A combined immersed body and adaptive mesh method for simulating neutron transport within complex structures

A.G. Buchan[a,b], S. Dargaville[b], C.C. Pain[b]

[a]*School of Engineering and Material Science*
Queen Mary University of London, E1 4NS

[b]*Applied Modelling and Computation Group*
Department of Earth Science and Engineering
Imperial College London, SW7 2AZ, UK

Abstract

This article describes a new adaptive immersed body method for the efficient and accurate modelling of neutron transport within geometrically complex domains. It combines two techniques of immersed body projections and self-adaptive mesh refinement to form a unique method that can efficiently resolve problems that contain complex internal structures. The approach allows complete freedom of where mesh resolution is placed and the meshes, which do not need to conform to the problem structure, are optimised for resolving the physics of the problem.

Importantly the benefits can be found for problems with many internal structures such as fuel pins, control rods or cooling pipes and channels. Standard finite element meshes typically become large as they capture the geometrical detail and as a result solving times typically increase. This method overcomes this issue and allows for smaller meshes that are optimised to increase solution accuracy, as demonstrated in the numerical examples.

Keywords: Neutron Transport; Boltzmann Transport Equation; Adaptive Finite Elements; Immersed Body Method

Email address: `andrew.buchan@imperial.ac.uk` (A.G. Buchan)

1. Introduction

Simulating the transport of neutrons within reactor cores and shields can be computationally expensive due to both the complexity of the equations describing their movement, and the intricate geometric detail of the problems involved. The equation itself, namely the Boltzmann transport equation (BTE) [1, 2] is set within a 7 dimensional phase-space and this high dimensionality alone can lead to very high numbers of degrees of freedom to solve following their discretisation. Moreover this is exasperated by the fact that reactors, radiation shields and nuclear processing facilities have complex designs involving networks of internal structures. These range from fuel pins, control elements, cooling channels and pipes, and a variety of structural and support components. As a result, explicit meshes of such geometries can also become exceptionally large, and the problem can become infeasible to solve even with the use of large HPC facilities. Alternatively coarse meshes can be used that do not resolve the geometry fully, and whilst this improves efficiency of solutions, the approximation accuracy can easily suffer as a result.

A wide and varying array of methods have been proposed over the years to provide efficient solutions to reactor physics as well as general radiation transport problems. A common approach is to recast the first order form of the Boltzmann transport equation into a more computationally efficient setting, typically with a reduced dimensionality, examples of which include the diffusion [3], integral [1] or second order [4, 5, 6, 7] equations. However, this recasting of the equations either assumes a simplistic nature of the transport process (e.g. highly diffusive fluxes when applying diffusion theory) or suffer short-comings when simulating neutron transport across particular material regimes (e.g. voids when using second order approaches). Regardless of the appropriateness of the equation's use with a problem type, all still require a spatial discretisation scheme with resolution sufficient to resolve the profile of the flux over the domain. Complicated geometries such as whole reactor cores therefore pose an issue since large spatial meshes or grids may become necessary to capture the detail of both the geometry and solution; this may substantially increase computer storage requirements and run times.

The finite element method (FEM) is widely employed in resolving radiation transport problems [1, 5], the first discontinuous formulation was in fact developed for this field [8]. The method has many advantages including the ability to easily discretise arbitrary geometries, and to vary resolution across the solution domain. These properties make the method suitable for nuclear engineering applications involving complex structures with heterogeneous materials. The ability to vary resolution has led to much work being devoted to adaptive mesh methods that place resolution appropriately. These approaches vary from Adaptive Mesh Refinement techniques (AMR) [9, 10], that adapt meshes through a hierarchical partitioning of elements into smaller sub-elements, to fully adaptive methods using unstructured grids [11, 12, 13, 14]. One limiting factor of the FEM is its common use of straight

2

edged elements meaning curved surfaces and interfaces between regions are difficult to capture. In reactor physics this is particularly problematic as non-conserved volumes, due to meshes that do not conform to the geometry, can significantly affect accuracy [15], particularly for eigenvalue calculations. One approach around this could be to use curvilinear elements [16] that map linear elements to the curved surfaces, this has been particularly useful in forming appropriate meshes for boundary-layers in flow problems. Alternative, and successful, approaches are that of isogeometric methods [17]. These make use of the NURBS (Non-Uniform Rational B-Spline) representation of surfaces within CAD packages to formulate meshes that do conform the geometry. In reactor physics these methods have been developed for both the diffusion [18] and the first order S_N [19] equations.

In this article an alternative approach is proposed for the efficient modelling of reactor systems through the combination of adaptive finite elements [13] and immersed body methods (IBM) [20] for the spatial discretisation of the BTE. The immersed body method uses the supermeshing algorithm [21, 22] to project the meshes of internal structures onto arbitrary global FEM meshes, upon which the neutron radiation field is solved. Importantly these global meshes are arbitrary and are not forced to conform to a problem's detail. Instead the supermesh algorithm conservatively maps the volumes of internal structures onto the mesh, and so each mesh element will know what materials it contains. With this information appropriate material cross-sections are formed to enable the radiation flux to be resolved. Combining the use of non-conforming meshes together with the adaptive mesh refinement scheme, resolution can be placed exactly where it is needed. It allows appropriate meshes to form that resolves the physics of the problem, and not be forced to place unnecessary resolution to capture geometric detail. This ultimately this leads to reduced mesh sizes that increases efficiency without affecting solution quality. The benefits of the approach is demonstrated in this article by resolving both eigenvalue and fixed source problems. The intention here is to show how the accuracy and efficiency of the solutions can be improved in comparison to other standard FEM methods, thus a more comprehensive verification and validation exercise will be left to future works.

The use of immersed body and/or adaptive methods have been previously applied to other fields of physics successfully. The first immersed body method was developed by Peskin [23, 24] which was formulated to simulate flows around the complex structure of the heart. Since then a number of methods based on the similar principles have been proposed, a useful review has been published by Mittal and Iaccarino [25]. More recent developments include the work of [26] where their application extends to unstructured finite element discretisations, to modelling wind turbines [27], and break water simulation [28]. These employ the supermeshing method for mapping information between fluid-structure meshes. The supermeshing method has also been used in the general field of computing diagnostics of functions across multiple meshes [29]. In reactor physics IBM methods have been applied to resolve neutron radiation field [20] and used in coupled neutron-thermal hydraulic calculations [30, 31]. It may have the potential to be used for full sub-channel analysis, provided appropriate

correlations (e.g. heat transfer and drag) are applied, and this may be investigated in future articles. However it could certainly benefit other nuclear applications such as those involving fissile solutions with complex internal structures, for example the SUPO reactor which is described in the numerical examples. The anisotropic adaptive method that forms the basis of this article has also been applied across many fields, ranging from ocean models [32, 33] to radiation transport problems [34, 35]. However this is the first time that the combined IBM and adaptive resolution method have been used in reactor physics problems.

The structure of this article is as follows. Section 2 introduces both the fixed source and eigenvalue form of the Boltzmann transport equation. Section 3 presents the immersed body projection algorithm and section 3 gives details of the adaptive mesh refinement procedures and how it is combined with the IBM for use in solving the BTE. Section 5 presents three numerical examples to demonstrate the capability of the method and conclusions are drawn in section 6.

2. The Boltzmann Transport Equation

This article will consider both the time independent and eigenvalue form of the Boltzmann transport equation. These are written as,

$$\mathbf{\Omega} \cdot \nabla \psi\left(\mathbf{r}, \mathbf{\Omega}, E\right) + \Sigma_t\left(\mathbf{r}, E\right) \psi\left(\mathbf{r}, \mathbf{\Omega}, E\right) =$$
$$\int_{4\pi} dE' \int_{4\pi} d\mathbf{\Omega}' \Sigma_s\left(\mathbf{r}, \mathbf{\Omega}' \to \mathbf{\Omega}, E' \to E\right) \psi\left(\mathbf{r}, \mathbf{\Omega}', E'\right) + q_{ex}\left(\mathbf{r}, \mathbf{\Omega}, E\right), \tag{1}$$

and

$$\mathbf{\Omega} \cdot \nabla \psi\left(\mathbf{r}, \mathbf{\Omega}, E\right) + \Sigma_t\left(\mathbf{r}, E\right) \psi\left(\mathbf{r}, \mathbf{\Omega}, E\right) =$$
$$\int_{4\pi} dE' \int_{4\pi} d\mathbf{\Omega}' \Sigma_s\left(\mathbf{r}, \mathbf{\Omega}' \to \mathbf{\Omega}, E' \to E\right) \psi\left(\mathbf{r}, \mathbf{\Omega}', E'\right) +$$
$$\frac{\chi_p(E)}{K_{\text{eff}}} \int d\mathbf{\Omega}' \int_{4\pi} dE' \nu \Sigma_f(\mathbf{r}, E') \psi(\mathbf{r}, \mathbf{\Omega}', E'), \tag{2}$$

respectively. Both equations require finding the solution to the angular flux $\psi\left(\mathbf{r}, \mathbf{\Omega}, E\right)$ in addition to the latter also requiring the solution to the system's dominant eigenvalue K_{eff}. The angular flux is written in terms of its 6 dimensional phase-space of space $(\mathbf{r})$, angle $(\mathbf{\Omega})$ and energy (E). The terms Σ_t, Σ_s and Σ_f denote the total, scatter and fission macroscopic cross-sections of the hosting media. The variables ν represent the average number of neutrons born through fission and $\chi_p(E)$ denotes the fission spectrum.

4

The spatial and angular treatments are independent of the work presented here as the adaptive and immersed body procedures can be applied to arbitrary discretisations, provided that the spatial treatment is mesh based. In the work presented the angular treatment will vary between the traditional S_N and P_N methods whilst the spatial discretisation will use a sub-grid scale FEM approach [36]. Energy discretisation is via the standard multi-group method that solves a finite number of fluxes that represent and span the energy range of interest. For both equations, when the space, angle and energy dimensions are discretised using any arbitrary method, the resulting linear systems can be written as,

$$(A + H)\Phi = S \tag{3}$$

and

$$(A + H)\Phi = \frac{1}{K_{eff}}F\Phi \tag{4}$$

where A and H represent the discretised advection (including the boundary treatment), and scattering-removal terms. The terms S and F denoted the discretised external sources and fission operators, and Φ is the discretised angular flux. In both equations the spatial dependence arises from the cross-sections that are contained within the terms H and F. How these are represented within the FEM formulation and how the spatial mesh is selected is the focus of this article and the following sections detail how this is achieved, regardless of the spatial or angular treatment used.

3. The Immersed Body Method

The immersed body method allows a problem to be modelled efficiently by relaxing the requirement to use a mesh that conforms to all geometric detail of the domain. In all that follows, the BTE is solved upon a global mesh M that is arbitrary in the sense that it covers the problem's entire domain, but has elements that may overlap with any internal components the problem contains. The internal components are then represented by separate meshes m_i, $i \in \{1, \ldots, c\}$, where c denotes the number of components. These meshes are projected onto the global mesh M through a supermeshing algorithm [22]. This provides each element of the global mesh information of how much of the internal component it contains. An illustration of this projection is presented in figure 1 where the blue mesh denotes the global mesh M, and the red mesh an immersed body representing a 2D fuel pin. The supermeshing algorithm projects the pin's mesh onto the global mesh so that each blue element knows the exact fraction of materials it contains. It is worth mentioning that the projection can also map information of surface area and position of mesh m_i onto M, although this is not required in this article but will be used in future developments. For this consideration the total volume (mass) in the internal meshes are conserved, and the projection can be executed efficiently in log-linear time.
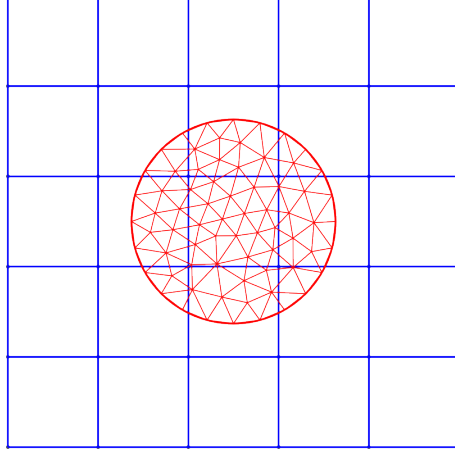
Figure 1: Example of immersed body projection: red mesh representing a 2D pin projected on a non-conforming global (blue) mesh.

Although any geometric information of an internal body can be conservatively mapped onto the global mesh, here it is sufficient to consider just the volumetric information. For a given element e of the global mesh M, the volume $V(e)$ occupied by an internal body, i, can be defined as,

$$V(e) = \sum_{f \in I(e)} \mu(e \cap f). \tag{5}$$

Here $\mu(.)$ represents the volume of the intersection between elements $e \in M$ and $f \in m_i$. The set $I(e)$ denotes the elements of m_i that intersect with global element e, i.e. $I(e) = \{f, \ f \cap e \neq \emptyset\}$. The article [22] describes how both the set $I(e)$ is efficiently constructed and the intersection volume computed; the details are therefore omitted here. Volume fractions are easily calculated from the supermeshing procedure from which appropriate material cross-sections can be formed for each element. That is, for each element e on the global mesh M, material cross sections are defined by,

$$\Sigma_t^e = \alpha_b^e \Sigma_{t,b}^e + \sum_{i=1}^{c} \alpha_i^e \Sigma_{t,i}^e, \qquad \Sigma_s^e = \alpha_b^e \Sigma_{s,b}^e + \sum_{i=1}^{c} \alpha_i^e \Sigma_{s,i}^e, \qquad \Sigma_f^e = \alpha_b^e \Sigma_{f,b}^e + \sum_{i=1}^{c} \alpha_i^e \Sigma_{f,i}^e, \tag{6}$$

where α_i^e denotes the volume fraction of component i occupying the element, and α_b^e the volume fraction of the remaining background volume. The corresponding material cross-sections of each component, i, and background material are denoted by $\Sigma_{.,i}^e$ and $\Sigma_{.,b}^e$, respectively.

4. Mesh Adaptivity procedure

With material cross-sections defined in the previous section, the BTE, equations 1 or 2, can be solved over the global mesh M using any finite element spatial discretisation and angular approximation scheme. In this article, the spatial discretisation is based on a sub-grid scale finite element method described in [36]. This formulation has a unique capability that enables an arbitrary angular discretisation scheme to be used and this is due to it being built within a Riemann framework that enables the advection of particles and the boundary conditions to be resolved irrespective of the angular treatment. This enables one to choose the most appropriate scheme for a given problem, and in this article both spherical harmonic and discrete ordinate solutions are presented.

With a given angular and spatial discretisation, the discrete solutions to the angular flux ψ on the global mesh M can be obtained by solving equations 1 or 2. In the following subsections these solutions are used to form a error metric that guides mesh refinement in order to produce new solutions with improved accuracy. The following subsections describe how the error metric is formed, the mechanics of the mesh adaptation procedure and the integration of the adaptivity with the immersed body method.

4.1. Anisotropic Error Metric

Mesh adaptivity is guided by anisotropic error metrics. For a given mesh M, one resolves the flux by solving either equations 1 or 2, and from this the metric is formed to determine the solution error. A separate error metric is formed for each element to determine the error it introduces, and these are based on a Hessian tensors. The mechanics of the metric can be explained by considering the scalar flux $\phi(\mathbf{r}) = \int \psi(\mathbf{r}, \mathbf{\Omega}) d\Omega$ for a single energy group obtained from the approximated solution on mesh M. This Hessian is defined as a 3×3 matrix H containing second order derivatives in its elements at some given position $\mathbf{r}$,

$$
H(\mathbf{r}) = \begin{bmatrix} \frac{\partial^2 \phi(\mathbf{r})}{\partial x^2} & \frac{\partial^2 \phi(\mathbf{r})}{\partial x \partial y} & \frac{\partial^2 \phi(\mathbf{r})}{\partial x \partial z} \\ \frac{\partial^2 \phi(\mathbf{r})}{\partial y \partial x} & \frac{\partial^2 \phi(\mathbf{r})}{\partial y^2} & \frac{\partial^2 \phi(\mathbf{r})}{\partial y \partial z} \\ \frac{\partial^2 \phi(\mathbf{r})}{\partial z \partial x} & \frac{\partial^2 \phi(\mathbf{r})}{\partial z \partial y} & \frac{\partial^2 \phi(\mathbf{r})}{\partial z^2} \end{bmatrix} .
\tag{7}
$$

This can be calculated from the approximated solution and the specific details are provided in [14]. However with this at hand one can utilise Cea's Lemma to form an upper bound of the approximation error over a given element which is denoted here as $\triangle$. The error is expressed through the scalar quantity given as,

$$
\epsilon_\phi = \|\phi - \Pi_h \phi\|_{\infty \triangle}.
\tag{8}
$$

This can be interpreted as being the maximum difference within the element between the flux ϕ and its interpolated linear projection onto the element, denoted here as $\Pi_h\phi$. This projection $\Pi_h\phi$ has the values of ϕ at the nodes of the element and varies linearly within the element's interior. The use of the infinity norm in equation 8, together with the $\triangle$ subscript, denotes that the measured norm is the maximum value of $\phi - \Pi_h\phi$ within the region spanned by $\triangle$. Applying the Lemma gives

$$\epsilon_\phi \leq \gamma \max_{\mathbf{r}\in\triangle} \max_{\mathbf{v}\in\triangle} \mathbf{v}'|H(\mathbf{r})|\mathbf{v} \leq \gamma \max_{\mathbf{e}\in E_\triangle} \mathbf{e}'|H|\mathbf{e}, \tag{9}$$

which holds provided the flux solution is sufficiently smooth - a full description of the above conditions can be found in [9]. The first inequality states that the error is bounded from above by the absolute Hessian $|H|$ (see equation 11), measured at some point $\mathbf{r}$ within the element, pre and post multiplied by an arbitrary vector $\mathbf{v}$ that is contained within the bounds of the element. The term $\mathbf{v}'$ denotes the vector's transpose. This is then multiplied by some constant γ. The matrix $|H|$ is defined from the eigenvalue decomposition of H,

$$H = R'\Lambda R, \tag{10}$$

where R is the 3×3 matrix containing the eigenvectors of H in its columns, and Λ is the 3×3 diagonal matrix containing the eigenvalues of H in its diagonal. The absolute matrix $|H|$ is defined as

$$|H| = R'|\Lambda|R \tag{11}$$

where the signs of the eigenvalues are made all positive. The first inequality in equation 9 is bounded from above by a second inequality which states the error is no larger that the absolute Hessian pre and post multiplied by the vector $\mathbf{e} \in E_\triangle$, where $E_\triangle$ contains the vectors aligned with the element's edges.

Whilst the metric provides a scalar measure of the quality of the solution over a given element, the use of the Hessian means one can still measure the solution quality with respect to direction. That is, the Hessian is dictating where and in which direction resolution is required, and this can be understood from the eigen-structure of H in equation 10. The matrix H defines a mapping that transforms the set of unit vectors forming the unit sphere onto a 3D ellipsoid with axes aligned in the directions defined by the eigenvectors R and with radii defined by the eigenvalues Λ. It is this ellipsoid that defines where the appropriate resolution is required; resolution will be proportional to the radii in each axial direction. As an example, figure 2 presents a 2 dimensional ellipsoid illustrating such an eigenvalue decomposition. The two eigenvectors are denoted as v_1 and v_2 and their corresponding eigenvalues are listed as λ_1 and λ_2 respectively. The eigenvectors define the axes of the ellipsoid and the eigenvalues define the respective radius. In this example, the eigenvalue decomposition produces a metric that indicates the higher spatial resolution is required in the direction of v_1 as this is where the largest second order solution gradients occur.
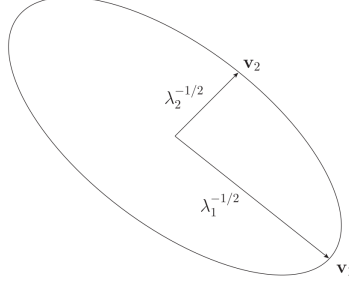
Figure 2: An ellipsoid resulting from an eigenvalue decomposition of some Hessian H. This illustrates the relationship of the ellipsoid and the eigenstructure of H where v_1 and v_2 denote the eigenvector and λ_1 and λ_2 the eigenvalues.

The metric supplied to the adaptivity algorithm that guides resolution to reduce the global approximation error is therefore based on the Hessian and given by,

$$\mathcal{M} = \frac{\gamma}{\epsilon} H. \tag{12}$$

In this article $\gamma = 1$ and ϵ is a user defined tolerance.

For multigroup calculations, a mesh will need to resolve a set of fluxes that may require different levels of resolution with varying elements sizes and orientations over the same location in space. To generate an appropriate element shape a single metric needs forming that guides sufficient resolution for all flux profiles. This is achieved by combining metrics in the form of equation 12 for individual energy groups into a single all-encompassing metric. The route taken in this article is to form a metric with an eigen-structure that forms the largest ellipsoid that fits within the ellipsoids formed by the metric in each energy group. For two metrics, say $\mathcal{M}_1$ and $\mathcal{M}_2$, this is accomplished by projecting one, say $\mathcal{M}_1$, onto the unit sphere using its eigen-structure $\mathcal{M}_1 = \mathcal{R}_1' \Lambda_1 \mathcal{R}_1$. The matrix is pre and post multiplied in the following way to perform the projection onto the sphere,

$$\tilde{\mathcal{M}}_1 = \Lambda_1^{\frac{-1}{2}} \mathcal{R}_1 \mathcal{R}_1' \Lambda_1 \mathcal{R}_1 \mathcal{R}_1' \Lambda_1^{\frac{-1}{2}} = I, \tag{13}$$

where I is the 3×3 identity matrix. The metric $\mathcal{M}_2$ is then projected in the same manner and its new eigenvalue structure formed,

$$\tilde{\mathcal{M}}_2 = \Lambda_1^{\frac{-1}{2}} \mathcal{R}_1 \mathcal{M}_2 \mathcal{R}_1' \Lambda_1^{\frac{-1}{2}} = \tilde{\mathcal{R}}_2' \tilde{\Lambda}_2 \tilde{\mathcal{R}}_2. \tag{14}$$

With both metrics $\tilde{\mathcal{M}}_1$ and $\tilde{\mathcal{M}}_2$ being transformed in an identical fashion from their respective original metrics, and with $\tilde{\mathcal{M}}_1$ now forming a unit sphere, it is a trivial to find the largest ellipsoid contained within the two. That is, a new metric $\tilde{\mathcal{M}}$ is formed that has an eigen-structure defined by an eigenvector matrix $\tilde{\mathcal{R}}_2$ and an eigenvalue matrix $\tilde{\Lambda}$, where $\tilde{\Lambda}_{i,i} = \max(1, (\tilde{\Lambda}_2)_{i,i})$. This new metric is defined by

$\tilde{\mathcal{M}} = \tilde{\mathcal{R}}'_2 \tilde{\Lambda} \tilde{\mathcal{R}}_2$ and is associated with the 3D ellipsoid that has axes aligned with the column vectors of $\tilde{\mathcal{R}}_2$ and radii that never exceed $\tilde{\mathcal{M}}_1$ or $\tilde{\mathcal{M}}_2$. The final process is to map the metric $\tilde{\mathcal{M}}$ back to the original space, thus reversing the operations in equation 14, and forming the combined metric $\mathcal{M}$,

$$\mathcal{M} = \mathcal{R}'_1 \Lambda_1^{\frac{1}{2}} \tilde{\mathcal{M}} \Lambda_1^{\frac{1}{2}} \mathcal{R}_1. \tag{15}$$

To combine more than two metrics the same procedure is applied an inductive manner.

4.2. Adaptive Mesh Functional

Given a metric $\mathcal{M}$ in the form of equation 15 one can now guide an adaptive mesh refinement algorithm that provides suitable resolution for the given problem. The details of this algorithm are quite complex and fully explained in [34] and so here only an outline is provided. Mesh refinement begins with the forming of a mesh quality functional, calculated from a quality measure for each element based on its metric $\mathcal{M}$. For an element e the quality metric is given as,

$$Q_{e,\mathcal{M}} = \frac{1}{2} \sum_{l \in L_e} d_l^2 + q_e^2 = F_e \tag{16}$$

which is used to form the eth component of the vector F. This is used to form the overall mesh quality functional, which is given as,

$$\mathcal{F} = \|F\|_p \tag{17}$$

where the p norm is set to $1, 2$ or ∞.

In equation 16, the two components forming the quality measure of an element relate to the appropriateness of its edge orientations and shape, respectively. For each edge l of the element's edge set, denoted as L_e, d_l is defined as $r_l - 1$, where $r_l = l' \mathcal{M} l$ defines the edge length with respect to the metric $\mathcal{M}$. By minimising d_l the length r_l will tend to 1 meaning that the edge length will be unity with respect to the element's metric. The second inequality q_e of equation 16 is defined as,

$$q_e = (\alpha/\rho_e - 1), \tag{18}$$

where $\alpha = 1/(2\sqrt{6})$ and ρ_e is the radius of the largest inscribed sphere of the element. Minimising this second quantity ensures that the element shape is unity with respect to the metric $\mathcal{M}$ - that is it has optimal aspect ratio for the given metric. Details of calculating both quantities in equation 16 can be found in [34].

10

4.3. Combined Adaptive Meshing and Immersed Body Procedure

Underpinning the adapting process, with a solution on some initial global mesh M, one can find a new adapted mesh M^* that reduces the quality functional defined in equation 17. The algorithm used is detailed in [34]. This selects the poorly performing elements of M that contribute the most to the functional, and adapts locally around their regions to form a new mesh with improved quality. As detailed in [34], this new adapted mesh M^* is generated from the current mesh M by collapsing edges, splitting edges, switching faces between elements and moving nodes. This is illustrated in figure 3, and further illustrations are given in [37]. These adjustments are tested one after the other, changing the mesh resolution locally about the poorly resolved areas, and their associated quality metrics then formed. If a new mesh M^* results in a reduced functional, it is accepted for use in finding the next flux solution before repeating the next adapting step.
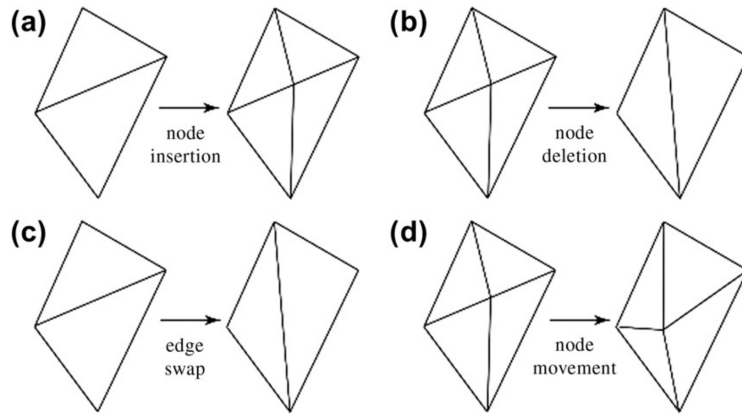
Figure 3: An illustration of the mesh adapting procedure through node insertion/deletion/movement and edge splitting of an element.

The combined adaptive mesh and immersed body method now follows the following procedures.

1. For some mesh M, the immersed body, represented by mesh M^b, is projection onto M (section 3).
2. Material cross-sections are formed on M and the flux solution found (solutions to equations 1 or 2).
3. The Hessian and error metric detailed in section 4.1 and 4.2 are formed, and a new mesh M^* is found that reduces the error metric.
4. Set $M = M^*$, go to step 1.

This is iterated until mesh convergence is reached.

5. Numerical Examples

In this section three numerical examples are presented to demonstrate the potential benefits in applying the Immersed Body Method with Adaptive mesh resolution to resolve neutron transport problems. Both fixed source and eigenvalue problems are resolved and the performance regarding accuracy and efficiency discussed. For this analysis the method is compared against the solutions when using the IBM without mesh adaptivity, and when standard FEM is used either with or without adaptive mesh resolution. Thus 4 methods are analysed here, their description and naming convention is listed in table 1 .

Name	Description
Immersed Body Adaptivity (IBA)	Uses IBM and adaptive mesh resolution
Immersed Body (IB)	Uses IBM but no adaptive mesh resolution
Geometry Conforming Adaptivity (GCA)	Uses adaptive mesh resolution with FEM meshes that conform to problem geometry
Geometry Conforming (GC)	Uses FEM with meshes that conform to problem geometry but without adaptive mesh resolution

Table 1: List of FEM methods analysed and their brief description.

5.1. Numerical Example 1: Absorbing Cylinder within a Box

This first numerical example demonstrates the IBA method's ability to resolve fixed source radiation problems (equations 1 and 3) containing immersed bodies and compares its performance with that of the GC and GCA methods. The problem domain, presented in figure 4, is a box with side lengths 16cm. It contains a cylindrical body, of length 3cm and radius 0.475cm, that is aligned with the vertical axis with the centre of its lower base positioned at point (3cm,3cm,1cm). A unit source of 1 neutron/cm^{2}sec occupies the entire domain and vacuum boundary conditions are applied to all sides. The material cross sections are selected so there is mild interactions within box ($\Sigma_t = 1.9$ cm^{-1} , $\Sigma_s = 0.89$cm^{-1}) and high absorption within the cylinder ($\Sigma_t = 100.9$ cm^{-1} , $\Sigma_s = 0.89$cm^{-1}).

This problem was solved through the IBA approach using the two angular discretisations P_1 and S_6. In both cases an initial mesh consisting of 384 tetrahedral elements was used, and the approach of projecting the cylinder, solving the RT problem and adapting the mesh was applied until mesh convergence was reached. The cylinder was represented as an immersed body via a fine FEM mesh consisting of 71997 tetrahedral elements. This large mesh was necessary to ensure the geometric detail of the cylinder and its curvature was adequately captured.
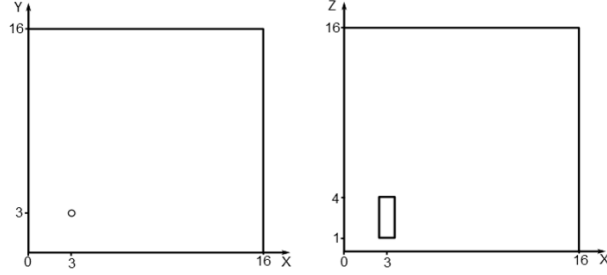
12

Figure 4: Diagram of the pin in box problem.

The IBA P_1 flux solutions and adapting meshes are presented in figure 5. These present the solutions through horizontal and vertical planes about the cylinder where a sharp attenuation in the flux occurs due to the high absorption of the neutrons. The results show the flux about the cylinder to be poorly resolved during the early stages of the adapting process whilst the mesh is still coarse around this region. However, as the mesh is adapted, elements begin to concentrate around the cylinder and resolve the detail of the solution more accurately. This is further illustrated in figure 6 presenting the flux solutions together with the cylinder, and the global mesh elements that intersect with the cylinder during the adapting process. The meshes in latter stages of the calculation are shown to capture most of the cylinder's detail. This implies that, in this numerical example, the immersed body is important to resolve (which stands to reason due to the large solution gradient) and that the adaptive method is reacting correctly to this. Similar behaviour was found with the S_6 discretisation.

A benefit in using the IBA method over normal adaptivity, GCA, is demonstrated in the graphs of figure 7. Figure 7a presents the mesh sizes for both methods during the adapting process. It is shown that when using GCA, a large initial mesh of 90K elements was used as this was necessary to capture the geometric detail of the cylinder. This mesh was selected to appropriately suit the problem, where a substantial proportion of the elements (70K) were used to resolve the cylinder and the remaining elements (20K) to coarsely resolve the rest of the domain (this reduced the mesh size whilst resolving geometric detail). The GCA scheme then underwent a sharp drop in mesh elements following the first adapt. This implies computational effort was wasted on the first solve by using a mesh that over resolved certain regions. Following this the GCA meshes then began to increase in size as the mesh resolution is appropriately added. Conversely the IBA method begins by solving on coarse meshes that increase in size as resolution was added where required. Therefore at no stage in the solving process is effort wasted through the use of meshes that over resolve certain regions. Both methods are shown to generate meshes that converge in size following 5 or 6 adapting steps, but it is also shown the IBA approach requires a smaller mesh. The additional resolution used by the GCA scheme was due to the enforced constraint of preserving the cylindrical region defined by the original mesh. It means that resolution is being applied in order to capture the detail of the cylinder and not to improve the
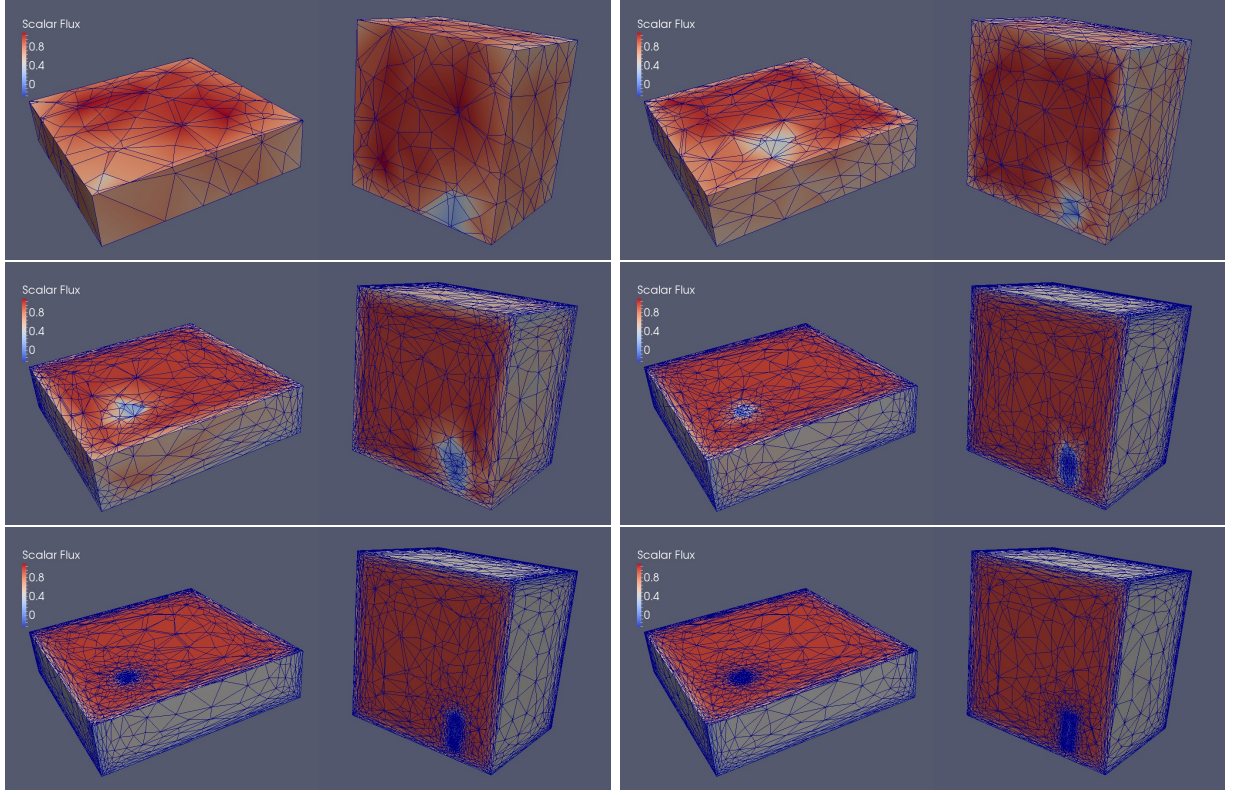
Figure 5: The IBA P_1 scalar flux solution upon the first 6 adapted meshes. Ordered left to right, top to bottom.

quality of the solution. This does not occur using the IBA scheme, thus allowing for smaller meshes that are more efficient to solve upon.

The graph presented in figure 7b shows the cumulative computational times for the IBM projection, the mesh adapting, and the linear solver for each adapting step of the IBA. These are compared against the times for computing the mesh adapting and linear solve for the GCA calculation. These results were obtained from the S_6 calculations. It is shown that the costs of the IBM projections are around half that of the adapting procedure, thus only modest increases in time is incurred when using the IBM. The graph also shows that the combined computing times for the IBM projection and mesh adapts, for the IBA method, is less than the mesh adapting computing time for the GCA, and this is due to the GCA forming larger meshes. The IBA solver times are also similar to the projections and adapting procedures combined. Importantly the total costs of the IBA method is around 1/3 smaller of that of the total time in the GCA calculation. This demonstrates that computational resources regarding both memory (due to smaller meshes) and time can be reduced without affecting accuracy.
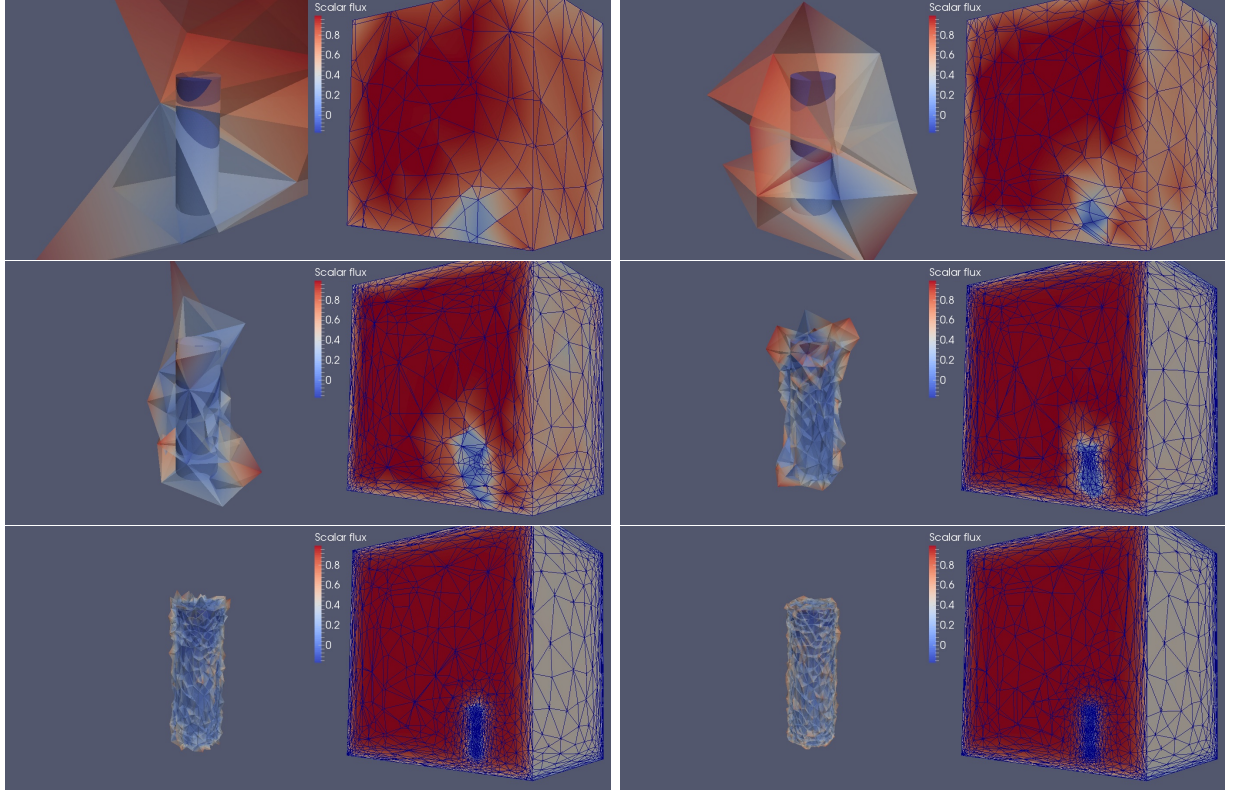
Figure 6: The IBA P_1 scalar flux solutions upon the first 6 adapts (left to right, top to bottom). In each solution the image of the fluxes are shown around the cylinder region in addition to the global mesh elements that intersect with the cylinder's geometry.

Some advantages of the IBA method over the GC method are demonstrated in the graphs in figure 8 a-c. These present the P_1 scalar flux solution errors along the vertical line intersecting the centre of the cylinder between positions (1,1,0)-(1,1,8) (this region should exhibit the largest solution errors due to the large solution gradients). Graph (a) presents 5 solution errors using the GC method with uniform resolution meshes of size 6K, 13K, 24K, 48K and 94K elements. Graphs (b) and (c) present the evolving solution errors (i.e. the errors at each adapting step) from two IBA calculations which were generated using varying adapting tolerances (equation 12) set to allow for approximately 65K and 150K elements to form, these are labeled IBA1 and IBA2, respectively. For these results comparisons are drawn against an "exact" numerical solution calculated upon a sufficiently fine spatial mesh. The graph shows the regions with the largest solution errors to be located over the cylinder's boundaries - from position 1cm to 4cm. In the IBA results, large errors also formed across the cylinder (between positions 1cm-4cm) and between the cylinder and bottom edge (between positions 0cm-1cm) during the early stages of adapting. However as the meshes evolved these errors were largely mitigated through
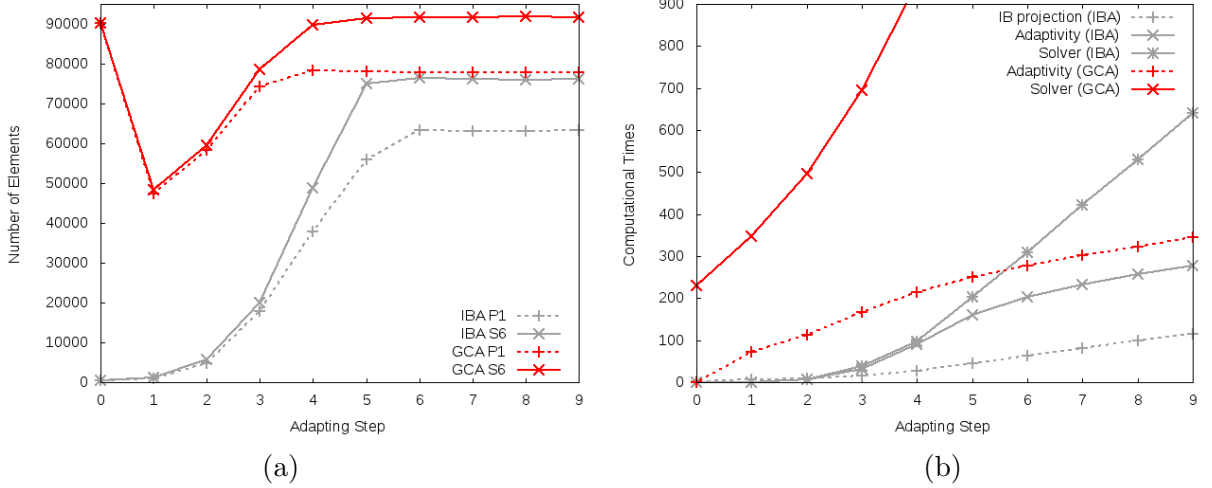
Figure 7: (a) Mesh sizes during the adapting process for the IBA and GCA methods using S_6 and P_1 angular discretisations. (b) Computational times of the S_6 calculations during the adapting process using the IBA and GCA methods. The figures stated show processing times for the IBM projection (labelled IBM projection), the mesh adaptivity calculation (labelled Adaptivity) and the linear solver times (labelled Solver).

local refinement. In comparison, the solutions from GC were consistently poor, even for the largest mesh, since the use of uniform mesh meant few elements being placed around these relatively small, but important, regions. The graph of figure 8d presents the scalar flux L_2 error norms, measured along the line (1,1,0)-(1,1,8). The GC result shows the errors from each of the 5 varying resolution meshes. The IBA results show the errors of the solution on each mesh as it evolves during the mesh adapting process. It is shown that the GC errors are practically stagnant with increasing mesh resolution, indicating that enormous meshes would be required for accurate solutions. The IBA results show the error to remain flat during the first few adapts but to then sharply reduce as the mesh beings to converge. Mesh convergence occurs around the 6th adapt, for both calculations, at which point large reductions in errors are made without increasing the mesh size.

5.2. Numerical Example 2: Fuel Pin Problem

This numerical example demonstrates the potential of the IBA method to resolve eigenvalue problems (equations 2 and 4) efficiently and accurately in comparison to the other schemes. In addition it also highlights the potential benefits the IBA brings in terms of avoiding the non volume conserving meshes due to curved surfaces. The problem is a single pressure water reactor (PWR) fuel pin as presented in figure 9. The pin is 0.475cm in radius which is contained within a box of width and
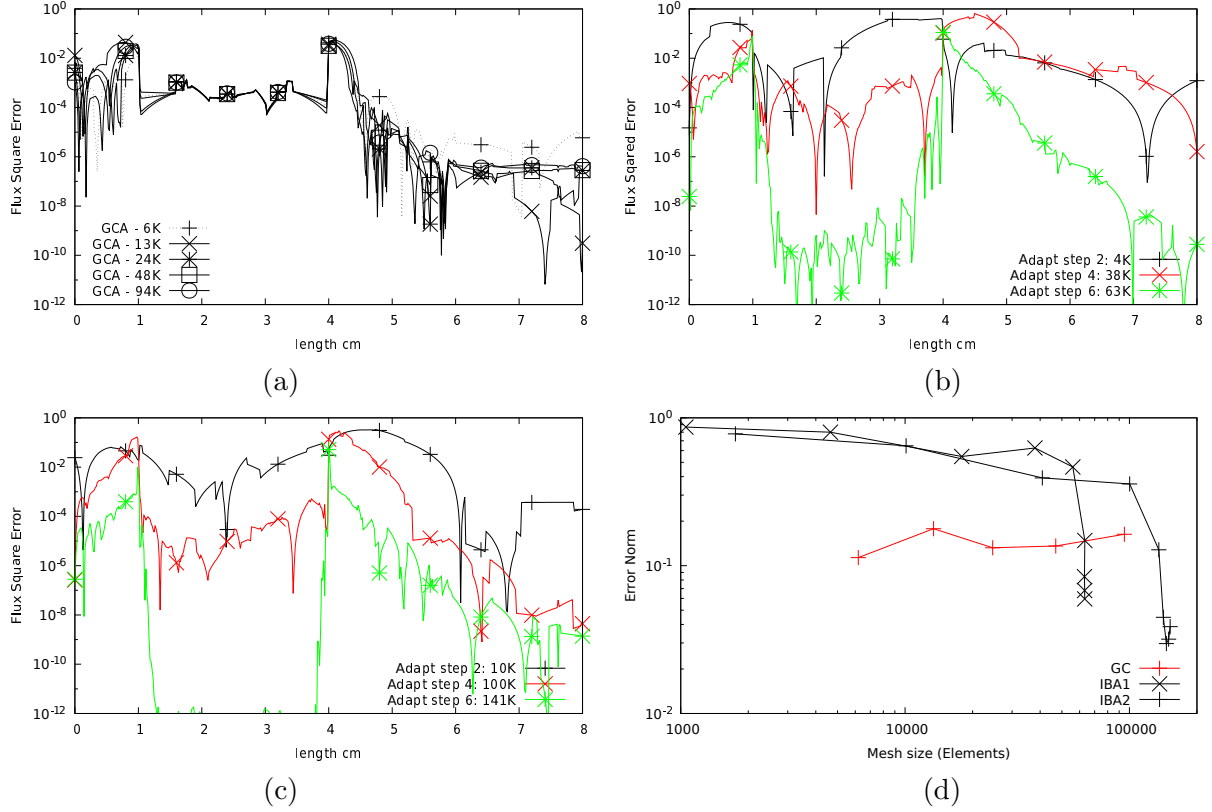
16

(a)

(b)

(c)

(d)

Figure 8: Figs a-c: Scalar flux solution errors along the line (1,1,0)-(1,1,8). a) Using GC method with mesh of uniform resolution, b) IBM Adaptive mesh allowing up to ∼65K elements - labelled IBA1, c) IBM Adaptive mesh allowing up to ∼150K element - labelled IBA2. d) Error norm measured along the line (1,1,0)-(1,1,8) using the IBM adaptive and uniform meshes.

depth 1.26cm, and height 2cm. For simplicity the pin contains the regions that would normally be occupied by the fuel, the clad and the fuel-clad gap; this is combined into a single material and is surrounded by a moderating body of water. Reflective boundary conditions are applied to all surfaces meaning the problem represents an infinite array of fuel pins separated by a 1.26cm pitch, and which is infinite in the vertical axis. A 2 group material cross section data set was generated through the WIMS-ANSWERS software [38] to represent the fuel and moderator regions. Here an infinite array of fuel pins was solved using the MOC CACTUS module in conjunction with a 172 energy group cross-section data, and condensed to form the representative fast and thermal energy ranges. These material cross-sections are listed in table 2.

In the following analysis all IBA calculations used an initial global mesh consisting of 4623 elements
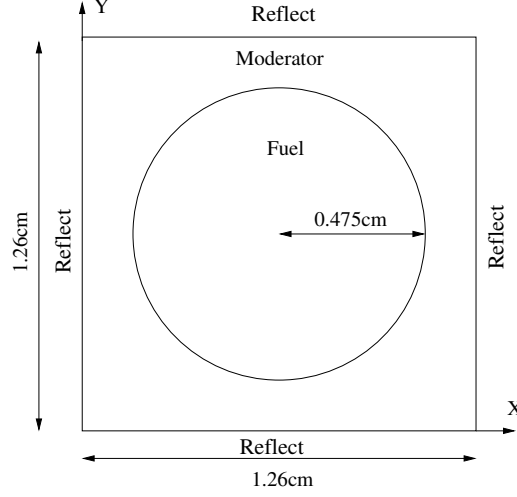
17

Figure 9: The problem domain of the fuel pin problem. The geometry is based on standard PWR fuel but without the fuel-clad gap.

	fuel		Moderator	
Group	1	2	1	2
Fission Spectrum	9.996266E-01	3.723757E-04	0.0	0.0
Σ_a	6.002802E-03	9.583142E-02	1.447155E-04	6.584086E-03
$\nu\Sigma_f$	2.744700E+00	2.438048E+00	0.0	0.0
$\Sigma_s(1 \rightarrow g)$	3.363515E-01	1.336288E-03	3.395332E-01	5.954274E-02
$\Sigma_s(2 \rightarrow g)$	0.0	4.048443E-01	0.0	1.278956E+00

Table 2: The 6 energy group material properties of the fuel pin.

to start the solving process - the pin, being the immersed body, was represented by a fine mesh of $\sim$300K elements. Six calculations were performed with varying error tolerance settings, see equation 12, and their meshes were given freedom to evolve in size and resolution to satisfy this parameter - these ended up ranging in size from 8K-55K elements. For comparison with the GCA method, a similar set of calculations were conducted with parameters (equation 12) set so they also evolved meshes with approximately the same range of 8K-55K elements. These calculations were initialised using four meshes formed of approximately 20K, 50K, 110K and 400K elements, all of which conserved the fuel pin's volume. For comparison with the IB method, 5 calculations were performed using global meshes that varied in size from 2000 to 50000 elements. Five calculations using the GC approach were also made with similar sized meshes that ranged in size from 2000 to 50000 elements. For these GC results two mesh types were used. The first mesh type used the exact pin radius in CAD (0.475cm) but with a resulting mesh that failed to conserve the pin's volume exactly due to the straight edged

element's representation of the pin's curved surface. The second mesh type used an adjusted pin's radius which resulted in a mesh that did preserve its volume. In all the proceeding analysis an S_8 angular discretisation was used. A reference eigenvalue solution, to within 5 decimal places, is given as $K_{eff} = 1.19142$.

The graph presented in figure 10 shows the converged eigenvalue errors as a function of mesh size using the various schemes under consideration. The result demonstrates the importance in using a volume conserving mesh as the eigenvalue error in the non-conserving GC method are orders of magnitude larger than the other schemes. Applying the IBM approach, without adaptivity, can mitigate this error by around an order of magnitude whilst the GC method, using a conserving mesh, can provide almost another order of magnitude reduction. However the IBA is shown to provide the most accurate prediction of the eigenvalue and it shown to be able to reduce the error by a further 75% for the same mesh size. Thus the IBA provides substantially more accurate predictions and mitigates the need to form volume conserving meshes.
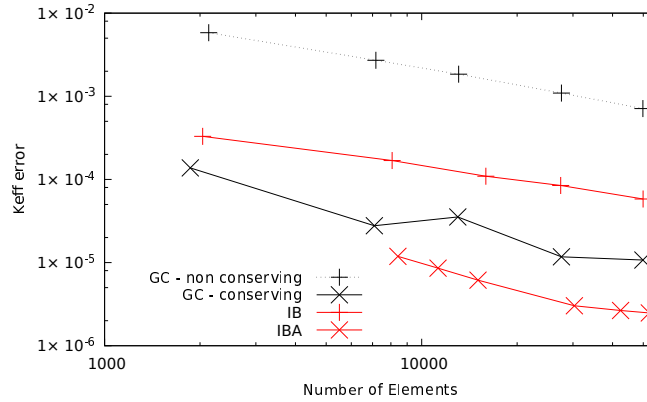

Figure 10: Eigenvalue errors comparing GC with/without conserving meshes to IB and IBA.

The graph presented in figure 11 present the eigenvalues from the adaptive IBA and GCA methods. The results show the converged eigenvalues against mesh resolution (number of elements) when the error tolerances are decreased. The graph includes the results of the GCA when initialised with the 4 different meshes of varying resolution. A common feature in all results is that as the error tolerance is decreased, the mesh sizes increase and the eigenvalues converge to some value. However it is apparent that all calculations are converging to different eigenvalues, and that the GCA eigenvalue is highly dependent on the initial spatial mesh. This variation is due to an inherent feature of the GCA procedure where it is forced to use the initial mesh to define the problem's geometry. That is, once the initial mesh is set, the slight differences between the meshed geometry and the true geometry

19

will remain within the GCA calculation. The GCA method will therefore never converge to the true solution until a sufficiently fine initial mesh is used to resolve the geometry adequately.
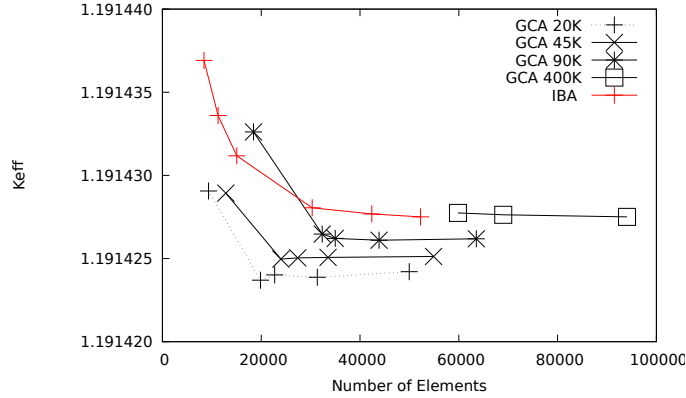

Figure 11: Eigenvalues of IBA and GCA for varying error tolerances and initial meshes. The values following the GCA indicate the initial mesh size.

Importantly the IBA method does not inherit the volume conserving issue as the pin's geometry is adequately represented through the immersed body using an extremely fine mesh. The solutions from the IBA method will therefore converge to the true solution as the error tolerance is tightened. This is supported from results of the GCA method converging towards the IBA solutions as the initial mesh size is increased - as shown in figure 11. In fact the solution using an initial mesh of 400K elements is in very good agreement with that of the IBA method for which the meshes only required 50K elements. This is opposed to the GCA method that required an initial mesh of 400K elements that was then reduced to around 90K elements through adaptivity. However resolving the problem on these large meshes has a profound impact on computational efficiency as run times were drastically increased using GCA. This is demonstrated in the graph of figure 12, showing run times and meshes sizes during the adapting process. Two calculations are presented, using IBA and GCA, that resulted in similar sized meshes. It shows the GCA solving times, initialised with a 110K element mesh, is dominated by the initial solves of the adapting process. One can see that after 3 adapting steps, the meshes have converged in size, but the solver time is a factor of 3 smaller using IBA.

Figure 13 presents the evolving meshes and solutions of the IBA calculation. They are shown to form highly anisotropic elements that place fine resolution through the problem's horizontal directions, and coarse resolution through its vertical. This would seem desirable since the flux solution to this particular problem would have no variation through the z-axis, therefore little effort is being used in resolving this part of the problem. The meshes also form anisotropically through the x-y plane, where
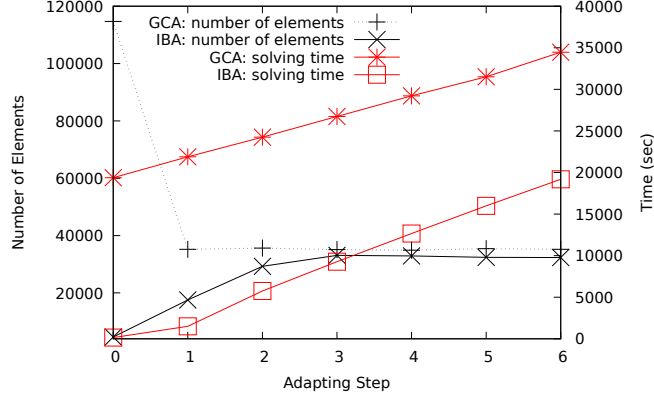
Figure 12: Mesh and run times of the IBA and GCA method during adaptivity.

highest resolution is devoted to resolving the directions between the fuel and moderator at the pin's interface. Again this is desirable as the largest flux variations occur at these positions and through these directions. This demonstrates the advantage in allowing the complete freedom of the meshes' shape, as it has enabled the use of highly anisotropic elements that resolve specific directions over specific regions. It is this that has led to the large reductions in mesh size whilst maintaining solution accuracy.

5.3. Numerical Example 3: The SUPO Reactor

This numerical example demonstrates how the IBA method can be utilised to solve reactor physics problems that would be problematic and computationally expensive to resolve using standard FEM approaches. It shows how domains containing internal bodies with intricate detail can be resolved efficiently by avoiding the issues already highlighted regarding the need for large initial meshes to capture geometric detail. In this demonstration the SUPO reactor is resolved (this is an eigenvalue problem involving equations 2 and 4), a picture and CAD design using Cubit [39] is presented in figure 14, other views can be found in [20, 40]. As this problem has been resolved previously in [20, 40] the full details of the geometry and materials will not be repeated here. However the reactor had a spherical core (approximately 30cm in radius) containing a uranium nitrate liquid fuel that was contained within steel vessel, and surrounded by graphite reflector. The fuel was a mix of water and uranium nitrate and for this demonstration the liquid fuel was set to a height of 5cm above the centre point of the reactor. The reactor core contained several internal structures including three helical cooling coils, two control rod sleeves and a neutron beam generator, these can all be seen in figure 14. It can be appreciated from the CAD design that a mesh that conforms to the geometry and which
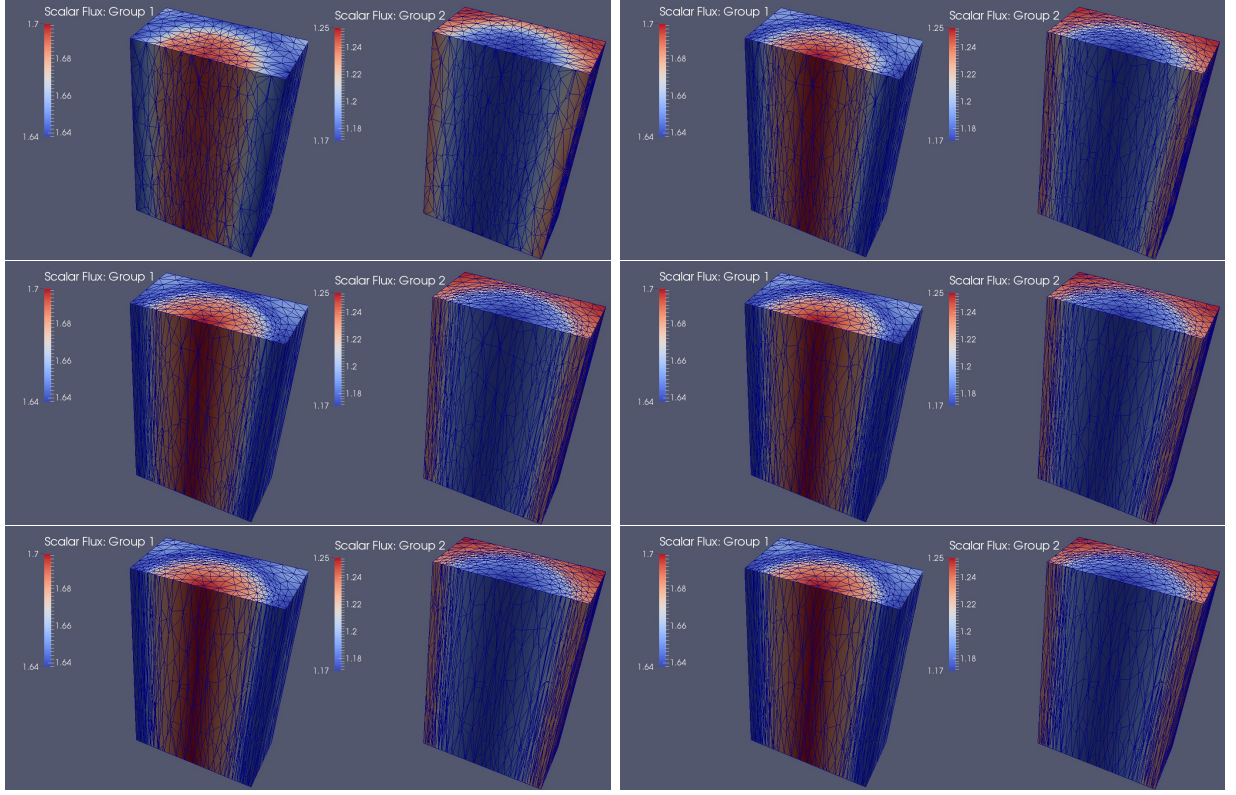
21

Figure 13: Scalar flux solutions and adapted meshes (left-right, top-bottom) for the S_8 approximation of the fuel pin problem.

conserves all component volumes would be extremely large. It can also be appreciated that due to the size of such a mesh, one would expect large solving times using methods such as GC or GCA. The purpose of this numerical example is therefore not about comparing all the method types, but rather show that these problems can be computed efficiently through the IBA method.

For the IBA calculation, the global mesh was made to conform to the spherical fuel region, steel vessel and outer graphite reflector, whilst the coils, rods and the neutron beam generator were represented as immersed bodies. Accurate representation of the immersed bodies required meshes with approximately 60K elements for the cooling coils and 5-10K elements for the control rods and beam generator. The adapting process was initiated from a global mesh consisting of approximately 270K elements which applied near uniform resolution across the entire domain. Using a P_1 angular approximation the reactor was resolved with control rods fully inserted which resulted in a K_{eff} value of 0.96130 when mesh convergence was reached. This value of reactivity is in line with the other results of [20, 40] but

there is no benchmark solution for comparison. Mesh convergence was reached within 5 mesh adapts which resulted in using approximately 2.6M elements. The resulting scalar flux solutions, together with their meshes, formed during the adapting process are presented in figure 15. It can be seen that different regions of the reactor require substantially different levels of mesh resolution. One region requiring the highest resolution is that at the interface between the fuel, steel vessel and graphite reflector. Here the flux undergoes a rapid change as neutrons are scattered from high to low energies as they flow into the reflector. Appropriately, anisotropic elements have also formed, applying the highest resolution in the directions through the material interface.

When resolving the internal components, figure 15 shows that IBA focuses high resolution near the regions containing the control rods and neutron beam generator. This is more clearly illustrated in the figures plotting mesh vertices. Adapting in this manner is preferable as these components attenuate the flux, causing sharp variations that require high resolution to be resolved accurately. The elements around these regions are also highly anisotropic, with the highest resolution being directed through the body surfaces where the solution gradients are largest. An interesting feature of these solutions is that the region containing the cooling coils and fuel requires approximately the same resolution. The strong similarities between the fuel and coils (i.e. the high water content) mean gradual changes in the flux occur, thus only moderate resolution is required. This important feature demonstrates that IBA resolves regions and components to the level necessary; the GC or GCA would have required significantly more resolution to capture the geometric detail. Figure 16 further illustrates this by showing the global mesh elements that intersect with the structures. It is shown that the control rod has the highest resolution followed by the guide tube, but the cooling coil is coarsely represented in comparison. Here only 15K elements resolve the coil which is significantly less than the ~60K elements required to capture its geometry.

6. Conclusions

This article has presented the first method that combines self-adaptive finite elements with an immersed body projection method to enable the efficient solving of particle transport problems containing complex internal structures. The efficiency gains, benefitting both run times and memory, were realised through the relaxing of the mesh requirement of needing to conform to the detail of the problem's geometry. This was achieved through the immersed body projection approach which conservatively mapped the geometric detail of internal components onto arbitrary meshes upon which the radiation field was solved. The projection informed each mesh element of the materials it contained, thus appropriate cross-sections could be computed. Blending this with a self-adaptive meshing method, where the model determines where and how the resolution is placed, enabled optimal meshes to be formed
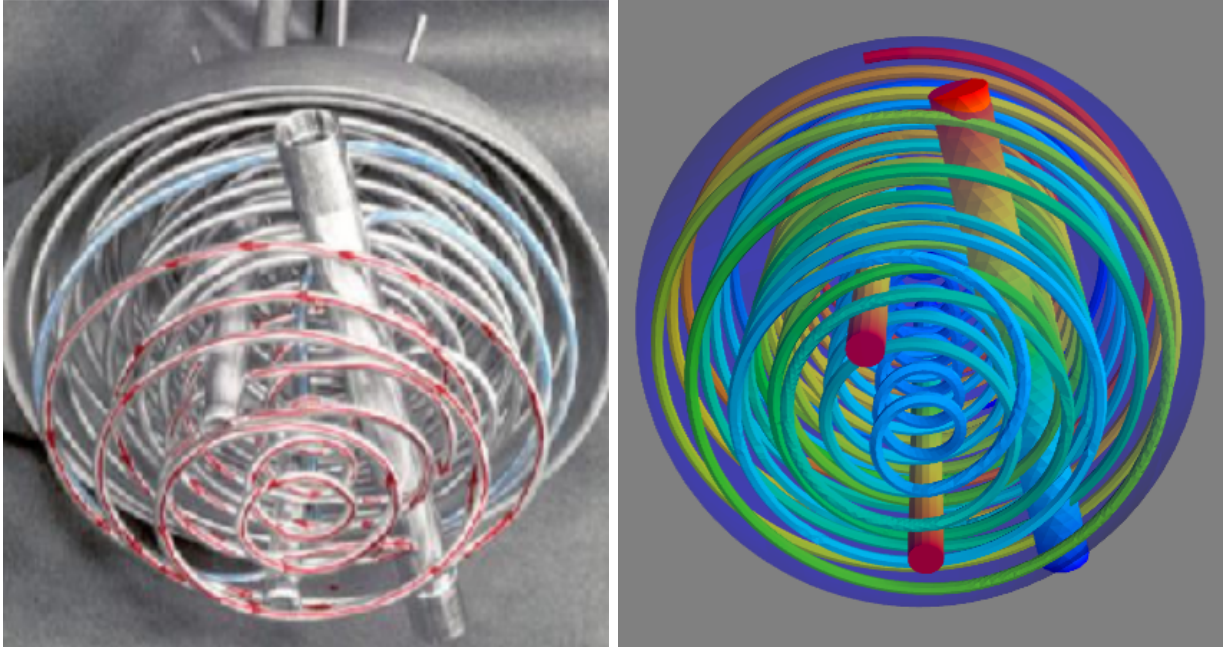
Figure 14: A picture and CAD design of the SUPO reactor, viewed from below. The core was a spherical shaped and contained internal components including cooling coils, control rods and beam generators.

that applied resolution specifically to mitigate solution errors.

The three numerical examples presented here were designed to demonstrate the benefits the IBA method can provide. There was clear evidence that in comparison to using uniform resolution meshes, with either GC of IB, the accuracy and solution convergence is significantly improved with IBA. This was particularly the case for problems that required high mesh resolution to only isolated and localised regions - e.g. numerical example 1. It was also demonstrated that the method can be advantageous over the GCA and GC approaches. Firstly, the use of arbitrary meshes meant smaller meshes were formed when using IBA. This stems from GCA always needing to conform to the components of a problem and being forced into maintaining this resolution regardless of whether it was needed. Ultimately this led to reduced computational times using IBA to generate solutions of comparable accuracy to GCA. Secondly, IBA avoided the use of large initial meshes that might be required by GC and GCA in order to start the solving process. These were necessary to resolve all geometric detail and often over applied resolution to certain regions; in turn this lead to increased solving times during the initial stages of adapting. Thirdly the IBA overcomes the issue of creating volume conserving meshes for GCA and GC solutions. It is a well known fact that volume conserving meshes are important in radiation problems, but generating such meshes can require substantial time and effort if meshing

tools with such capabilities are unavailable. These properties of the IBA method enables complex radiation transport problems to be resolved that could otherwise be too computationally expensive using conventional methods.

Future directions will be to integrate this technology with goal based adaptivity, where mesh resolution adapted to a specific goal rather than for the global reduction of errors presented here. The technology also lends itself to efficient coupled radiation-fluid calculations; the IBA has already benefited the fluid flow community as demonstrated in the articles cited throughout this paper.
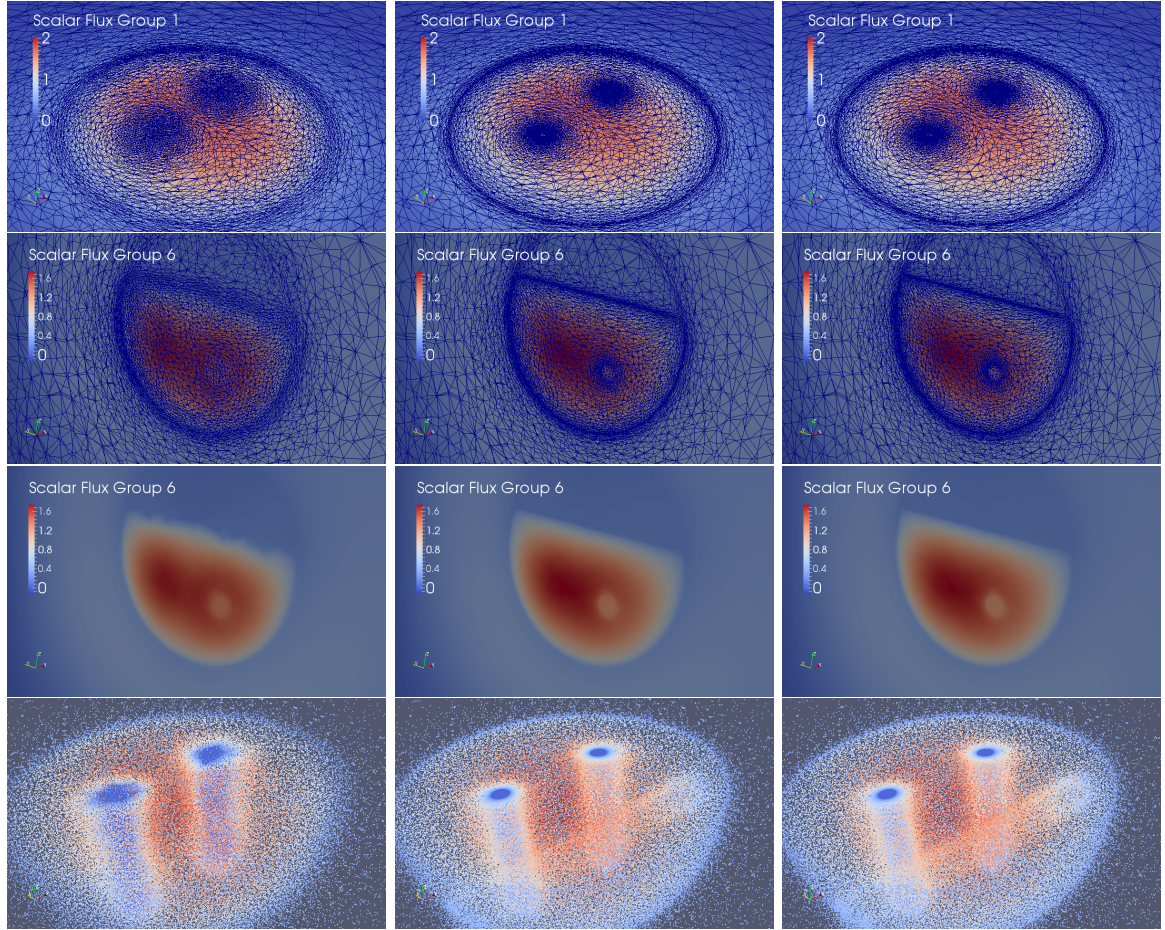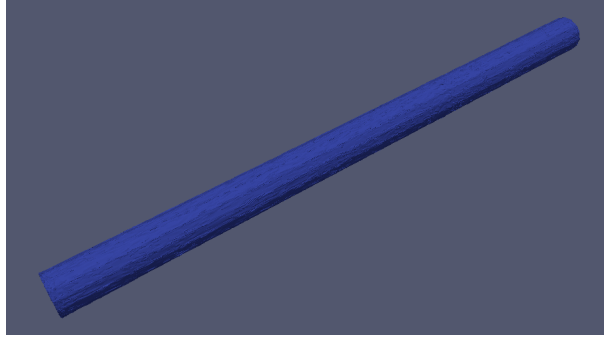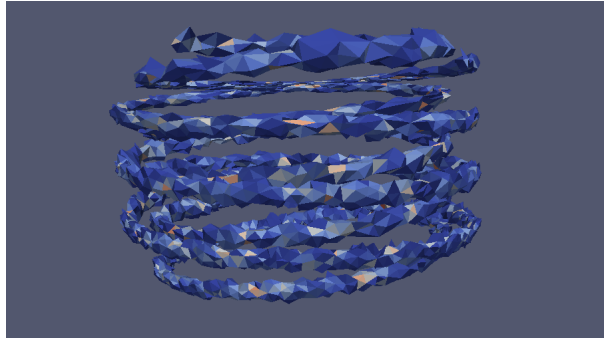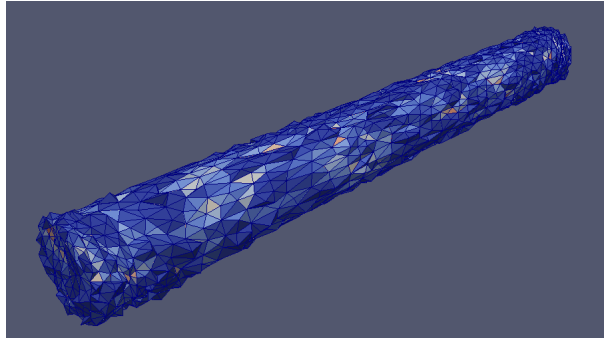
7. ACKNOWLEDGEMENTS

Figure 15: Adaptive mesh solutions of the SUPO reactor. The columns show solutions upon adapted meshes 1,3 and 5 respectively. Row 1: Fast flux with mesh, Row 2: Thermal flux with mesh, Row 3: Thermal flux without mesh: Row 4 mesh nodes.

a)

b)

c

Figure 16: Filtered elements showing only those intersecting the immersed bodies on the final adapted mesh. a) control rod, b) outer cooling coil c) beam generator

References

[1] E. E. Lewis and W. F. Miller. *Computational Methods of Neutron Transport.* American Nuclear Society, 1993.

[2] J. J. Duderstadt and W. R. Martin. *Transport Theory.* John Wiley and Sons, NY, 1979.

[3] J. J. Duderstadt and L. J. Hamilton. *Nuclear Reactor Analysis.* John Wiley and sons, NY, 1976.

[4] J. E. Morel and J. M. McGhee. A self-adjoint angular flux equation. *Nuclear Science and Engineering*, 132:312–325, 1999.

[5] R. T. Ackroyd. *Finite Element Methods for Particle Transport: Applications to Reactor and Radiation Physics.* Research studies press, Taunton, Somerset, England, 1997.

[6] C.R.E. De Oliveira. An arbitrary geometry finite element method for multigroup neutron transport with anisotropic scattering. *Progress in Nuclear Energy*, 18(12):227 – 236, 1986.

[7] L. Cao and H. Wu. A spherical harmonicsfinite element discretization of the self-adjoint angular flux neutron transport equation. *Nuclear Engineering and Design*, 237(23):2232 – 2239, 2007.

[8] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.

[9] J. C. Ragusa. A simple hessian-based 3d mesh adaptation technique with applications to the multigroup diffusion equations. *Annals of Nuclear Energy*, 35(11):2006 – 2018, 2008.

[10] Yaqi Wang, Wolfgang Bangerth, and Jean Ragusa. Three-dimensional h-adaptivity for the multigroup neutron diffusion equations. *Progress in Nuclear Energy*, 51(3):543 – 555, 2009.

[11] D. Lathouwers. Spatially adaptive eigenvalue estimation for the SN equations on unstructured triangular meshes. *Annals of Nuclear Energy*, 38(9):1867 – 1876, 2011.

[12] D. Lathouwers. Goal-oriented spatial adaptivity for the SN equations on unstructured triangular meshes. *Annals of Nuclear Energy*, 38(6):1373 – 1381, 2011.

[13] M. A. Goffin, C. M. J. Baker, A. G. Buchan, C. C. Pain, M. D. Eaton, and P. N. Smith. Minimising the error in eigenvalue calculations involving the boltzmann transport equation using goal-based adaptivity on unstructured meshes. *Journal of Computational Physics*, 242(0):726–752, 2013.

[14] C. M. J. Baker, A. G. Buchan, C. C. Pain, B. S. Tollit, M. A. Goffin, S. R. Merton, and P. Warner. Goal based mesh adaptivity for fixed source radiation transport calculations. *Annals of Nuclear Energy*, 55(0):169 – 183, 2013.

[15] M. Chin K. Manalo and G. Sjoden. Cartesian meshing impacts for pwr assemblies in multigroup monte carlo and sn transport. In *Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo*, 2014.

[16] D. Moxey, M.D. Green, S.J. Sherwin, and J. Peir. An isoparametric approach to high-order curvilinear boundary-layer meshing. *Computer Methods in Applied Mechanics and Engineering*, 283:636 – 650, 2015.

[17] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194:4135–4195, 2005.

[18] S. K. Hall, M. D. Eaton, and M. M. R. Williams. The application of isogeometric analysis to the neutron diffusion equation for a pincell problem with an analytic benchmark. *Annals of Nuclear Energy*, 49:160 – 169, 2012.

[19] A.R. Owens, J.A. Welch, J. Kópházi, and M.D. Eaton. Discontinuous isogeometric analysis methods for the first-order form of the neutron transport equation with discrete ordinate (sn) angular discretisation. *Journal of Computational Physics*, 315:501 – 535, 2016.

[20] A. G. Buchan, P. E. Farrell, G. J. Gorman, A. J. H. Goddard, M. D. Eaton, E. T. Nygaard, P. L. Angelo, R. P. Smedley-Stevenson, S. R. Merton, and P. N. Smith. The immersed body supermeshing method for modelling reactor physics problems with complex internal structures. *Annals of Nuclear Energy*, 63(0):399 – 408, 2014.

[21] P. E. Farrell, M. D. Piggott, C. C. Pain, G. J Gorman, and C. R. G. Wilson. Conservative interpolation between unstructured meshes via supermesh construction. *Computer Methods in Applied Mechanics and Engineering*, 198(33-36):2632–2642, 2009.

[22] P. E. Farrell and J. R. Maddison. Conservative interpolation between volume meshes by local Galerkin projection. *Computer Methods in Applied Mechanics and Engineering*, 200(1-4):89–100, 2011.

[23] C. S. Peskin. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, 10(2):252 – 271, 1972.

[24] C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[25] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.

[26] Axelle Viré. How to float a wind turbine. *Reviews in Environmental Science and Bio/Technology*, 11(3):223–226, 2012.

[27] A. Viré, J. Xiang, F. Milthaler, P. E. Farrell, M. D. Piggott, J.P. Latham, D. Pavlidis, and C. C. Pain. Modelling of fluid-solid interactions using an adaptive mesh fluid model coupled with a combined finite-discrete element model. *Ocean Dynamics*, 62(10-12):1487–1501, 2012.

[28] J. Xiang, J.-P. Latham, A. Viré, E. Anastasaki, and C. C. Pain. The immersed body method and simulation tools for numerical breakwater modelling. In *International Symposium Coastal and Offshore Geotechnics*, 2012.

[29] P. E. Farrell. The addition of fields on different meshes. *Journal of Computational Physics*, 230(9):3265–3269, 2011.

[30] S. Jewer, A. G. Buchan, C. C. Pain, and D. G. Cacuci. An immersed body method for coupled neutron transport and thermal hydraulic simulations of PWR assemblies. *Annals of Nuclear Energy*, 68(0):124 – 135, 2014.

[31] A. G. Buchan, S. Jewer, P. E. Farrell, C. C. Pain, D. G. Cacuci, and M. Moatamedi. Simulating coupled neutron-fluid-heat transfer for the study of unintended operating scenarios in pressure water reactors using an immersed body modelling technique. In *NAFEMS*, 2014.

[32] M. D. Piggott, P. E. Farrell, C. R. Wilson, G. J. Gorman, and C. C. Pain. Anisotropic mesh adaptivity for multi-scale ocean modelling. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1907):4591–4611, 2009.

[33] M. D. Piggott, G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin, and M. R. Wells. A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes. *Int. J. Numer. Methods Fluids*, 56:1003–1015, 2008.

[34] C. C. Pain, A. P. Umpleby, C. R. E. de Oliveira, and A. J. H. Goddard. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Computer Methods in Applied Mechanics and Engineeringng*, 190(29-30):3771–3796, April 2001.

[35] M A. Goffin, A. G. Buchan, A. C. Belme, C. C. Pain, M. D. Eaton, P. N. Smith, and R. P. Smedley-Stevenson. Goal-based angular adaptivity applied to the spherical harmonics discretisation of the neutral particle transport equation. *Annals of Nuclear Energy*, 71:60–80, September 2014.

[36] A. G. Buchan, A. S. Candy, S. R. Merton, C. C. Pain, J. L. Hadi, M. D. Eaton, A. J. H. Goddard, R. P. Smedley-Stevenson, and G. J. Pearce. The inner-element subgrid scale finite element method for the Boltzmann transport equation. *Nuclear Science and Engineering*, 164:105–122, 2010.

[37] J Southern, GJ Gorman, MD Piggott, PE Farrell, MO Bernabeu, and J Pitt-Francis. Anisotropic mesh adaptivity for cardiac electrophysiology. pages 929–938. ELSEVIER SCIENCE BV, 2010.

[38] P. Smith, J. Lillington, and C. Middlemas. Radiation transport modelling and the ANSWERS codes suite. *Nuclear Future*, 7, 2011.

[39] CUBIT 15.4 User Documentation. Technical Manual SAND2019-3478 W, Sandia National Laboratories.

[40] A. G. Buchan, C. C. Pain, A. J. H. Goddard, M. D. Eaton, J. L. M. A. Gomes, G. J. Gorman, C. M. Cooling, B. S. Tollit, E. T. Nygaard, D. E. Glenn, and P. L. Angelo. Simulated transient dynamics and heat transfer characteristics of the water boiler nuclear reactor - SUPO - with cooling coil heat extraction. *Annals of Nuclear Energy*, 48:68 – 83, 2012.