higher-order hierarchic finite elements, the natural extension of linear interpolation is the so-called *projection-based interpolation* [16].)

The corresponding approximation $u_h^{(1)}$, computed on the mesh \mathcal{T}_A using linear finite elements and the approximate thermal conductivity $k_h^{(1)}$, is shown in part (a) of Fig. 11. The approximation error $e_h^{(1)}(x, y) = u(x, y) - u_h^{(1)}(x, y)$ is shown in part (a) of Fig. 12.

(2) *Transferring $k_h(x, y)$ from \mathcal{T}_B to \mathcal{T}_A via L^2 -projection and solving on \mathcal{T}_A*

Let us denote by V_h^A the finite element subspace of $H^1(\Omega)$ on the piecewise-linear mesh \mathcal{T}_A . The projection problem $\|k_h^{(2)} - k_h\|_{L^2} \rightarrow 0$ can be rewritten in weak sense as $(k_h^{(2)} - k_h, v_i)_{L^2} = 0$ or

$$(k_h^{(2)}, v_i)_{L^2} = (k_h, v_i)_{L^2} \quad (3)$$

for all basis functions v_i of the space V_h^A . The unknown projection $k_h^{(2)}$ is written as a linear combination

$$k_h^{(2)} = \sum_{j=1}^N y_j v_j$$

where y_j are unknown coefficients. Substituting this sum into the variational identity (3), one obtains a system of linear algebraic equations. The mass matrix integrals of the form $\int_{\Omega} v_j v_i \, dx dy$ are computed over elements of the mesh \mathcal{T}_A . Following the common refinement algorithm proposed by [8], the right-hand side integrals $\int_{\Omega} k_h v_i \, dx dy$ are computed using elements of the geometrical union of the meshes \mathcal{T}_A and \mathcal{T}_B (that is \mathcal{T}_B in our case). The corresponding piecewise-linear approximation $u_h^{(2)}$ on the mesh \mathcal{T}_A is shown in part (b) of Fig. 11. The approximation error $e_h^{(2)}(x, y) = u(x, y) - u_h^{(2)}(x, y)$ is shown in part (b) of Fig. 12.

(3) *Using the multimesh discretization method (no data transfer between meshes)*

Next we compute a piecewise-linear approximation $u_h^{(3)}$ using the multimesh discretization method that was described in Section 3. Note that in this case, the union mesh of the meshes \mathcal{T}_A and \mathcal{T}_B is \mathcal{T}_B . The approximation is shown in part (c) of Fig. 11, and the approximation error $e_h^{(3)}(x, y) = u(x, y) - u_h^{(3)}(x, y)$ is shown in part (c) of Fig. 12.

(4) *Solving on \mathcal{T}_A with the exact thermal conductivity $k(x, y)$*

For comparison purposes we also compute a piecewise-linear approximation $u_h^{(4)}$ on the mesh \mathcal{T}_A using the original function $k(x, y)$. This is the best finite element approximation one can obtain on the mesh \mathcal{T}_A . The result is shown in part (d) of Fig. 11 and the corresponding approximation error $e_h^{(4)}(x, y) = u(x, y) - u_h^{(4)}(x, y)$ is shown in part (d) of Fig. 12.

The results of the four computations are summarized in Table 1 which shows H^1 -norms of the approximation errors on the mesh \mathcal{T}_A calculated with respect to the exact solution u . Approaches (1) and (2) resulted in a large error in the solution. In particular, notice the extreme error in case (2). As discussed in [8], one must typically use a higher order integration on the target mesh. Thus, a second calculation was performed using quadratic elements on \mathcal{T}_A to yield the third row of Table 1. Note that the multimesh discretization provides slightly better results than the approximation calculated using the exact data $k(x, y)$ on the mesh \mathcal{T}_A . This is due to the fact that the multimesh method integrates over elements of the union mesh (in this case the mesh \mathcal{T}_B) which is much finer than the mesh \mathcal{T}_A used for assembling in case (4).

Table 1: Comparison of relative H^1 -norm errors for the four methods described above.

Method	Relative H^1 error
Interpolation of $k_h^{(1)}$ on mesh \mathcal{T}_A	17.766%
L^2 -projection of $k_h^{(1)}$ on mesh \mathcal{T}_A (lin.)	31.808%
L^2 -projection of $k_h^{(1)}$ on mesh \mathcal{T}_A (quadr.)	6.734%
Multimesh discretization method	6.727%
Using exact data $k(x, y)$ on mesh \mathcal{T}_A	6.729%

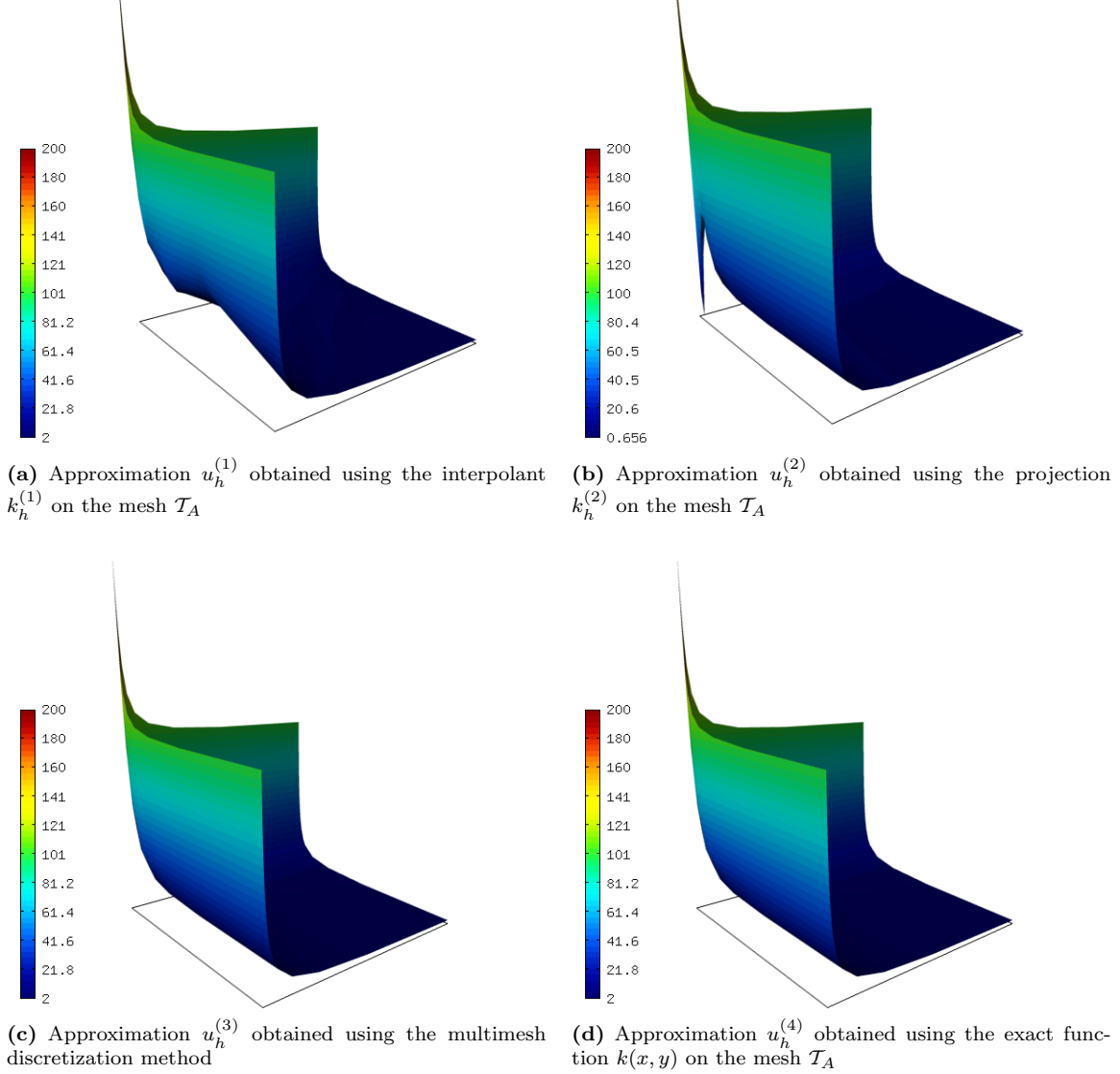


Figure 11: Finite element solutions to (2). Note that the two methods that transfer the thermal conductivity between the meshes \mathcal{T}_B and \mathcal{T}_A (a and b) possess significant error close to the origin, while the multimesh discretization yields results that are quite close to the exact solution.

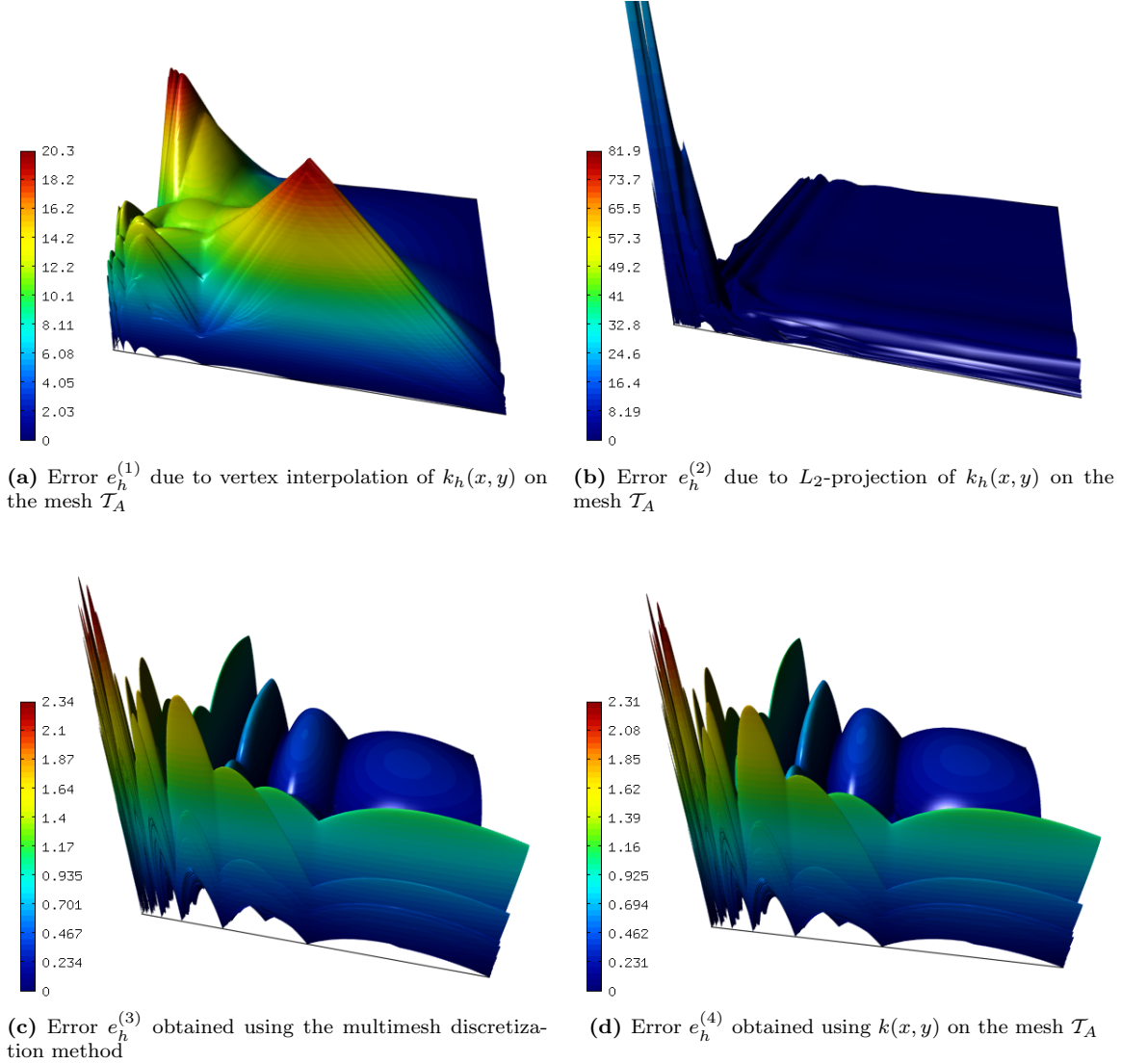


Figure 12: Approximation errors on the mesh \mathcal{T}_A calculated with respect to the exact solution u . Note that the scale of the error magnitude is significantly different between the plots.

5. Automatic adaptivity in multimesh hp -FEM

Automatic adaptivity in multimesh hp -FEM [6] is much more involved than standard adaptive hp -FEM [5, 16] where only one mesh is used. In the multimesh case, one needs to define a global error norm that includes error contributions from all meshes considered as a group, and refine the meshes for each physical model to minimize that global error norm. As discussed in [17], Fig. 4, one may take each physical model in turn and adapt it (using the standard hp -FEM approach) until the error for that model is below some tolerance. At this point, one moves to the next model and repeats the process. Unfortunately, refinement on each mesh may effect the refinement needed on the others at particular spatial locations. Figure 5 in [17] presents a second algorithm that selects elements to be refined based on the maximum error across all models and across all elements. While this approach requires that the error magnitudes be normalized across

all physical models, it is employed for these results. Thus, in the multimesh hp -FEM presented here, every element competes directly with all elements on all meshes in the system.

5.1. Example: Multigroup neutron diffusion

This section compares the performance of adaptive hp -FEM and h -FEM by solving a multigroup neutron diffusion problem on a two dimensional multimesh. The reactor core shown in Fig. 13 is modeled in two dimensions using cylindrical coordinates. The computational domain is divided into 5 zones; center, outer, upper, and lower reflector regions, and a fissile core. Detailed geometry within each zone is homogenized into the zone for simplification.

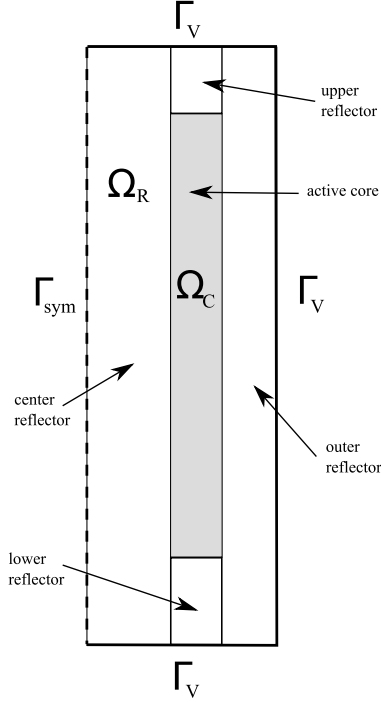


Figure 13: Diagram of the simplified prismatic VHTR core model.

The reactor neutronics is given by the following eigenproblem,

$$-\nabla \cdot D_g \nabla \phi_g + \Sigma_{Rg} \phi_g - \sum_{g' \neq g} \Sigma_s^{g' \rightarrow g} \phi_{g'} = \frac{\chi_g}{k_{\text{eff}}} \sum_{g'} \nu_{g'} \Sigma_{fg'} \phi_{g'}, \quad g = 1, \dots, 4, \quad (4)$$

with boundary conditions

$$\frac{\partial \phi_g}{\partial n} = 0 \quad \text{on } \Gamma_{\text{SYM}}, \quad \text{and} \quad D_g \frac{\partial \phi_g}{\partial n} = -\frac{1}{2} \phi_g \quad \text{on } \Gamma_V. \quad (5)$$

Here, ϕ_g is the neutron flux, D_g is the diffusion coefficient, $\Sigma_{Rg} = \Sigma_{ag} + \Sigma_s^{g \rightarrow g+1}$ is the removal cross section, Σ_{fg} is the fission cross section, Σ_s is the scattering cross section, ν is the average number of neutrons born per fission event, and the subscript g denotes the group index in the multigroup neutron diffusion equation. Lastly, k_{eff} is the eigenvalue.

This eigenproblem is numerically solved using the power method (or power iteration), as outlined by the following algorithm:

1. Make an initial estimate of $\phi_g^{(0)}$ and $k^{(0)}$

2. For $n = 1, 2, \dots$

(a) solve for $\phi_g^{(n)}$ using

$$-\nabla \cdot D_g \nabla \phi_g^{(n)} + \Sigma_{Rg} \phi_g^{(n)} - \sum_{g' \neq g} \Sigma_s^{g' \rightarrow g} \phi_{g'}^{(n)} = \frac{\chi_g}{k^{(n-1)}} \sum_{g'} \nu_{g'} \Sigma_{fg'} \phi_{g'}^{(n-1)} \quad (6)$$

(b) solve for $k^{(n)}$ using

$$k^{(n)} = k^{(n-1)} \frac{\int_{\Omega_C} \sum_{g=1}^4 \nu_g \Sigma_{fg} \phi_g^{(n)}}{\int_{\Omega_C} \sum_{g=1}^4 \nu_g \Sigma_{fg} \phi_g^{(n-1)}} \quad (7)$$

3. Finally, stop iterating when

$$\left| \frac{k^{(n)} - k^{(n-1)}}{k^{(n)}} \right| < \epsilon \quad (8)$$

5.1.1. Results

The calculations that follow are using the material properties shown in Tab. 2:

Table 2: Material properties for the geometry shown in Fig. 13.

Active core properties					Reflector properties				
	1	2	3	4		1	2	3	4
D_g (m)	0.0235	0.00121	0.0119	0.0116	D_g (m)	0.0164	0.0085	0.00832	0.00821
Σ_{Rg} (m ⁻¹)	0.00977	0.162	0.156	0.535	Σ_{Rg} (m ⁻¹)	0.00139	0.000218	0.00197	0.0106
Σ_{fg} (m ⁻¹)	0.00395	0.0262	0.0718	0.346	Σ_{fg} (m ⁻¹)	0	0	0	0
ν_g	2.49	2.43	2.42	2.42	ν_g	0	0	0	0
χ_g	0.9675	0.03250	0.0	0.0	χ_g	0	0	0	0
$\Sigma_s^{1 \rightarrow 2}$ (m ⁻¹)	1.23	0	0	0	$\Sigma_s^{1 \rightarrow 2}$ (m ⁻¹)	1.77	0	0	0
$\Sigma_s^{2 \rightarrow 3}$ (m ⁻¹)	0	0.367	0	0	$\Sigma_s^{2 \rightarrow 3}$ (m ⁻¹)	0	0.533	0	0
$\Sigma_s^{3 \rightarrow 4}$ (m ⁻¹)	0	0	2.28	0	$\Sigma_s^{3 \rightarrow 4}$ (m ⁻¹)	0	0	3.31	0

The problem was solved using both hp -adaptation and standard h -adaptation on low-order elements (with bilinear and biquadratic quadrilateral elements). The relative error between the coarse mesh solution ϕ_g and the reference solution $\tilde{\phi}_g$ was calculated using

$$E_{\text{rel}}^2 = \frac{\sum_g \|\phi_g - \tilde{\phi}_g\|_{H^1}^2}{\sum_g \|\tilde{\phi}_g\|_{H^1}^2}.$$

Figure 14 shows the neutron flux for each of the four energy groups (eigenvectors), where flux intensity is shown in color (blue is the lowest flux and red is the highest). The eigenvalue in this problem was $k_{\text{eff}} = 1.14077$.

Using hp adaptation, a relative error of 0.0164% was achieved using 27903 degrees of freedom (DOF). The h refined results provided a relative error of 3.87% using 30112 DOF.

Figures 15 and 16 show four optimal hp -meshes and corresponding h -meshes, respectively. Figure 15 shows hp adapted solutions to each group of the four group problem. It is clear from this figure that the coupled problem is being solved with a mesh of a different refinement and elements of different order in each group. Note that the solutions for groups 1–3 shown in Fig. 14 are similar, as are their corresponding hp meshes. However, with detailed examination, both the solutions and meshes differ between groups. Note that both the solution and mesh for group 4 differs significantly from the others.

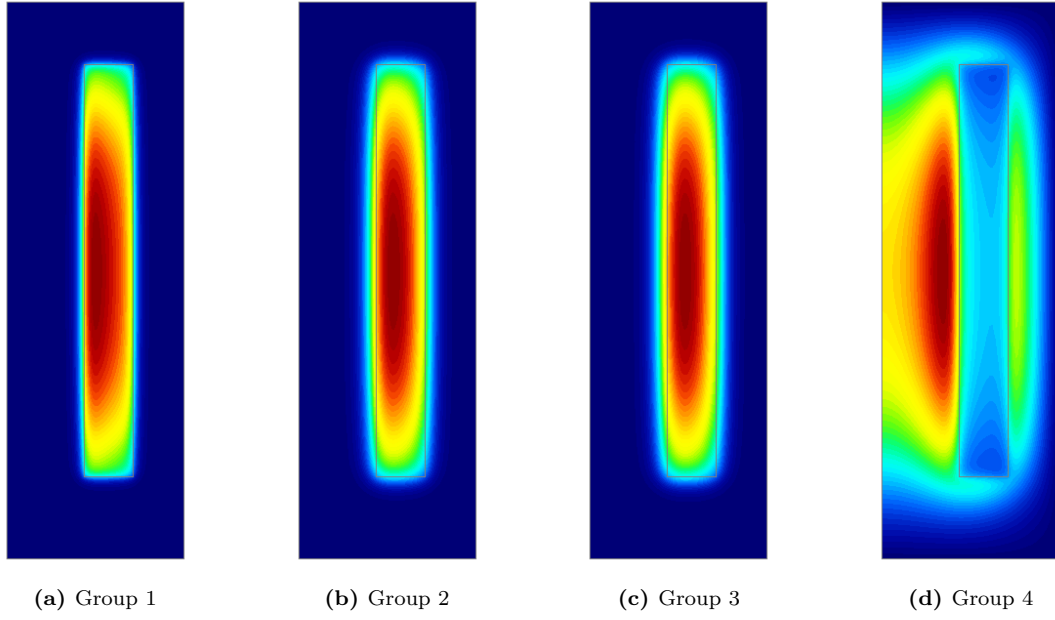


Figure 14: Neutron flux for the four group neutron diffusion calculation on the VHTR geometry. Color indicates the intensity of the flux in each group.

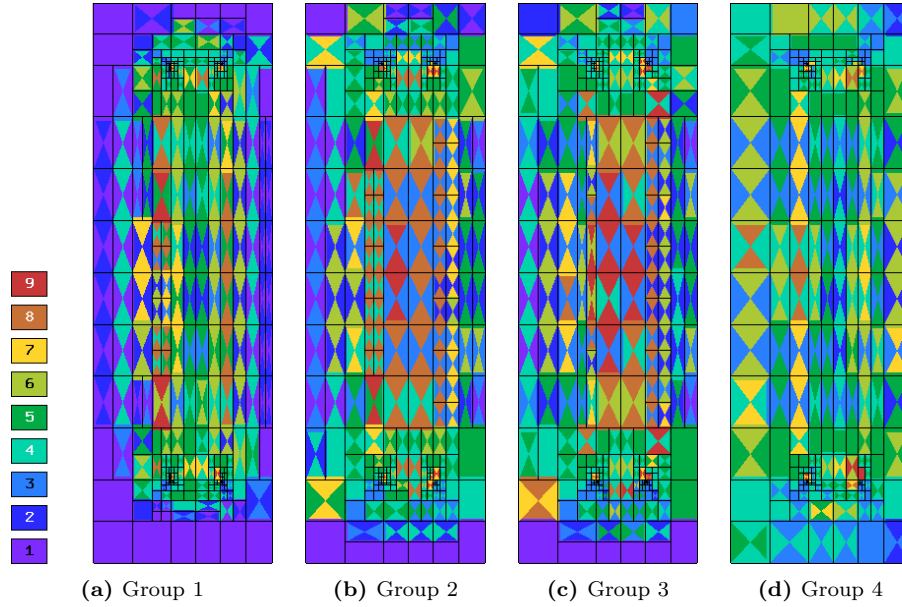


Figure 15: hp -meshes for that correspond to the above neutron flux results. Colors have the same meaning as in Fig. 6. Note that different levels of subdivision and different polynomial orders are used for each energy group. This solution contains 27903 DOF, and has a relative error of 0.0164%.

Figure 17 compares convergence curves on this problem using hp -FEM and standard FEM using both bilinear and biquadratic elements. It is apparent that in regions where the solutions are smooth, the algorithm tends to use larger elements with higher polynomial degree. In areas near singularities or geometric corners,

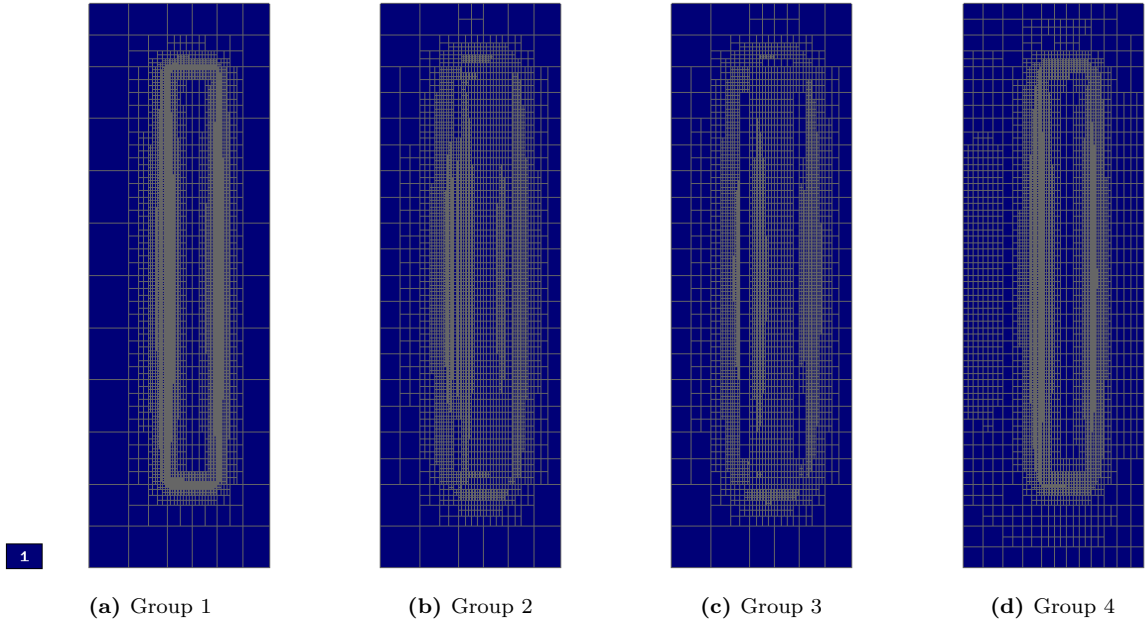


Figure 16: h -meshes (linear elements, $p = 1$) for the four group neutron flux problem. Again, each group uses a different refinement. Solution contains 30112 DOF, and has a relative error of 3.87%.

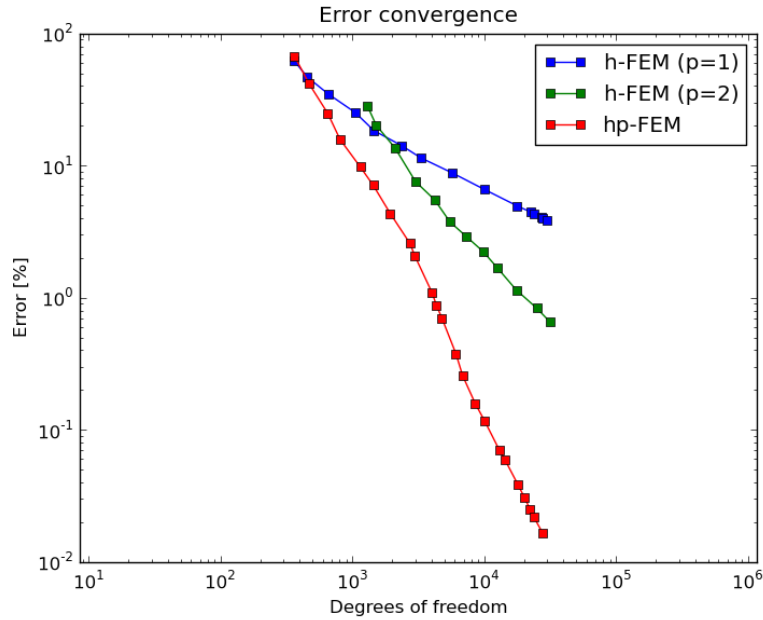


Figure 17: Comparison of the spatial convergence of hp - vs. h -adaptation on the multigroup multimesh neutron diffusion problem shown in Fig. 13. Three curves are shown, h results with both linear ($p = 1$) and quadratic ($p = 2$) polynomial order, and hp . Note the significant accuracy advantage of the hp method for a given number of DOFs.

smaller elements with lower polynomial degree yield the best results. This is in good correspondence with the theory of hp -FEM. On the other hand, h -adaptation can refine only spatially, which results in more degrees of freedom and a larger relative error as illustrated in Fig. 17.

Figure 18 shows two solutions of the multigroup problem, the first using a single mesh for each of the physics models, and the second using the multimesh approach to solve each group on its own mesh (Figs. 14 and 15). It is apparent that more degrees of freedom are needed to solve the four group problem using a single mesh, to obtain comparable accuracy. On this problem, the savings in total DOF obtained by using a multiple mesh strategy is not striking. Future work includes more study of the error metric used to adapt each mesh in turn, as a more discriminating metric may yield a greater performance difference when compared to a single mesh approach. Secondly, while there is some diversity in the spatial resolution requirements of the physics solutions hosted on each mesh, it is fairly small across the groups on this multigroup problem. A much greater diversity is expected between multigroup neutronics and models for fluid (coolant) flow, especially in areas of the turbulent boundary layer of the fluids calculation.

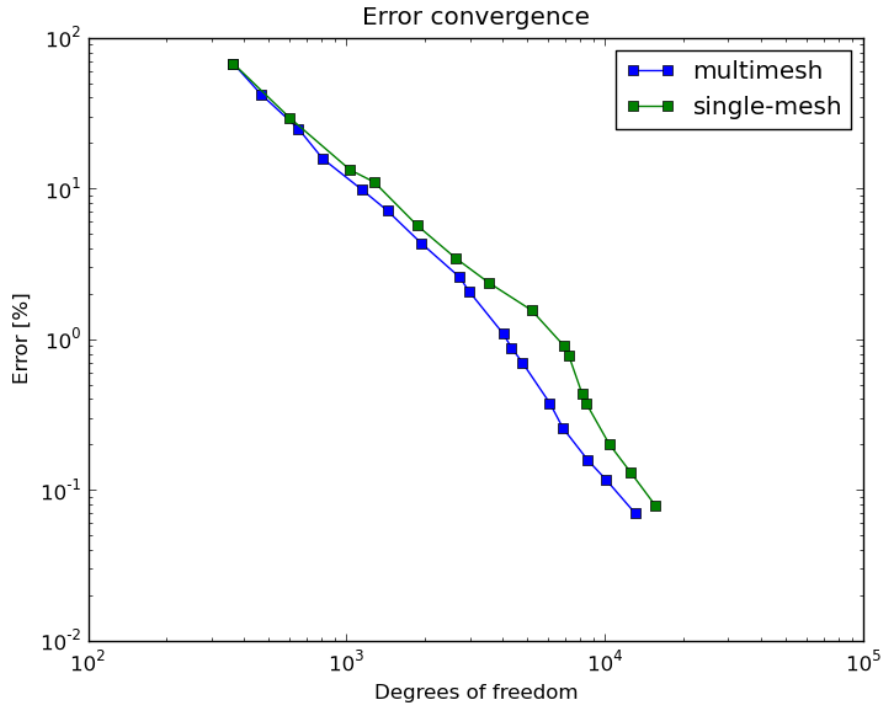


Figure 18: Comparison of hp -FEM spatial convergence for solving each group on a separate mesh using the multimesh approach, vs. using the same mesh for all the groups.

6. Conclusion

This paper began by introducing some of the multiphysics challenges within the field of computational nuclear engineering and motivating the need for integration methods that handle a wide diversity of spatial resolution needs across multiphysics models. The hp -FEM method was introduced, and solutions were presented that compared the method vs. h -FEM on the IAEA benchmark EIR-2.

The study then considered the use of hp -FEM as a methodology to spatially integrate and couple multiphysics problems using multiple meshes. Here, a hierarchy of meshes were used to define the method to

simplify the implementation and to eliminate the challenges of performing geometric intersection operations between topologically-distinct meshes. The proposed approach was compared to commonly used techniques for data transfers between meshes, on a problem with a known solution. It was shown that by using the proposed multimesh assembly process no additional error arises due to the use of distinct meshes in the problem considered.

The final portion of this study explored the adaptive, higher order finite element method (*hp*-FEM) and potential advantages of the approach over typical low-order FEM for the four group neutron diffusion equations. In this case, a multiple mesh problem was developed that hosted the solution for each group on a independent (but hierarchically related) mesh. An example problem was considered that exhibited an error of 3.87% using 30112 DOF, employing *h* adaptation and linear elements, but using a different mesh for each group. The second example used the proposed *hp* method, and produced an error of 0.0164% using 27903 DOF, again with a separate mesh per group. These results may be interpreted two ways; for a given level of accuracy the *hp* method will require considerable fewer DOF, or for a similar number of DOF, the *hp* method will provide over 200 times the accuracy (on this problem). Finally, a multimesh vs. single mesh calculation was compared. Here, the reduction in the number of DOFs needed for the same level was limited, likely in part due to the “closeness” of the multigroup solutions on each mesh. Future work includes study of this approach on problems with greater diversity such as Navier-Stokes coupled to neutronics, three dimensional problems, and more sophisticated error indicators and refinement strategies to refine each individual mesh separately.

Acknowledgments

The submitted manuscript has been authored by a contractor of the U.S. Government under Contract No. DEAC07-05ID14517 (INL/JOU-10-18340). Accordingly, the U.S. Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. The authors also acknowledge the financial support of the DOE NEUP program under BEA Contract No. 00089911. This work was also supported financially by the Grant Agency of the Academy of Sciences of the Czech Republic under Grant No. IAA100760702.

The authors wish to thank Yaqi Wang for sharing the convergence data from his paper [17].

References

- [1] I. Babuska and B. Q. Guo. The *h*, *p*, and *h-p* version of the finite element method; basis theory and applications. *Advances in Engineering Software*, 15:159–174, 1992.
- [2] I. Babuška and M. Suri. The *p* and *h – p* versions of the finite element method, basic principles and properties. *SIAM Rev.*, 36:578–632, 1994.
- [3] L. Demkowicz. *Computing with hp-Adaptive Finite Elements, Volume 1: One and Two Dimensional Elliptic and Maxwell Problems*. Chapman & Hall/CRC, Boca Raton, FL, 2006.
- [4] L. Demkowicz and C.-W. Kim. 1D *hp*-adaptive finite element package, Fortran-90 implementation (1Dhp90). Technical Report TICAM Report 99-38, Texas Institute of Computational and Applied Mathematics, University of Texas, Austin, 1999.
- [5] L. Demkowicz, W. Rachowicz, and Ph. Devloo. A fully automatic *hp*-adaptivity. *J. Sci. Comput.*, 17(1–3):127, 2002.
- [6] L. Dubcova, P. Solin, J. Cervený, and P. Kus. Space and time adaptive two-mesh *hp*-FEM for transient microwave heating problems. *Electromagnetics*, 2010. In press.
- [7] Glen Hansen and Steve Owen. Mesh generation technology for nuclear reactor simulation; barriers and opportunities. *Nucl. Engrg. Design*, 238(10):2590–2605, 2008.

- [8] X. Jiao and M. T. Heath. Common-refinement-based data transfer between non-matching meshes in multiphysics simulations. *Internat. J. Numer. Methods Engrg.*, 61(14):2402–2427, 2004.
- [9] George Karniadakis and Spencer Sherwin. *p- and hp-Finite Element Methods*. The Clarendon Press Oxford University Press, New York, 1998.
- [10] George Karniadakis and Spencer Sherwin. *Spectral/hp Element Methods for CFD*. Oxford University Press, Oxford, U.K., 2005.
- [11] A. Kavenoky, J. Stepanek, and F. Schmidt. Benchmark problems. Transport Theory and Advanced Reactor Simulations IAEA-TECDOC-254, International Atomic Energy Agency, 1979.
- [12] S. J. Owen and G. Hansen. Mesh generation for reactor modelling and simulation: Practices, procedures, and uncertainties. In *20th International Conference on Structural Mechanics in Reactor Technology (SMiRT 20)*, paper 2028, Espoo (Helsinki), Finland, August 9–14 2009.
- [13] P. Solin, J. Cerveny, and I. Dolezel. Arbitrary-level hanging nodes and automatic adaptivity in the *hp*-FEM. *Math. Comput. Simulation*, 77:117–132, 2008.
- [14] P. Solin, J. Cerveny, L. Dubcova, and D. Andrs. Monolithic discretization of linear thermoelasticity problems via adaptive multimesh *hp*-FEM. *J. Comput. Appl. Math.*, 2010. In press.
- [15] P. Solin, L. Dubcova, and J. Kruis. Adaptive *hp*-FEM with dynamical meshes for transient heat and moisture transfer problems. *J. Comput. Appl. Math.*, 2010. In press.
- [16] P. Solin, K. Segeth, and I. Dolezel. *Higher-Order Finite Element Methods*. Chapman & Hall/CRC Press, Philadelphia, PA, 2003.
- [17] Yaqi Wang and Jean Ragusa. Application of *hp* adaptivity to the multigroup diffusion equations. *Nuclear Science and Engineering*, 161:22–48, 2009.