Annals of Nuclear Energy 223 (2025) 111601

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene

NURBS-enhanced finite element spatial discretisation methods for the steady-state multigroup neutron diffusion equation

J. Trainor [a,*], M.D. Eaton [a], J. Kópházi [b,a], S.G. Wilson [c], C. Latimer [c], L. Smith [c], D. Baker [c], I. Jordan [c]

[a] Imperial College London, Nuclear Engineering Group, Department of Mechanical Engineering, City and Guilds Building, South Kensington Campus, London, SW7 2BX, United Kingdom

[b] Budapest University of Technology, Institute of Nuclear Techniques, Budapest, Hungary

[c] Rolls-Royce PLC, PO BOX 2000, Derby DE21 7XX, United Kingdom

ARTICLE INFO

Dataset link: https://zenodo.org/records/15607929

Keywords:
Non-Uniform Rational B-splines (NURBS)
Multigroup neutron diffusion equation
Lagrangian finite element method (FEM)
Nuclear reactor physics
Method of manufactured solutions (MMS)

ABSTRACT

The NURBS-enhanced finite element method (NEFEM) is a recent innovation in spatial discretisation methods. The NEFEM combines the conventional FEM with computer-aided geometric design (CAGD) boundary representation (B-rep) approaches based upon Non-Uniform Rational B-spline (NURBS) geometrical representations of the computational domain. The aim of the NEFEM is to streamline the CAGD to computer-aided engineering (CAE) modelling and simulation (M&S) pipeline and provide improved geometrical representations of the underlying curvilinear geometry in nuclear reactor physics and reactor shielding simulations. This eliminates the requirement for local modifications to the underlying computational mesh to preserve the surface areas and volumes of curvilinear geometrical features within the computational domain. Such local mesh modifications are required, within conventional isoparametric Lagrangian FEM approaches, to preserve fissile mass and neutron leakage within curvilinear geometrical and computational domains.

This paper presents the application of the NEFEM to the multigroup neutron diffusion equation (NDE) for three nuclear reactor physics benchmark verification test cases. A further method of manufactured solution (MMS) benchmark verification test case is used to establish the order of convergence of the NEFEM compared to the FEM for both linear and quadratic elements. In addition, an analytical Wigner–Seitz pincell problem is used to further investigate the accuracy of the NEFEM. The results from these benchmark verification test cases demonstrate that the NEFEM yields improved numerical accuracy compared to the conventional FEM. This improved numerical accuracy is primarily achieved through the improved geometrical representation of curvilinear geometries. While the NURBS-enhancement of elements necessitates a small increase to the pre-processing time associated with the method, the increased accuracy of the NEFEM allows it to achieve competitive computational solution times compared to the standard Lagrangian FEM.

1. Introduction

The development of modern, geometrically complex, nuclear reactor designs, such as the transformational challenge reactor (TCR) (Betzler et al., 2020), increasingly requires the use of the latest state-of-the-art high-fidelity multidimensional nuclear reactor core design methods (Gaston et al., 2015; Schmidt et al., 2015; Hales et al., 2015; Martineau et al., 2020; Prince et al., 2024). The advent of additive manufacture (Nelson, 2023), digital twin nuclear reactor design (Lu et al., 2023; Mengyan et al., 2024) and artificial intelligence/machine learning (AI/ML) engineering design optimisation algorithms (Betzler et al., 2020; Sobes et al., 2021), embodied in the nuclear industry 4.0, will likely drive the development of improved high-fidelity nuclear reactor core design methods that streamline the computer-aided geometric design (CAGD) boundary representation (B-rep) and computer-aided engineering (CAE) pipeline (Hughes et al., 2005). Current whole core nuclear reactor design methods utilise a two-step modelling and simulation (M&S) process (Smith, 2017). First two-dimensional (2D) nuclear reactor lattice physics computations are performed for nuclear fuel pins and fuel assemblies with prescribed periodic boundary conditions (Smith, 2017). The 2D nuclear reactor lattice physics computations utilise collision probability (CP) methods in the reference library group structure for a single, or group, of nuclear fuel pins (e.g., a couple of hundred neutron energy groups) (Cacuci, 2010). This is followed by single nuclear fuel assembly computation performed using the method

* Corresponding author.
E-mail address: j.trainor20@imperial.ac.uk (J. Trainor).

https://doi.org/10.1016/j.anucene.2025.111601
Received 5 February 2025; Received in revised form 1 May 2025; Accepted 20 May 2025
Available online 16 June 2025

Nomenclature

Acronyms

B-rep	Boundary Representation
CAE	Computer-Aided Engineering
CAGD	Computer-Aided Geometric Design
CBG	Continuous Bubnov–Galerkin
CDF	Cell Disadvantage Factor
DoF	Degrees of Freedom
FEM	Finite Element Method
IGA	Isogeometric Analysis
M&S	Modelling & Simulation
MMS	Method of Manufactured Solutions
NDE	Neutron Diffusion Equation
NEFEM	NURBS-enhanced Finite Element Method
NURBS	Non-Uniform Rational B-Splines
pcm	per cent mille

Geometric Parameters

(ξ, η, ζ)	FEM parameter space
(ξ^*, η^*, ζ^*)	NURBS-enhanced parameter space
(x, y, z)	Physical space
$\boldsymbol{J}$	Jacobian matrix
Γ	Domain boundary
Γ_R	Reflective boundary
Γ_V	Vacuum boundary
$\mathcal{F}$	Finite Element parametric mapping
$\mathcal{N}$	NURBS parametric mapping
$\mathcal{T}$	Computational mesh of domain V
∂T_e	FEM boundary element e
∂T_e^*	NURBS-enhanced boundary element e
ε	Element edge
E_V	Number of elements in domain V
N_i^*	NURBS-enhanced basis function
n_Γ	Nodes in domain boundary Γ
n_e	Nodes in element T_e
N_i	FEM basis function
n_V	Nodes in domain V
T_e	FEM volumetric element e
T_e^*	NURBS-enhanced volumetric element e
V	Domain

Physical Constants

$\bar{\nu}$	Average prompt neutrons per fission [neutrons fission^{-1}]
$\bar{\nu}_{i,d}$	Average delayed neutrons per fission from ith group [neutrons fission^{-1}]
$\bar{\nu}_t$	Average total neutrons per fission [neutrons fission^{-1}]
β	Delayed neutron fraction [–]
χ	Prompt neutron energy spectrum [–]
Σ_f	Macroscopic neutron fission cross-section [cm^{-1}]
Σ_r	Macroscopic neutron removal cross-section [cm^{-1}]
Σ_s	Macroscopic neutron scattering cross-section [cm^{-1}]
Σ_t	Macroscopic total neutron cross-section [cm^{-1}]
D	Neutron diffusion coefficient [cm]

Solution Variables

$\bar{u}$	Average of a quantity u
$\boldsymbol{f}^e$	Force (load) vector for element e
$\boldsymbol{K}^e$	Stiffness matrix for element e
$\boldsymbol{M}^e$	Mass matrix for element e
$\boldsymbol{w}^T$	Vector of transpose groupwise weight functions $\boldsymbol{w}^T = (w_1, \ldots, w_G)$
$\boldsymbol{x}$	Position vector $\boldsymbol{x} = (x, y, z)$
$\hat{\mathcal{H}}_{h,p}(.,.)$	Bilinear scattering term
$\hat{F}_{h,p}(.,.)$	Bilinear fission term
$\hat{L}_{h,p}(.,.)$	Bilinear leakage and removal term
$\hat{Q}_{h,p}(.)$	Linear source term
$\mathcal{H}_{g,g'}$	Group g to g' scattering operator
Q	Neutron source [neutrons cm^{-3} s^{-1}]
$\mathcal{S}$	Collection of trial solutions
$\mathcal{V}$	Collection of weight functions
$\boldsymbol{\Phi}$	Set of groupwise scalarflux values $\boldsymbol{\Phi} = (\phi^1, \ldots, \phi^G)$
ϕ^g	Neutron Scalar flux of energy group g [cm^{-2} s^{-1}]
E	Energy [MeV]
$F_{g,g'}$	Group g to g' fission operator
$H^1(V)$	Sobolev space
$L^2(V)$	Lebesgue space
L_g	Group g destruction (leakage and removal) operator
Q_g	Group g source
$u_{h,p}$	Finite dimensional approximation of function u with principal size h and order p
w	Weight function $w \in \mathcal{V}$
$\mathrm{k_{eff}}$	Effective neutron multiplication factor [–]

of characteristics (MoC) (Brough and Chudley, 1980) for a smaller number of energy groups (e.g., 10–20 neutron energy groups) (Cacuci, 2010). The aim of these nuclear reactor lattice physics computations is to produce resonance self-shielded, group condensed (i.e., two to four neutron energy groups) (Boyd et al., 2018), homogenised macroscopic neutron cross-sections and assembly discontinuity factors (ADFs) for whole core, coarse Cartesian mesh, nodal methods (Smith, 2017, 1986; Sanchez, 2009; Hébert and Benoist, 1991). The whole core, coarse Cartesian mesh, nodal methods are usually combined with the nuclear reactor lattice physics computations using flux reconstruction algorithms to compute the fine-scale nuclear fuel reaction rates and pin powers (Smith, 1986; Sanchez, 2009; Hébert and Benoist, 1991).

The nuclear reactor lattice physics computations solve the first–order from of the neutron transport equation (NTE). The first–order form of the NTE was first derived by Wick (1936). It was independently derived by Ornstein and Uhlenbeck in 1937 (Gould and Sharapov, 2024). Subsequently, Halpern et al. (1938) used modern particle balance considerations to derive the first-order NTE. Amaldi (1984) provides a comprehensive historical review of the development of nuclear reactor physics which also discusses neutron transport theory. The NTE is a hyperbolic partial–integral–differential equation (PIDE) that models the migration, interaction, and leakage of neutrons within a host medium (Cacuci, 2010). The NTE is a linearised form of the nonlinear Boltzmann transport equation from the kinetic theory of rarefied gases (Boltzmann, 1872) and assumes neutron–neutron interactions are negligible (Cacuci, 2010). In general, it is a function of a seven-dimensional phase space of position (x, y, z), angle (θ, φ), energy (E) and

time. The whole core, coarse Cartesian mesh, nodal methods solve the neutron diffusion equation (NDE) which is a continuum approximation of the NTE (Lawrence, 1986). The NDE is an elliptic PIDE and was first derived by Fermi (1936) when he was analysing the absorption and diffusion of neutrons within host media. It assumes linear anisotropy of the angular distribution of neutrons within the host medium (Cacuci, 2010). The NDE is a reasonably accurate approximation to the NTE for neutron migration in weakly absorbing media several mean free paths away from isolated neutron sources, boundaries of the computational domain or highly absorbing materials (Cacuci, 2010). As such, industrial uses of NDE solvers include whole-core calculations and methods such as Diffusion Synthetic Acceleration (DSA), where a diffusion solve is used to accelerate the convergence of a transport code (Turcksin and Ragusa, 2014).

A wide variety of deterministic discretisation methods have been developed to solve both the NTE and the NDE. Multigroup methods, first developed in their modern form by Ehrlich and Hurwitz Jr in 1954, are used to discretise the energy domain (Feynman et al., 1947; Erlich et al., 1947; Hurwitz and Ehrlich, 1948, 1953). A historical review of the development of multigroup methods for discretisation the energy domain has been provided by Wachspress (1992).In contrast, there are many different spatial discretisation methods that have been developed for the NDE and these include coarse mesh finite difference (CMFD) (Wang and Zhu, 2021), coarse Cartesian mesh nodal (CMN) methods (Lawrence, 1986), finite volume (FV) (Ge et al., 2015), finite element (FE) (Semenza et al., 1972; Kang and Hansen, 1973; Hansen and Kang, 1975), spectral element (SE) (Mund, 2011; Barbarino et al., 2013, 2015), isogeometric analysis (IGA) (Hall et al., 2012; Owens et al., 2016; Welch et al., 2017a,b; Owens et al., 2017a,b; Latimer et al., 2020a,b, 2021; Wilson et al., 2024a,b,c) and virtual element (VE) methods (Ferguson et al., 2021, 2022). Note that while CMFD is typically used as an acceleration scheme for the nodal method and the first order NTE, in the fine mesh limit CMFD is equivalent to the standard centred finite difference method (FDM) discretisation of the NDE. Within the nuclear industry, the most widely used spatial discretisation methods for solving the NDE are the CMN and CMFD methods (Lawrence, 1986; Wang and Zhu, 2021). These are computationally efficient methods for performing whole-core nuclear reactor core design simulations. However, these methods cannot model curvilinear geometrical and computational domains accurately (Lawrence, 1986; Wang and Zhu, 2021). The advent of modern multicore Central Processing Unit (CPU) and manycore Graphics Processing Units (GPU), embodied in the latest exascale leadership class (Tier-0) high-performance computing (HPC) hardware architectures (Leslie, 2023), is driving the development of high-order spatial discretisation methods through the centre for efficient exascale discretisations (CEED) (Abdelfattah et al., 2021).

The CEED programme of research has indicated that high-order spatial discretisation methods are more computationally efficient than low-order methods on the latest hybrid multicore CPU and manycore GPU HPC hardware architecture (Abdelfattah et al., 2021). For such HPC hardware architectures, memory bandwidth is becoming critically important in determining the optimal CPU/GPU performance (Moxey et al., 2016). As the number of CPU/GPU cores increases, and with increasingly more complex memory hierarchies, the data must be managed more efficiently by using algorithms that utilise the cache in a more optimal manner (Löhner, 2010). Therefore, methods and algorithms that minimise indirection and cache misses are becoming more important for the latest HPC hardware architectures (Löhner and Galle, 2002). Such methods will lead to improved computational efficiency and reduced execution times for large-scale problems (Xu, 2021). High-order spatial discretisation methods are ideally suited to these new HPC hardware architectures as they utilise dense matrix operations that are typically more memory compact than sparse matrix operations (Tomov et al., 2010). In tandem with these innovations in HPC hardware architectures, the streamlining of the CAGD B-rep to CAE pipeline is also driving the development of high-order curvilinear FE, SE, IGA, spatial discretisation methods which can model multidimensional curvilinear geometries and computational domains more accurately (Hall et al., 2012; Owens et al., 2016; Welch et al., 2017b,a; Owens et al., 2017b,a; Latimer et al., 2020a,b, 2021; Wilson et al., 2024a,b,c).

The last two decades have seen the emergence of IGA spatial discretisation methods. These methods aim to streamline the CAGD B-rep to CAE pipeline and have been driven by the complexity of modern engineering design and the need to model the geometries of engineering systems with greater numerical accuracy (Hughes et al., 2005). A further aim has been to reduce the time required to perform CAGD repair of geometries and subsequent meshing of the computational domain (Marchandise et al., 2012). There are a wide variety of different IGA methodologies that have been developed over the past two decades such as Non-Uniform Rational B-splines (NURBS) (Hughes et al., 2005), locally refinable B-splines (LR-B-splines) (Dokken et al., 2013), T-splines (Bazilevs et al., 2010), S-splines (Li and Sederberg, 2019), and U-splines (Thomas et al., 2022). Such methods are capable of exactly modelling conic sections in two-dimensions (2D) and quadric surfaces in three-dimensions (3D) and any geometries that can be produced by commercial CAGD B-rep software (Hughes et al., 2005). One of the primary challenges of IGA technologies is that the volumetric splines produced by these methods can only mainly model volumes that are isomorphic to quadrilaterals in 2D and hexahedra in 3D (Chen et al., 2022). However, recent advances in multidimensional NURBS enhanced mesh generation algorithms are addressing this challenge (Ferguson et al., 2022; Zou et al., 2024a; Otoguro et al., 2018; Sevilla et al., 2011b; Montanari et al., 2024; Kirilov et al., 2024). In the field of nuclear reactor physics, the methods were first applied by Hall, Eaton, and Williams (Hall et al., 2012). These methods have been subsequently developed for both the first and second-order forms of the NTE (Owens et al., 2016; Welch et al., 2017a; Owens et al., 2017b,a; Latimer et al., 2020a,b, 2021).

In the last decade, the definition of IGA has been expanded to include more geometrically flexible NURBS-enhanced finite element spatial discretisation methods (Sevilla et al., 2008; Sevilla and Fernández-Méndez, 2011; Sevilla et al., 2011a). These methods encompass a greater range of curvilinear polytopes and remove the restrictions imposed using conventional IGA spatial discretisation methods (Sevilla et al., 2008; Sevilla and Fernández-Méndez, 2011; Sevilla et al., 2011a). These methods can utilise either super-parametric or isoparametric formulations where the geometry field is represented by NURBS basis (or shape) functions, and the solution field variables are represented by Lagrangian finite element basis functions. In tandem with these NURBS-enhanced spatial discretisation methods, IGA technology has been expanded to include novel U-splines which also attempt to expand the range of curvilinear polytopes that can be used (Thomas et al., 2022). More recently, virtual element (VE) spatial discretisation methods have been developed that are also NURBS enhanced and can use arbitrary convex or concave/reentrant polygons and polyhedra (Ferguson et al., 2021). These NURBS-enhanced VE methods have been applied to both the neutron diffusion (Ferguson et al., 2021) and the first-order form of the neutron transport equation (Ferguson, 2022). These NURBS-enhanced VE methods represent the current state-of-the-art in the field due to their numerical robustness, their ability to model convex and re-entrant curvilinear polygonal and polyhedral elements and their ability to model arbitrarily complex CAGD curvilinear geometries.

This paper focuses on the further development and application of the NURBS-enhanced finite element (NEFEM) to the field of nuclear reactor physics. More specifically, this paper focuses on the numerical solution of the two-dimensional (2D) multigroup NDE. The aim is to elucidate the fundamental mathematical formulation of the NEFEM when applied to the multigroup NDE. A further aim is to determine the numerical and geometrical accuracy of the NEFEM compared to conventional Lagrangian FEM. Both superparametric NEFEM and isoparametric NEFEM will be analysed and assessed within this study.

Superparametric NEFEM utilises high-order NURBS basis functions for the geometry field and linear Lagrangian FE basis functions for the solution field variables. The isoparametric NEFEM utilises NURBS for the geometry and Lagrangian basis functions for the solution field variables with the same order of polynomial representation. This paper is organised as follows: in Section 2 the mathematics and geometry representation capability of NURBS basis functions is introduced. In Section 3 the mathematics underpinning the NEFEM is developed. In Section 4 the multigroup NDE is then discretised using the NEFEM. In Section 5 the NEFEM and conventional FEM are applied to a series of nuclear reactor physics benchmark verification test cases. Finally, in Section 6 we provide concluding remarks and suggestions for future research work.

2. Non-Uniform Rational B-Splines (NURBS)

This section provides a concise description of the theory associated with NURBS curves and surfaces. A more comprehensive analysis of the history and theory associated with these methods can be found in the original paper by Hughes et al. (2005).

2.1. NURBS curves

NURBS curves are the one-dimensional form of a NURBS patch, defined through a piecewise rational function that is described parametrically as:

$$\mathbf{x}(\xi^*) = \left(\sum_{i=1}^{n} w_i C_i N_{i,p}(\xi^*)\right) \Bigg/ \left(\sum_{i=1}^{n} w_i N_{i,p}(\xi^*)\right), \tag{1}$$

where $\mathbf{x}(\xi^*)$ is the position in physical space associated with parametric co-ordinate ξ^*, C_i are the n control points with corresponding weights w_i and Bezier spline basis functions $N_{i,p}(\xi^*)$. These basis functions are defined recursively, where for $p = 0$ the B-spline basis functions are:

$$N_{i,0}\left(\xi^*\right) = \begin{cases} 1, & \text{if } \xi_i^* \le \xi^* < \xi_{i+1}^*, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

while for $q = 1, 2, 3, \ldots, p$ they are given by:

$$N_{i,q}\left(\xi^*\right) = \frac{\xi^* - \xi_i^*}{\xi_{i+q}^* - \xi^* i} N_{i,q-1}\left(\xi^*\right) + \frac{\xi_{i+q+1}^* - \xi^*}{\xi_{i+q+1}^* - \xi^* i + 1} N_{i+1,q-1}\left(\xi^*\right), \tag{3}$$

where q is the order of the basis function and ξ_i^* is the i_{th} 'knot' in the 'knot vector' of length $k + p + 1$. The knot vector is a non-decreasing set of coordinates which describes the boundaries and degree of the B-spline basis functions, allowing it to control the local representation of the curve. A general open knot vector can be seen in Eq. (4).

$$\Xi = \left\{0, \ldots, 0, \xi_{p+1}^*, \ldots, \xi_{n-p-1}^*, 1, \ldots, 1\right\}. \tag{4}$$

Eqs. (2) and (3) are known as the 'Cox–de Boor recursion relation' and are used to describe the basis functions for Bezier splines. Similarly to FEM basis functions, a partition of unity holds over the span of the element:

$$\sum_{i=1}^{n} N_{i,p}\left(\xi^*\right) = 1. \tag{5}$$

In addition, B-spline basis functions are always non-negative over the entire domain:

$$N_{i,p}\left(\xi^*\right) \ge 0, \forall \xi^*. \tag{6}$$

Similarly to the FEM, these basis functions can then be evaluated to map a position in parametric space to physical space:

$$\mathbf{x}(\xi^*) = \sum_{i=1}^{n} C_i N_i(\xi^*). \tag{7}$$

Eq. (8) shows an example open knot vector for a one-dimensional quadratic NURBS patch or element. The basis functions associated with such a knot vector can be seen in Fig. 1, where weights of $w_i = 1.0$ have been used for each of the seven control points. The first and last knots are repeated with multiplicity of p+1, such as in Eq. (8), the knot vector is described as 'open' and the NURBS will be interpolatory at element boundaries.

$$\Xi = \{0, 0, 0, 0.2, 0.5, 0.5, 0.8, 1, 1, 1\}. \tag{8}$$

The basis functions presented in Fig. 1 can be used to produce the complex curvilinear NURBS curve presented in Fig. 2(a). The thick blue line shows the NURBS curve while the points $C_1, \ldots, C_7$ and connecting hashed black line shows the control points and control polygon respectively.

The non-uniform nature of NURBS introduces the weighting function $W(\xi^*)$, allowing the user to apply a value ω_i to any relevant control point, modifying the relative 'weight' of the point within the control polygon on the final physical representation of the geometry. This additional feature allows NURBS to exactly model complex geometries, such as those that involve conics, curves or ellipses (Nguyen et al., 2015). Another feature of NURBS utilised by CAGD software is that of 'trimmed' NURBS, where the initial parametrisation of the polygon is limited to a portion of the parametric space (Kim et al., 2009). An example of this can be seen in Fig. 2(b), where the curve from Fig. 2(a) has been restricted to $[0.1, 0.9]$ in the parameter space.

2.2. NURBS surfaces

The theory of NURBS curves can similarly be extended to higher dimensional representations. For example, NURBS surfaces are represented in parametric form by:

$$\mathbf{x}\left(\xi^*, \eta^*\right) = \left(\sum_{i=1,j=1}^{n} w_{ij} C_{ij} N_{ij}^{p,q}\left(\xi^*, \eta^*\right)\right) \Bigg/ \left(\sum_{i=1,j=1}^{n} w_{ij} N_{ij}^{p,q}\left(\xi^* \eta^*\right)\right), \tag{9}$$

where $0 \le \xi^* \le 1$, $0 \le \eta^* \le 1$, C_{ij} are the k control points with corresponding weights w_{ij} and two-dimensional Bezier spline basis functions $N_{ij}^{p,q}(\xi^*, \eta^*)$. Each of these two-dimensional Bezier basis functions is defined as the tensor product of the one-dimensional basis functions described previously:

$$\mathbf{N}_{i,j}^{p,q}\left(\xi^*, \eta^*\right) := \mathbf{N}_i^p\left(\xi^*\right) \mathbf{N}_j^q\left(\eta^*\right). \tag{10}$$

An example of a two-dimensional B-spline basis function is presented in Fig. 3(a), where a single two-dimensional (2D) B-spline basis function is shown to be the product of the two one-dimensional (1D) B-spline basis functions highlighted in red in the ξ^* and η^* parametric dimensions.

To further display the ability of higher order NURBS patches, or elements, to accurately model complex shapes, an example NURBS surface and associated control polygon are presented in Fig. 3(b). While such a shape can be modelled using a single NURBS patch, conventional geometric modelling techniques, such as the FEM, would require many elements to model the shape accurately. Alternatively, a high-order curvilinear finite element could be used to model such a shape. However, this might not be able to model all shapes exactly such as conic sections in 2D and quadric surfaces in 3D whereas a single NURBS patch, or volume, can model such shapes. In addition, for the FEM, significant increases in the order of the finite elements, or the spatial refinement of the computational mesh, would impact upon the overall computational solution time.

3. NURBS-Enhanced Finite Element Method (NEFEM)

This section covers the NURBS-enhanced finite element method (NEFEM) as it is applied to solve the NDE. For the sake of simplicity, the method will be shown for an element with one side on a NURBS boundary. However, the method can incorporate NURBS boundaries on any number of sides of the element without necessitating any

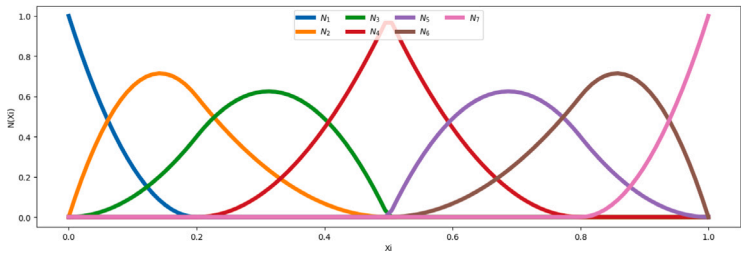
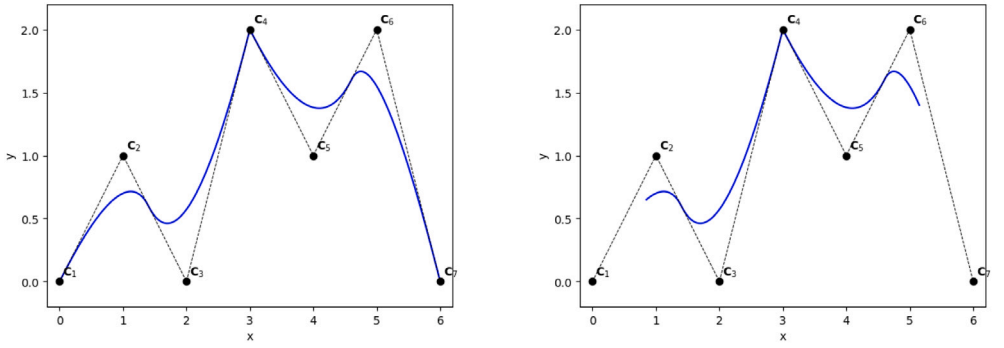


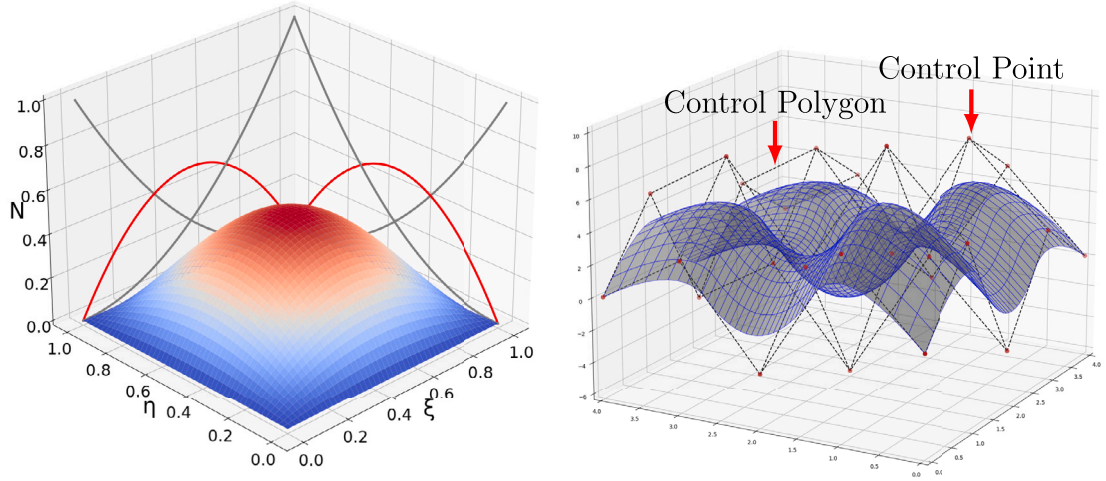
Fig. 1. Quadratic (p = 2) NURBS Basis functions associated with the open knot vector $\{0, 0, 0, 0.2, 0.5, 0.5, 0.8, 1, 1, 1\}$.



(a) Full NURBS Curve.

(b) NURBS Curve trimmed to $\xi^* = [0.1, 0.9]$.

Fig. 2. Control Polygon associated with the knot vector $\{0, 0, 0, 0.2, 0.5, 0.5, 0.8, 1, 1, 1\}$.



(a) Two-dimensional IGA basis functions as a tensor product of one-dimensional IGA basis functions.

(b) Example NURBS surface with associated control polygon.

Fig. 3. Two-dimensional IGA Basis functions and example NURBS surface.

changes in implementation. Additionally, this method may be extended to three dimensions (3D), without loss of generality. However a two-dimensional (2D) implementation will be presented here for the sake of brevity. A physical domain $V \subset \mathbb{R}^2$ has a boundary Γ, where all or part of the boundary is curvilinear and can be defined by NURBS. We assume that any NURBS in the domain can be parametrised by:

$$\mathbf{C}([0, 1]) \subseteq \Gamma \subset \mathbb{R}^2. \tag{11}$$

The domain V is meshed into a discrete number of elements T, some of which will have a side on a boundary Γ, which may be curvilinear. It is assumed that any curvilinear edges of these 'boundary' elements can be described by a single NURBS curve. This does not constrain the boundary description of elements which use a NURBS definition, as the breakpoints which separate the piecewise curve description do not need to coincide with the nodes. Where elements lie in the 'interior' of the domain, and hence do not have a side on a curvilinear boundary, they are treated as standard Lagrangian finite elements. Therefore, each computational mesh $\mathcal{T}$ may be described as:

$$\mathcal{T} = T \cup T^*, \tag{12}$$

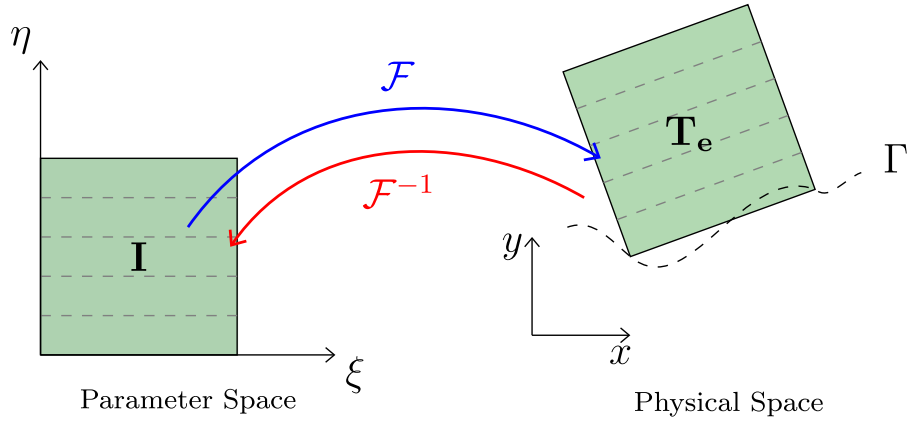


Fig. 4. Transform $\mathcal{F}$ maps the reference element I to the straight sided element T_e in physical space.

where T is the set of standard finite elements, $T = \{T_e\}$ and T^* is the set of NURBS-enhanced elements, $T^* = \{T_e^*\}$. A sufficiently finely meshed domain will consist of a large number of these 'interior' elements, meaning that standard FEM quadrature/integration methods can be utilised for the majority of the system. This section will describe the modifications that are made in NEFEM to facilitate the NURBS boundary definitions for NURBS-enhanced elements T^* which require a curvilinear boundary description.

3.1. Parameter space transforms

A linear transform $\mathcal{F}$ is defined, which maps the $[0,1]^2$ reference element, I, from FEM parametric space (ξ,η) to physical space (x,y), presented in Fig. 4. When applied to a unit square element I, this produces the element T_e, such that $\mathcal{F} : I \rightarrow T_e$, following Eq. (13).

$$\boldsymbol{x}(\xi,\eta) = \sum_{i=1}^{n_e} C_i N_i(\xi,\eta) = C_1 N_1(\xi,\eta) + \cdots + C_{n_e} N_{n_e}(\xi,\eta), \tag{13}$$

where n_e is the number of nodes used to describe the finite element, C_i are the control points associated with the FEM mesh and N_i are the FEM basis functions. Similarly, a transform $\mathcal{N}$ is defined, which maps the $[0,1]^2$ reference element, I, from NURBS parametric space (ξ^*,η^*) to physical space (x,y), presented in Fig. 5. When applied to the reference element I, this produces the element T_e^*, such that $\mathcal{N} : I \rightarrow T_e^*$, following Eq. (14).

$$\boldsymbol{x}\left(\xi^*,\eta^*\right) = \sum_{i=1}^{n_e^*} C_i^* N_i^*\left(\xi^*,\eta^*\right) = C_1^* N_1^*(\xi^*,\eta^*) + \cdots + C_{n_e^*}^* N_{n_e^*}^*(\xi^*,\eta^*), \tag{14}$$

where n_e^* is the number of nodes used to describe the NURBS-enhanced element, C_i^* are the control points associated with the NURBS mesh and N_i^* are the NURBS basis functions. Note that the values of n_e and n_e^* may differ, as discussed in Section 3.2. It can be seen from Figs. 4 and 5 that the linear FEM approximation of the portion of the boundary Γ produced by the element T_e is considerably less accurate than its NEFEM counterpart, T_e^*. If these were interior elements, or if the relevant portion of the boundary Γ was linear in shape, then a NURBS description would match that of FE, so the geometry of T_e and T_e^* would be identical.

An example NEFEM mesh is presented in Fig. 6, where a domain V with curvilinear internal boundary Γ_I has been meshed for NEFEM. Lighter elements in the figure utilise a standard finite element description, while the shaded elements utilise a NURBS-enhanced representation, allowing them to model the curvilinear boundary Γ_I exactly. While geometric mappings may use Finite Element or NURBS basis functions for interpolation of spatial variables depending on whether standard FEM or NURBS-enhanced elements are used, interpolation of the field variable ϕ is always restricted to FEM basis functions, as shown in Eq. (15). These basis functions are the standard Lagrangian FEM basis functions, defined over the nodes used to describe the underlying FEM element T_e.

$$\phi(\xi,\eta) = \sum_{i=1}^{n_e} \phi_i N_i(\xi,\eta) = \phi_1 N_1(\xi,\eta) + \cdots + \phi_{n_e} N_{n_e}(\xi,\eta). \tag{15}$$

3.2. NURBS-enhanced finite element generation

Each of the NURBS-enhanced elements contains two sets of nodal information. This nodal information is the FEM nodes that define the field variable and the NURBS nodes that define the underlying geometry. These descriptions are not limited to the same order and as such elements may be subparametric, isoparametric or superparametric. Differences between these three types of parametric elements are presented in Fig. 7. Isoparametric elements are the most common element types, where the nodes defining the geometry and field variable are of the same order. Subparametric elements utilise geometric descriptions that are lower order than that of the field variable and vice versa for superparametric elements. In the case of NEFEM, the NURBS description will generally be of at least second ($p = 2$) order, as this is the lowest order that provides improved geometric accuracy over its standard Lagrangian FEM counterpart.

The geometric NURBS description of the element is created as the computational mesh is generated. This is achieved by detecting any elements that coincide with the curvilinear boundary Γ of the computational domain. A two-dimensional (2D) NURBS-enhanced element is then generated that matches the shape of the underlying finite element. Any sides of the element that lie on a curvilinear boundary are then locally modified to exactly model the CAGD boundary representation (B-rep). This process is illustrated in Fig. 8 for a linear finite element T_e with a simple quadratic $\Xi = \{0,0,0,1,1,1\}$ NURBS correction. The first element in Fig. 8 shows the isoparametric finite element T_e. This element is then converted into a superparametric NEFEM element, as it now has a quadratic NURBS geometry description with nine spatial nodes and four field variable nodes, corresponding to the knot vector $\Xi = \{0,0,0,1,1,1\}$. This NURBS description still conforms to the original shape of the element T_e, so the control points are now repositioned to match the NURBS description of the section of the boundary Γ_e which the element is approximating, creating the final NEFEM element T_e^*.

3.3. Numerical quadrature within the NEFEM

For the sake of clarity, this section will describe the method by which one would integrate a function $\phi(x,y)$ over a NURBS-enhanced element T_e^*, presenting the difference in implementation between the

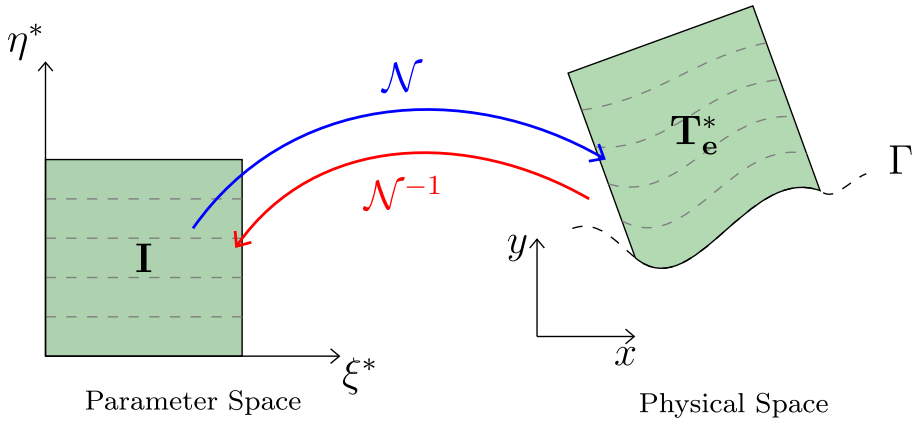


Fig. 5. Transform $\mathcal{N}$ maps the reference element I to the curved element T_e^* in physical space.

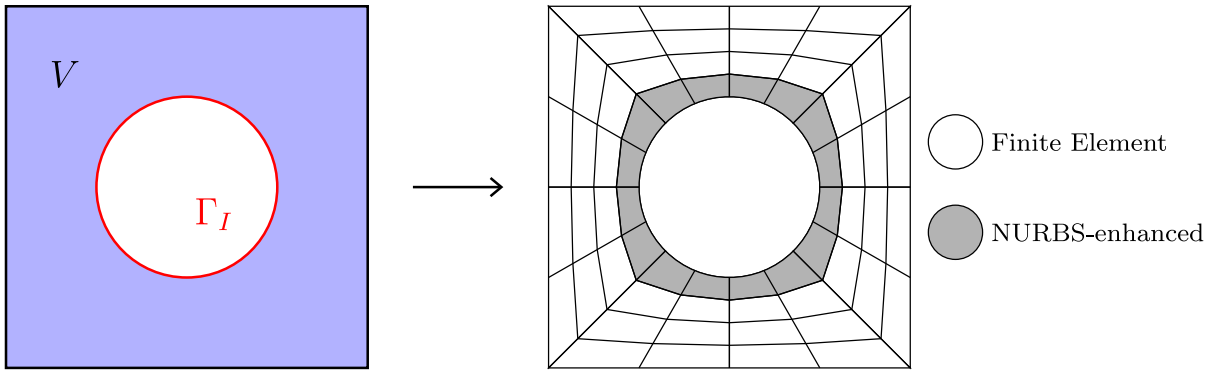


Fig. 6. Square domain V with an internal boundary Γ_I, which would be described by a NURBS curve in the CAGD representation. When spatially discretised into elements for the NEFEM, only those elements that lie on the curvilinear boundary need to be NURBS-Enhanced.

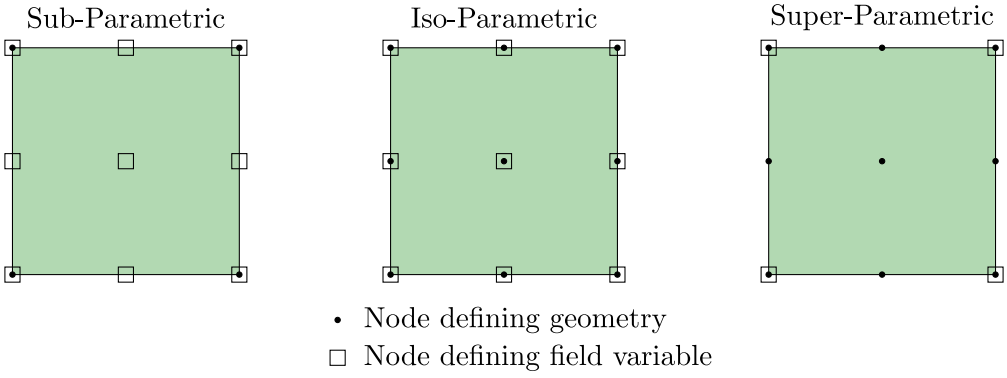


Fig. 7. Sub- Iso- and super-parametric elements which have geometric descriptions of lower, equal and higher order respectively. This separation of the geometry and field variable allows the NEFEM to exactly match the order of the CAGD description without requiring the field variable to be of the same order.

standard FEM and NEFEM for a two-dimensional problem. Such descriptions can then be trivially extended to any other geometric routines that could be relevant to a reactor physics code. The integral of a function over the linear element T_e is defined by:

$$\int_{T_e} \phi(x,y)dxdy = \int_I \left|\boldsymbol{J}_{\mathcal{F}}(\xi,\eta)\right| \phi(\xi,\eta)d\xi d\eta \approx \sum_{i=1,j=1}^{k} \left|\boldsymbol{J}_{\mathcal{F}}(\xi_i,\eta_j)\right| \phi(\xi_i,\eta_j)w_{i,j}, \tag{16}$$

where $\left|\boldsymbol{J}_{\mathcal{F}}\right|$ is the determinant of the Jacobian for the linear transformation $\mathcal{F}$ for the element. This integral can then be approximated with Gaussian quadrature, where k is the number of quadrature points, i, j are the individual quadrature points in each dimension and $w_{i,j}$ is the corresponding quadrature weight at the quadrature point. The Jacobian of the transform $\mathcal{F}$ is defined by:

$$\boldsymbol{J}_{\mathcal{F}} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}. \tag{17}$$

Multiplication by the Jacobian allows the derivatives of any function defined in (x, y) physical space to be evaluated in (ξ, η) parameter space as:

$$\boldsymbol{J}_{\mathcal{F}} \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial \xi} \\ \frac{\partial f}{\partial \eta} \end{pmatrix}. \tag{18}$$

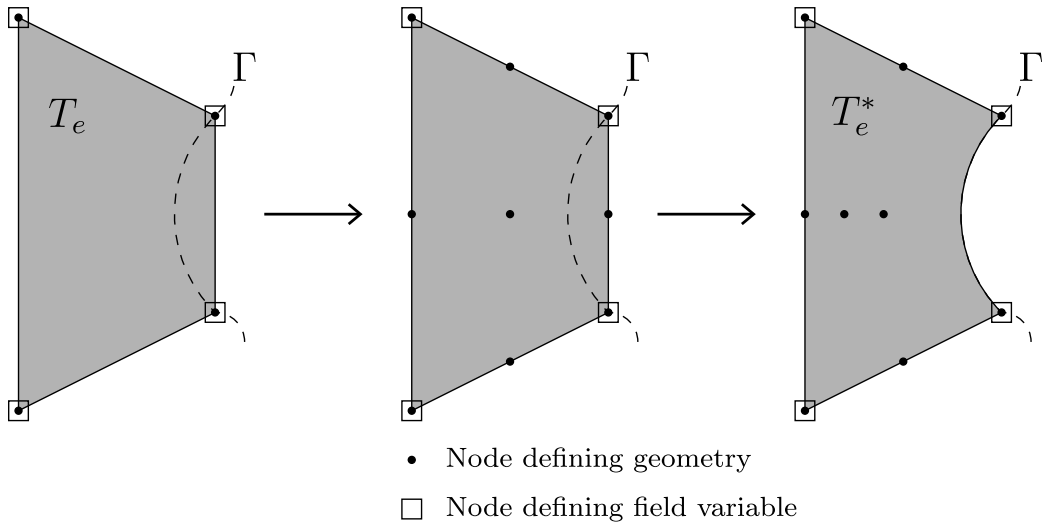


Fig. 8. NEFEM element generation routine. The mesh generator identifies finite elements T_e that sit on a curvilinear boundary Γ. The order of their geometric description is modified such that it matches that of the CAGD description. A NURBS-Enhanced description is then applied which matches the boundary, creating the NEFEM element T_e^*.

When an element is NURBS-enhanced, the integrations must be modified to instead account for the two different transforms $\mathcal{F}$ and $\mathcal{N}$ as well as the two respective parameter space descriptions (ξ, η) and (ξ^*, η^*). The value of a field variable ϕ at a parametric point (ξ, η) is evaluated from the summation of the basis functions over each of the nodes associated with the FE description, n_e, as presented in Eq. (19).

$$\phi(\xi, \eta) = \sum_{i=1}^{n_e} \phi_i N_i(\xi, \eta). \tag{19}$$

Conversely, the position in physical space is evaluated using the NURBS description and associated basis functions, though a summation over each of the nodes associated with the NE description, n_e^*:

$$\boldsymbol{x}(\xi^*, \eta^*) = \sum_{i=1}^{n_e^*} \boldsymbol{x}_i N_i^*(\xi^*, \eta^*), \tag{20}$$

where $\boldsymbol{x}$ is the position in physical space. The volume integral of a NEFEM element must be evaluated using the (ξ^*, η^*) space associated with the NURBS description. As shown in Eq. (21), this is approximated through Gaussian quadrature, as would be done for a standard Lagrangian FEM or IGA code.

$$\int_{T_e^*} dV = \int_I \left|\boldsymbol{J}_{\mathcal{N}}(\xi^*, \eta^*)\right| d\xi^* d\eta^* \approx \sum_{i=1,j=1}^{k} \left|\boldsymbol{J}_{\mathcal{N}}(\xi_i^*, \eta_j^*)\right| w_{i,j}, \tag{21}$$

where $\left|\boldsymbol{J}_{\mathcal{N}}\right|$ is the determinant of the Jacobian for curvilinear transform $\mathcal{N}$ for the element, defined by:

$$\boldsymbol{J}_{\mathcal{N}} = \begin{pmatrix} \frac{\partial x}{\partial \xi^*} & \frac{\partial y}{\partial \xi^*} \\ \frac{\partial x}{\partial \eta^*} & \frac{\partial y}{\partial \eta^*} \end{pmatrix}. \tag{22}$$

These equations are combined to produce Eq. (23) where the integral of a variable f over T_e^* in physical (x, y) space is given by Gaussian quadrature over (ξ^*, η^*) space multiplied at each point by the value of the field variable at the same parametric quadrature position in (ξ, η) space. While existing in different spaces, the values of ξ^* and η^* will therefore match ξ and η for the sake of this integration.

$$\begin{aligned} \int_{T_e^*} \phi(x, y) dx dy &= \int_I \left|\boldsymbol{J}_{\mathcal{N}}(\xi^*, \eta^*)\right| \phi(\xi, \eta) d\xi^* d\eta^* \\ &\approx \sum_{i=1,j=1}^{k} \left|\boldsymbol{J}_{\mathcal{N}}(\xi_i^*, \eta_j^*)\right| \phi(\xi_i, \eta_j) w_{i,j}. \end{aligned} \tag{23}$$

4. The multigroup Neutron Diffusion Equation (NDE)

The strong form of the primal (or forward) steady-state neutron diffusion equation (NDE) is given by:

$$-\nabla \cdot D(\boldsymbol{x}, E) \nabla \phi(\boldsymbol{x}, E) + \Sigma_t(\boldsymbol{x}, E) \phi(\boldsymbol{x}, E) = \mathcal{H}\phi(\boldsymbol{x}, E') + Q(\boldsymbol{x}, E), \tag{24}$$

where $D(\boldsymbol{x}, E)$ is the neutron diffusion coefficient, $\phi(\boldsymbol{x}, E)$ is the neutron scalar flux, $\Sigma_t(\boldsymbol{x}, E)$ is the total macroscopic neutron cross-section, $\mathcal{H}$ is the neutron scattering operator and $Q(\boldsymbol{x}, E)$ is the neutron source. The neutron scattering operator is defined as:

$$\mathcal{H}\phi(\boldsymbol{x}, E') = \int_0^\infty \Sigma_s(\boldsymbol{x}, E' \to E) \phi(\boldsymbol{x}, E') dE' \tag{25}$$

where $\Sigma_s(\boldsymbol{x}, E' \to E)$ is the macroscopic 'single' differential neutron scattering cross-section from energy E' to E. The definition of the neutron source $Q(\boldsymbol{x}, E)$ varies depending on the specific problem type. In the case of fixed source problems this incorporates an extraneous or 'fixed' source of neutrons, generating a source term that is independent of the neutron angular flux. Fixed sources include extrinsic sources such as Cf^{252} used in reactor start-up (Martin et al., 2000) and intrinsic sources such as spontaneous fission, (α, n), and spallation neutron sources (Wilson et al., 2009). The source term in the fixed source case is given by:

$$Q(\boldsymbol{x}, E) = Q_{\text{Fix}}(\boldsymbol{x}, E), \tag{26}$$

where $Q_{\text{Fix}}(\boldsymbol{x}, E)$ is the fixed neutron source. When modelling subcritical fissile systems, an autogenous fission source is incorporated, such that:

$$Q(\boldsymbol{x}, E) = Q_{\text{Fix}}(\boldsymbol{x}, E) + Q_{\text{Fiss}}(\boldsymbol{x}, E), \tag{27}$$

where $Q_{\text{Fiss}}(\boldsymbol{x}, E)$ is the fission neutron source, defined as:

$$Q_{\text{Fiss}}(\boldsymbol{x}, E) = \chi_t(\boldsymbol{x}, E) \int_0^\infty \bar{\nu}_t(\boldsymbol{x}, E') \Sigma_f(\boldsymbol{x}, E') \phi(\boldsymbol{x}, E') dE', \tag{28}$$

where $\chi_t(\boldsymbol{x}, E)$ is the neutron energy spectrum, $\bar{\nu}_t(\boldsymbol{x}, E)$ is the average number of neutrons per fission event and $\Sigma_f(\boldsymbol{x}, E)$ is the macroscopic neutron fission cross-section. When tackling criticality problems, where multiplying media have no extraneous neutron source, we instead use an eigenvalue fission source. For such a problem, we introduce a scalar parameter "k_{eff}", which modifies the value of $\bar{\nu}_t$ to achieve a steady-state solution. Examples of such computations include the C5G7 MOX supercell (colourset) nuclear reactor physics benchmark verification

problem found in Section 5.3, which uses a source term of the form:

$$Q(\boldsymbol{x}, E) = Q_{\text{Eig}}(\boldsymbol{x}, E), \tag{29}$$

where $Q_{\text{Eig}}(\boldsymbol{x}, E)$ is the eigenvalue fission neutron source, defined as:

$$Q_{\text{Eig}}(\boldsymbol{x}, E) = \frac{\chi_t(\boldsymbol{x}, E)}{\text{k}_{\text{eff}}} \int_0^\infty \bar{\nu}_t(\boldsymbol{x}, E') \Sigma_f(\boldsymbol{x}, E') \phi(\boldsymbol{x}, E') \, dE', \tag{30}$$

where k_{eff} is the effective neutron multiplication factor, constructing an eigenvalue problem such that a steady-state solution may achieved. The fissile source terms may be further split into prompt and delayed neutron contributions, such that:

$$\bar{\nu}_t = \bar{\nu} + \sum_i \bar{\nu}_{i,d}, \tag{31}$$

where $\bar{\nu}$ is the average number of prompt neutrons and $\bar{\nu}_{i,d}$ is the average number of neutrons from the ith delayed precursor group. The total spectrum may also be split into prompt and delayed fractions, where:

$$\chi_t = \left[1 - \sum_i \beta_i\right] \chi + \sum_i \beta_i \chi_{i,d}, \tag{32}$$

where β_i is the fraction of neutrons from the ith delayed precursor group, χ and $\chi_{i,d}$ are the fission spectra from the prompt and ith delayed precursor group respectively. As $\sum \beta \approx 0.01$, we may approximate the fission neutron source to be purely comprised of prompt neutrons (Cacuci, 2010), such that we take the fission source to be:

$$Q_{\text{Eig}}(\boldsymbol{x}, E) = \frac{\chi(\boldsymbol{x}, E)}{\text{k}_{\text{eff}}} \int_0^\infty \bar{\nu}(\boldsymbol{x}, E') \Sigma_f(\boldsymbol{x}, E') \phi(\boldsymbol{x}, E') \, dE'. \tag{33}$$

The energy domain of the problem can be split into G distinct energy ranges, $E_0 > E_1 > \cdots > E_G$. These energy ranges are referred to as 'groups', where group g represents the energy range $E_{g-1} \geq E > E_g$. The multigroup approximation allows for the creation of a finite number of coupled 'multigroup equations' (Erlich et al., 1947; Hurwitz and Ehrlich, 1948, 1953). For example, a neutron scalar flux integral over the energy domain is simplified into G group-wise integrals:

$$\int_0^\infty \phi(E, \boldsymbol{x}) \, dE \approx \int_{E_G}^{E_0} \phi(E, \boldsymbol{x}) \, dE \approx \sum_{g=1}^{G} \int_{E_g}^{E_{g-1}} \phi_g(\boldsymbol{x}) \, dE. \tag{34}$$

The primal (forward) multigroup NDE is then defined as:

$$-\nabla \cdot D^g(\boldsymbol{x}) \nabla \phi^g(\boldsymbol{x}) + \Sigma_t^g(\boldsymbol{x}) \phi^g(\boldsymbol{x}) = \sum_{g'=1}^{G} \Sigma_s^{g' \to g}(\boldsymbol{x}) \phi^{g'}(\boldsymbol{x}) + Q^g(\boldsymbol{x}), \tag{35}$$

where D^g, ϕ^g and Σ_t^g are the values of D, ϕ and Σ_t in group g and $\Sigma_s^{g' \to g}$ is the value of the macroscopic neutron scattering cross-section group g' to g. We may now introduce the macroscopic neutron removal cross-section for group g, Σ_r^g, defined as:

$$\Sigma_r^g(\boldsymbol{x}) = \Sigma_a^g(\boldsymbol{x}) + \sum_{g' \neq g}^{G} \Sigma_s^{g \to g'}(\boldsymbol{x}) = \Sigma_t^g(\boldsymbol{x}) - \Sigma_s^{g \to g}(\boldsymbol{x}), \tag{36}$$

where Σ_a^g is the macroscopic neutron absorption cross-section and $\Sigma_s^{g \to g}(\boldsymbol{x})$ is the within group scatter for group g. For a problem domain V with boundary Γ, the final form of the steady-state multigroup NDE is defined as:

$$-\nabla \cdot D^g(\boldsymbol{x}) \nabla \phi^g(\boldsymbol{x}) + \Sigma_r^g(\boldsymbol{x}) \phi^g(\boldsymbol{x}) = \sum_{g' \neq g}^{G} \Sigma_s^{g' \to g}(\boldsymbol{x}) \phi^{g'}(\boldsymbol{x}) + Q^g(\boldsymbol{x}) \quad \forall \boldsymbol{x} \in V, \tag{37}$$

$$D^g(\boldsymbol{x}) \nabla \phi^g(\boldsymbol{x}) \cdot \mathbf{n} = 0 \quad \forall \boldsymbol{x} \in \Gamma_R, \tag{38}$$

$$D^g(\boldsymbol{x}) \nabla \phi^g(\boldsymbol{x}) \cdot \mathbf{n} + \frac{1}{2} \phi^g(\boldsymbol{x}) = 0 \quad \forall \boldsymbol{x} \in \Gamma_V, \tag{39}$$

where Γ_R and Γ_V are the reflective and vacuum boundary conditions respectively. Note that the treatment of boundary conditions for both the FEM and NEFEM are identical, following the standard CBG implementation. The boundary Γ of the system is a combination of these conditions, such that $\Gamma_R \cup \Gamma_V = \Gamma$ and $\Gamma_R \cap \Gamma_V = \varnothing$. This may be converted into matrix form by defining:

$$L_g = -\nabla \cdot D^g \nabla + \Sigma_r^g, \quad Q_g = Q_{\text{Fix}}^g, \quad \mathcal{H}_{g',g} = \Sigma_s^{g' \to g}, \tag{40}$$

where L_g is the multigroup destruction operator for group g (leakage and removal), Q_g is the group-dependent fixed source and $\mathcal{H}_{g',g}$ is the group to group scattering operator. The matrix form is then given by:

$$\begin{pmatrix} L_1 & 0 & \dots & 0 \\ 0 & L_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & L_G \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^G \end{pmatrix} = \begin{pmatrix} Q_1 & 0 & \dots & 0 \\ 0 & Q_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q_G \end{pmatrix} + \begin{pmatrix} 0 & \mathcal{H}_{2,1} & \dots & \mathcal{H}_{G,1} \\ \mathcal{H}_{1,2} & 0 & \dots & \mathcal{H}_{G,2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{H}_{1,G} & \mathcal{H}_{2,G} & \dots & 0 \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^G \end{pmatrix}, \tag{41}$$

with equivalent operator form:

$$L\boldsymbol{\Phi} = Q + \mathcal{H}\boldsymbol{\Phi}, \tag{42}$$

where $\boldsymbol{\Phi} = (\phi^1, \dots, \phi^G)$. The neutron scalar flux may then be found in operator form through $(L - \mathcal{H})^{-1} Q = \boldsymbol{\Phi}$. For criticality problems, we introduce the multigroup fission production operator $F_{g',g} = \chi^g \bar{\nu}^{g'} \Sigma_f^{g'}$, such that:

$$\begin{pmatrix} L_1 & 0 & \dots & 0 \\ 0 & L_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & L_G \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^G \end{pmatrix} = \frac{1}{\text{k}_{\text{eff}}} \begin{pmatrix} F_{1,1} & F_{2,1} & \dots & F_{G,1} \\ F_{1,2} & F_{2,2} & \dots & F_{G,2} \\ \vdots & \vdots & \ddots & \vdots \\ F_{1,G} & F_{2,G} & \dots & F_{G,G} \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^G \end{pmatrix} + \begin{pmatrix} 0 & \mathcal{H}_{2,1} & \dots & \mathcal{H}_{G,1} \\ \mathcal{H}_{1,2} & 0 & \dots & \mathcal{H}_{G,2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{H}_{1,G} & \mathcal{H}_{2,G} & \dots & 0 \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^G \end{pmatrix}, \tag{43}$$

with equivalent operator form:

$$L\boldsymbol{\Phi} = \frac{1}{\text{k}_{\text{eff}}} F\boldsymbol{\Phi} + \mathcal{H}\boldsymbol{\Phi}, \tag{44}$$

where k_{eff} are the eigenvalues of the system $(L - \mathcal{H})^{-1} F$, the largest of which is equivalent to the effective neutron multiplication factor k_{eff} (Habetler and Martino, 1958). The domain V may be decomposed into a mesh $\mathcal{T}$ of elements T_e such that $\mathcal{T} = \{T_e\}$ and $V = \bigcup_e T_e$, over which material properties are assumed to be piecewise constant, such that:

$$D^g(\boldsymbol{x})|_{T_e} = D^{g,e}, \tag{45}$$

$$\Sigma_r^g(\boldsymbol{x})|_{T_e} = \Sigma_r^{g,e}, \tag{46}$$

$$Q_{\text{Fix}}^g(\boldsymbol{x})|_{T_e} = Q_{\text{Fix}}^{g,e}. \tag{47}$$

As such, discontinuities between material properties may never occur within an element. This notation will be used throughout the rest of the section to simplify the associated functional analysis descriptions.

4.1. The integral or weak form of the neutron diffusion equation

The integral or weak form of an equation may be derived by multiplying a strong formulation by a 'test function' and integrating over the domain, introducing a larger number of admissible solutions. When constructing solutions to PDEs, it is significantly easier to find

approximate solutions to a weak formulation, especially where the underlying equation and associated geometry are complex. In this subsection, we shall derive the weak form of the NDE, such that we may construct a coupled system of algebraic equations, through which an approximate solution may be found.

We start by defining the Lebesgue space $L^2(V)$ as the collection of all functions u that are square integrable over domain V, such that:

$$\int_V u^2 dV < +\infty, \tag{48}$$

with inner product and norm defined as:

$$(u,v)_{L_2(V)} = \int_V uv dV; \quad \|u\|_{L_2(V)} = \left(\int_V |u|^2 dV\right)^{1/2}, \tag{49}$$

where $v \in L_2(V)$. We define ϕ^g as a trial solution that exists in Sobolev space $H^1(V)$, such that:

$$H^1(V) = \left\{\phi^g \in L^2(V) : \nabla\phi^g \in L^2(V)\right\}. \tag{50}$$

The collection of trial solutions $\mathcal{S}$ may then be defined as the set of all solutions in Sobolev space that satisfy the boundary conditions, defined by:

$$\mathcal{S} = \left\{ \phi^g \in H^1(V);\ (D^{g,e}\nabla\phi^g \cdot \boldsymbol{n})|_{\Gamma_R} = 0, \right.$$
$$\left. \left(D^{g,e}\nabla\phi^g \cdot \boldsymbol{n} + \frac{1}{2}\phi^g\right)\Big|_{\Gamma_V} = 0 \right\}. \tag{51}$$

We also define a collection of weighting functions $\mathcal{V}$, defined by:

$$\mathcal{V} = \left\{ w^g \in H^1(V); w^g|_{\Gamma_R} = 0, w^g|_{\Gamma_V} = 0 \right\}, \tag{52}$$

where w^g is a test function $w^g \in \mathcal{V}$. The weak form may be obtained by multiplying the strong form by the test function w and integrating by parts to produce the result:

$$\sum_{T_e\in\mathcal{T}} \int_{T_e} \boldsymbol{w}^T L\boldsymbol{\Phi} dV = \frac{1}{\mathrm{k_{eff}}} \sum_{T_e\in\mathcal{T}} \int_{T_e} \boldsymbol{w}^T F\boldsymbol{\Phi} dV + \sum_{T_e\in\mathcal{T}} \int_{T_e} \boldsymbol{w}^T \mathcal{H}\boldsymbol{\Phi} dV, \tag{53}$$

where $\boldsymbol{w}^T = (w^1, \dots, w^G)$ is the transpose groupwise test function vector. We may then define our bilinear forms of the equations as:

$$\hat{L}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) = \sum_{g=1}^{G} \left(\nabla w^g, D^g\nabla\phi^g\right)_V + \left(w^g, \Sigma_r^g\phi^g\right)_V + \frac{1}{2}\left(w^g, \phi^g\right)_{\Gamma_V}, \tag{54}$$

$$\hat{F}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) = \sum_{g=1}^{G}\sum_{g'=1}^{G} \left(w^g, \chi^g \bar{\nu}^{g'} \Sigma_f^{g'\to g}\phi^{g'}\right)_V, \tag{55}$$

$$\hat{\mathcal{H}}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) = \sum_{g=1}^{G}\sum_{g'\neq g}^{G} \left(w^g, \Sigma_s^{g'\to g}\phi^{g'}\right)_V. \tag{56}$$

The variational problem may then be formally posed. Find $\mathrm{k_{eff}} \in \mathbb{R}$ such that $\boldsymbol{\Phi} \in H^1(V)$ satisfies:

$$\hat{L}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) = \frac{1}{\mathrm{k_{eff}}}\hat{F}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) + \hat{\mathcal{H}}\left(\boldsymbol{w}^T, \boldsymbol{\Phi}\right) \quad \forall \boldsymbol{w}^T \in H^1(V). \tag{57}$$

4.2. The finite dimensional approximation of the weak form

Following Galerkin's method, we construct a finite-dimensional approximation to the system through subsets $S_{h,p} \subset \mathcal{S}$ and $\mathcal{V}_{h,p} \subset \mathcal{V}$. Therefore, we may reconstruct the weak form using $\phi_{h,p}^g \in S_{h,p}$ and $w_{h,p}^g \in \mathcal{V}_{h,p}$ such that:

$$\hat{L}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \frac{1}{\mathrm{k_{eff}}}\hat{F}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right)$$
$$+ \hat{\mathcal{H}}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) \quad \forall \boldsymbol{w}_{h,p}^T \in H^1(V), \tag{58}$$

where $\boldsymbol{w}_{h,p}^T = (w_{h,p}^1, \dots, w_{h,p}^G)$, $\boldsymbol{\Phi}_{h,p} = (\phi_{h,p}^1, \dots, \phi_{h,p}^G)$ and bilinear forms are defined as a summation over each element by:

$$\hat{L}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \sum_{g=1}^{G}\sum_{T_e\in\mathcal{T}} \left(\nabla w_{h,p}^g, D^{g,e}\nabla\phi_{h,p}^g\right)_{T_e}$$
$$+ \left(w_{h,p}^g, \Sigma_r^{g,e}\phi_{h,p}^g\right)_{T_e} + \frac{1}{2}\left(w_{h,p}^g, \phi_{h,p}^g\right)_{\varepsilon_V}, \tag{59}$$

$$\hat{F}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \sum_{g=1}^{G}\sum_{g'=1}^{G}\sum_{T_e\in\mathcal{T}} \left(w_{h,p}^g, \chi^{g,e}\bar{\nu}^{g',e}\Sigma_f^{g'\to g,e}\phi_{h,p}^{g'}\right)_{T_e}, \tag{60}$$

$$\hat{\mathcal{H}}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \sum_{g=1}^{G}\sum_{g'\neq g}^{G}\sum_{T_e\in\mathcal{T}} \left(w_{h,p}^g, \Sigma_s^{g'\to g,e}\phi_{h,p}^{g'}\right)_{T_e}. \tag{61}$$

where $\varepsilon_V \in \Gamma_V$ is an edge associated with element T_e with vacuum boundary conditions. The finite element solution space contains a set of basis functions N_i that allows for functions to be interpolated within the domain, such that the value of $\phi_{h,p}^g$ and $w_{h,p}^g$ in element T_e is given by:

$$\phi_{h,p}^g|_{T_e}(\boldsymbol{x}) = \sum_{i=1}^{n_e} \phi_{h,p,i}^g N_i(\boldsymbol{x}), \quad w_{h,p}^g|_{T_e}(\boldsymbol{x}) = \sum_{i=1}^{n_e} w_{h,p,i}^g N_i(\boldsymbol{x}), \tag{62}$$

where $\phi_{h,p,i}^g$, $w_{h,p,i}^g$ and N_i are the values of $\phi_{h,p}^g$, $w_{h,p}^g$ and the basis functions at node i respectively and n_e is the number of degrees of freedom associated with element T_e. Note that a continuous Bubnov–Galerkin discretisation is both globally and locally conservative (Hughes et al., 2000). The local conservation holds over the 'tent function', which is comprised of the basis functions associated with the union of elements surrounding a node. The mass and stiffness matrices $\boldsymbol{M}^{T_e}$ and $\boldsymbol{K}^{T_e}$ for each element are defined through these basis functions as:

$$\left(\boldsymbol{M}^{T_e}\right)_{i,j} = \left(N_i, N_j\right)^{T_e}, \tag{63}$$

$$\left(\boldsymbol{K}^{T_e}\right)_{i,j} = \left(\nabla N_i, \nabla N_j\right)^{T_e}. \tag{64}$$

We may incorporate these definitions into the bilinear forms such that:

$$\hat{L}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \sum_{g=1}^{G} w_{h,p}^g \left(\sum_{T_e\in\mathcal{T}} D^{g,e}\boldsymbol{K}^{T_e} + \Sigma_r^{g,e}\boldsymbol{M}^{T_e} + \sum_{\varepsilon\in\Gamma_V}\frac{1}{2}\boldsymbol{M}^{T_e}\right)\phi_{h,p}^g$$
$$= \sum_{g=1}^{G} w_{h,p}^g \boldsymbol{L}^g \phi_{h,p}^g, \tag{65}$$

$$\hat{F}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}\right) = \sum_{g=1}^{G} w_{h,p}^g \sum_{g'=1}^{G}\left(\sum_{T_e\in\mathcal{T}} \chi^{g,e}\bar{\nu}^{g',e}\Sigma_f^{g'\to g,e}\boldsymbol{M}^{T_e}\right)\phi_{h,p}^{g'}$$
$$= \sum_{g=1}^{G} w_{h,p}^g \sum_{g'=1}^{G} \boldsymbol{F}^{g,g'}\phi_{h,p}^{g'}, \tag{66}$$

$$\hat{\mathcal{H}}_{h,p}\left(\boldsymbol{w}_{h,p}^T, \boldsymbol{\Phi}_{h,p}\right) = \sum_{g=1}^{G} w_{h,p}^g \sum_{g'\neq g}^{G}\left(\sum_{T_e\in\mathcal{T}} \Sigma_s^{g'\to g,e}\boldsymbol{M}^{T_e}\right)\phi_{h,p}^{g'}$$
$$= \sum_{g=1}^{G} w_{h,p}^g \sum_{g'\neq g}^{G} \boldsymbol{\mathcal{H}}^{g,g'}\phi_{h,p}^{g'}. \tag{67}$$

Substituting these into Eq. (58) gives:

$$\sum_{g=1}^{G} w_{h,p}^g \left(\boldsymbol{L}^g\phi_{h,p}^g - \frac{1}{\mathrm{k_{eff}}}\sum_{g'=1}^{G} w_{h,p}^g \boldsymbol{F}^{g,g'}\phi_{h,p}^{g'} - \sum_{g'\neq g}^{G} w_{h,p}^g \boldsymbol{\mathcal{H}}^{g,g'}\phi_{h,p}^{g'}\right) = 0. \tag{68}$$

As the weight function $w_{h,p}^g \in \mathcal{V}$ is arbitrary, the terms within the brackets must equal zero, from which we may calculate the value of $\mathrm{k_{eff}}$ for the system. Note that in the case of a fixed source problem, we instead solve:

$$\sum_{g=1}^{G} w_{h,p}^g \left(\boldsymbol{L}^g\phi_{h,p}^g - \boldsymbol{Q}^g - \sum_{g'\neq g}^{G} w_{h,p}^g \boldsymbol{\mathcal{H}}^{g,g'}\phi_{h,p}^{g'}\right) = 0. \tag{69}$$

Similarly, the terms in brackets must be equal to zero, allowing for the value of $\phi_{h,p}^g$ to be solved for. In this case, the fixed source is given by:

$$\hat{Q}_{h,p}\left(\boldsymbol{w}_{h,p}^T\right) = \sum_{g=1}^{G} w_{h,p}^g \left(\sum_{T_e\in\mathcal{T}} Q_{\mathrm{Fix}}^{g,e}\boldsymbol{f}^e\right) = \sum_{g=1}^{G} w_{h,p}^g \boldsymbol{Q}^g, \tag{70}$$

where $\boldsymbol{f}^{T_e}$ is the force (load) vector, defined as:

$$(\boldsymbol{f}^{T_e})_{i,j} = (N_i, 1)_{T_e}. \tag{71}$$

5. Numerical results

The NEFEM spatial discretisation of the multigroup NDE is implemented within a modern Fortran code (International Organization for Standardization (ISO), 2023) that was created for the paper. This NEFEM multigroup NDE code is used to provide numerical solutions to three nuclear reactor physics benchmark verification test cases. The performance of the NEFEM multigroup NDE code is compared against a standard Lagrangian FEM approach. Each of the three nuclear reactor physics benchmark verification test problems will progressively increase in geometric and computational complexity. These three test cases will enable the numerical accuracy, and computationally efficiency, of the NEFEM to be compared against existing standard Lagrangian FEM approaches. When analysing the suitability of each method, the degrees of freedom (DoF) associated with each computational mesh are compared against a relevant error measure (e.g., k_{eff} of the system). To generate the computational meshes required for the solution of the three benchmark verification test cases a bespoke mesh generator was implemented using the python programming language. This mesh generator produced simple meshes using the multiblock transfinite interpolation (MB-TFI) algorithm with variable uniform mesh refinement (Gordon and Hall, 1973).

As finite element meshes will not preserve the volume of material in the presence of curved boundaries, the mesh generator also incorporates a volume-preserved FEM (VPFEM) algorithm. The VPFEM algorithm moves the nodes of the finite elements to ensure the volume of material in the problem is conserved in the final FEM computational mesh. Surface-preserved FEM (SPFEM) meshes can also be generated, where the nodes of the finite elements are moved such that the surface area at both internal and external boundaries matches that of the problem. This method is utilised in systems where leakage terms dominate and hence the length of surfaces between different materials is of greater importance to the overall solution. Such methods are widely utilised to mitigate errors associated with the limited geometric accuracy of FEM. Where geometries involve curved boundaries, the analysis will incorporate VPFEM and SPFEM in addition to standard Lagrangian FEM.

Analysis has been performed on superparametric 'Linear NEFEM', where a linear ($p = 1$) FEM discretisation has been enhanced with a quadratic ($p = 2$) NURBS geometry, and isoparametric 'Quadratic NEFEM', where both the FEM discretisation and NURBS geometric descriptions are quadratic ($p = 2$). An investigation of the respective computational solution times of each method and the condition numbers of the assembled global matrices is performed in Section 5.4. In addition to the three benchmark problems, a Method of Manufactured Solutions (MMS) study can be found in Appendix A, demonstrating the stability of the FEM and NURBS-enhanced codes as well as their ability to achieve the correct convergence rates. In doing so, we verify the consistency of NEFEM as a spatial discretisation method. Furthermore, an analytical Wigner–Seitz pincell problem is presented in Appendix B, which further investigates the accuracy of the method. In all cases, Gauss–Legendre quadrature of order $p + 1$ was utilised to perform the numerical quadrature. For each problem in Sections 5.1–5.3, Appendix A–B, the matrix solution algorithms implemented within the Portable, Extensible Toolkit for Scientific Computation (PETSc) library was utilised to generate the solution to a tolerance of $1E{-}12$. The 'Cholesky', 'ILU', 'Hypre' and 'Jacobi' preconditioners were used to solve the problems of Sections 5.1, 5.2, 5.3 and the Appendix respectively. Each preconditioner was selected as it was seen to effectively reduce matrix solution times over the range of refinements tested in this paper. Each of the problems were run on an AMD Ryzen 5 3600XT 3.8 GHz processor through a virtual machine running the Linux Mint operating system.

5.1. One-group nuclear fuel pincell benchmark verification test case

The first nuclear reactor physics benchmark verification test case is a one-group single nuclear fuel pincell. This is a very common problem type that is utilised widely within the nuclear industry, where problems range from single fuel pins to larger assemblies or whole reactor cores. Additionally, this provides a simple problem which first introduces the use of curvilinear surfaces. As such, this problem allows for an investigation of the differences in numerical accuracy, and computational efficiency, of NEFEM compared to the standard Lagrangian FEM in the presence of such geometric features.

The problem consists of a central nuclear fuel pin with radius 1 cm centred in a square cell of moderator with width 4 cm. A schematic of this nuclear fuel pincell benchmark verification test case is presented in Fig. 9(a). Reflective boundary conditions are prescribed to all external boundaries of the square cell which is equivalent to modelling an infinite lattice of pincells. The reference solution for this problem comes from Welch et al. (2017b), where an IGA solution of the NDE was performed using quartic ($p = 4$) NURBS patches, or elements, with 14,689 degrees of freedom. The neutron scalar flux contours for the nuclear fuel pincell problem are presented in Fig. 9(b). The material properties for this problem and the benchmark neutron scalar flux integrals are presented in Table 1

The error metric for this problem is the Cell Disadvantage Factor (CDF), defined by:

$$\text{CDF} = \frac{\bar{\phi}_{\text{Moderator}}}{\bar{\phi}_{\text{Fuel}}}, \tag{72}$$

where $\bar{\phi}$ is the average neutron scalar flux. The absolute error is used as an error measure, where:

$$\text{Error} = \left|\text{CDF} - \text{CDF}_{\text{Ref}}\right|, \tag{73}$$

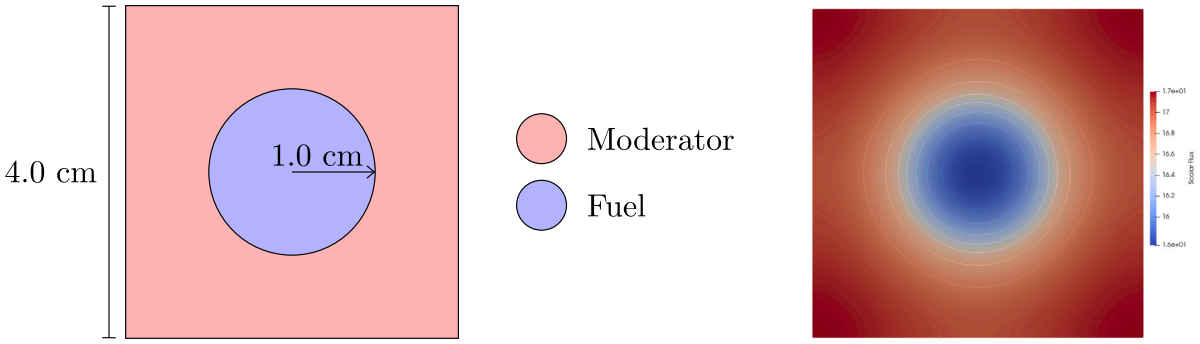
where CDF_{Ref} is the reference CDF value. Unlike purely global parameters such as the total neutron scalar flux integral, this metric compares neutron scalar fluxes in relative regions within the geometry, mitigating the effects of competing errors. The convergence plot for the nuclear fuel pincell benchmark verification test case is presented in Fig. 10.

Results for the Linear ($p = 1$) FEM methods are much as one would expect, with VPFEM outperforming SPFEM, which in turn outperforms FEM, suggesting that volumetric errors dominate in this problem. Superparametric Linear NEFEM achieves a lower error on the CDF value across all refinement levels compared to all other spatial discretisation methods.

Quadratic ($p = 2$) results for FEM follow a very similar trend to those of Linear FEM, where VPFEM achieves a much lower error on the CDF value due to its mitigation of volumetric errors in the problem. Isoparametric Quadratic NEFEM consistently achieves a lower error on the CDF compared to all other methods.

For this benchmark verification test case, the errors in volume and surface integrals of each different material have also been evaluated at each mesh refinement. This analysis will not be repeated for the Supercell and C5G7 test cases, as the results can be trivially extrapolated to fit any such problem consisting of nuclear fuel pincell assemblies. Results for quadratic spatial discretisation methods have been omitted as the same meshes were used and hence results would be identical.

Plots of the log of error in relative integrals against the refinement are presented in Figs. 11(a) and 11(b) for volume and surface respectively for linear forms. As expected, errors associated with VPFEM and SPFEM are very low across all spatial refinements for their respective corrections, as this is performed analytically. NEFEM displays similar performance for both volume and surface errors, where the numerical accuracy is simply limited by the degree of quadrature utilised in the integration of the computational mesh. Therefore, it can be seen that only the NEFEM meshes achieve high orders of geometric accuracy in both error measures. Surface corrections are shown to perform poorly in terms of their volume accuracy, with similar results shown



(a) Schematic of the nuclear fuel pincell problem.

(b) Neutron scalar flux [neutrons cm^{-2} s^{-1}] contour plot.

Fig. 9. Schematic (left) and neutron scalar flux contours (right) of the nuclear fuel pincell benchmark verification test case.

Table 1
Nuclear fuel pincell benchmark verification test case macroscopic neutron cross-section data and reference neutron scalar flux integral.

	Σ_t [cm^{-1}]	Σ_s [cm^{-1}]	Q [neutrons cm^{-3} s^{-1}]	$\int_V \phi\, dV$ [neutrons s^{-1}]
Moderator	1.00	0.95	1.00	218.424
Fuel	2.00	1.90	1.00	50.788

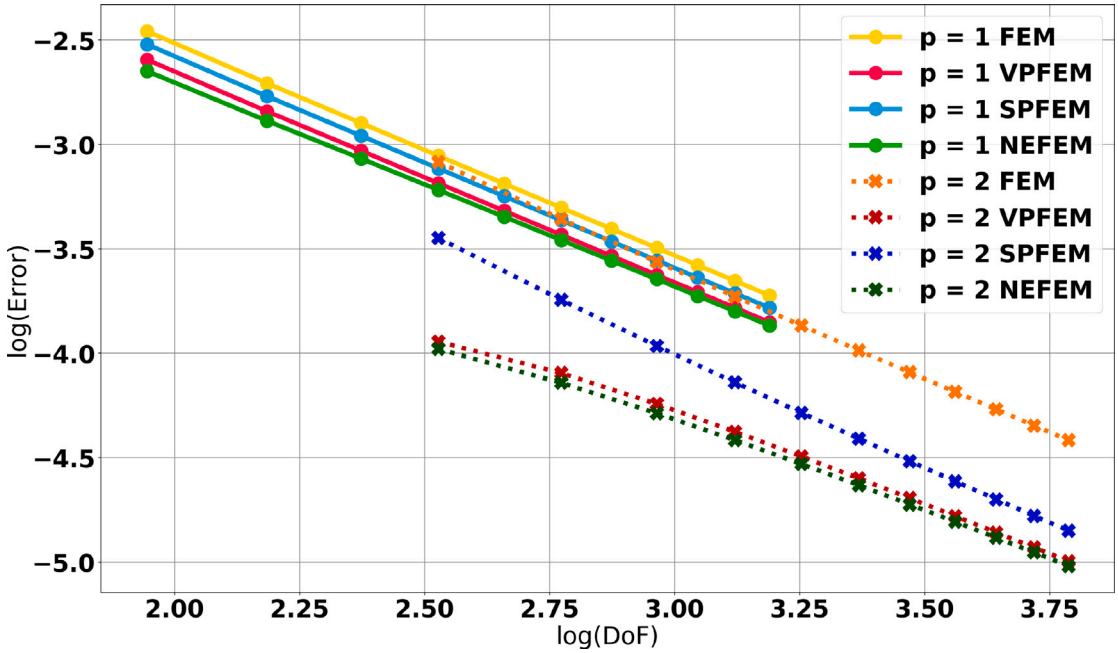
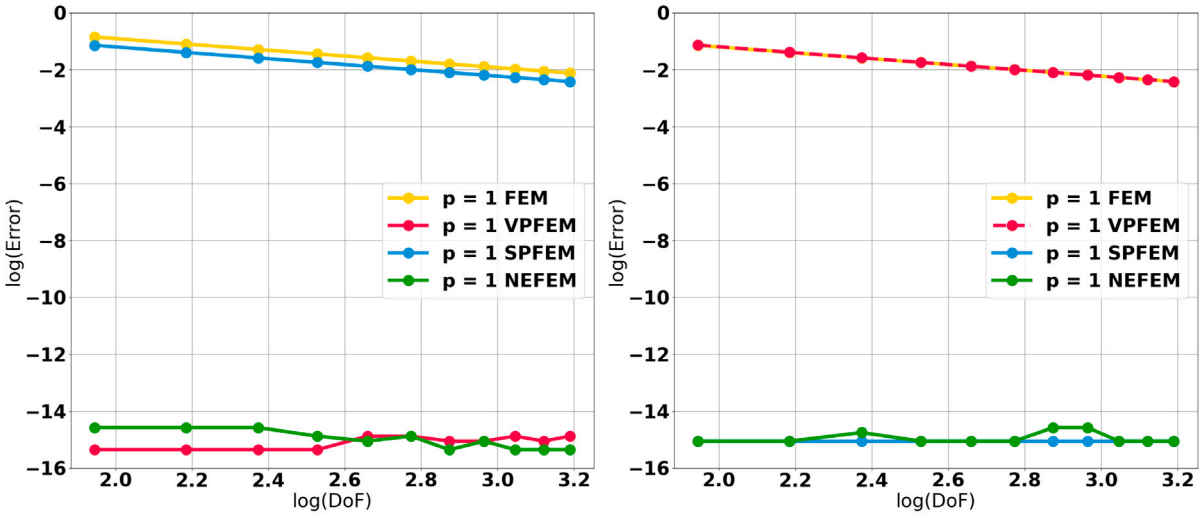


Fig. 10. Nuclear fuel pincell results of log(DoF) vs. log(Error) in the CDF for linear and quadratic spatial discretisation methods.

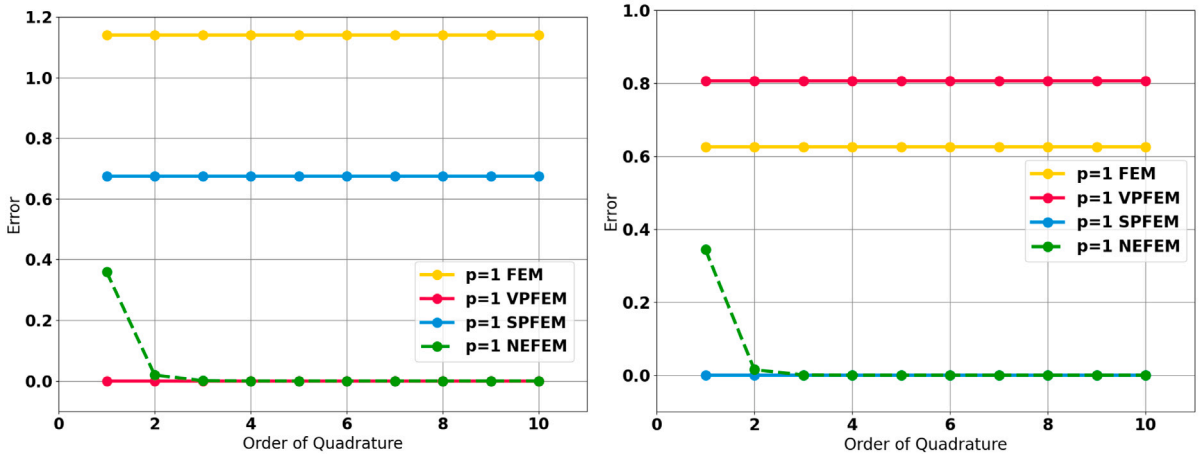
for the surface accuracy of volume correction, much as one would expect. Standard FEM performs worst across all measures, which is expected given it fails to accurately represent curved surfaces and must approximate them with straight-sided elements.

Plots of the volume and surface errors with increasing order of numerical quadrature for a pincell with 12 DoF is presented in Figs. 12(a) and 12(b) respectively. Similarly to Figs. 11(a) and 11(b), we see that the VPFEM and SPFEM achieve machine precision geometric accuracy for volume and surface errors respectively, but fail to maintain this accuracy for both metrics. Standard Lagrangian FEM once again demonstrates large geometric errors in both categories. The NEFEM is notable as its geometric accuracy clearly depends on the order of numerical quadrature employed, converging to machine precision alongside the VPFEM and SPFEM for volume and surface errors respectively. These results show that the $p+1$ order quadrature employed by FEM is sufficient to leverage the increased geometric accuracy of the NFEM, while avoiding the computational costs associated with higher order quadrature. Note that while the NEFEM clearly achieves low geometric errors at quadrature precisions $\geq p+1$, this will not hold for all geometries. A comprehensive study of the optimisation of quadrature sets and orders for different geometries with the NEFEM is outside the scope of this current paper. However, in terms of optimising the numerical accuracy, and computational efficiency, of the method this could be an important area for future research (see Table 2).



(a) Error on Volume Integral. (b) Error on Surface Integral.

Fig. 11. Pincell results of log(DoF) vs. log(Error) of volume integral (left) and surface integral (right) for linear FEM methods.



(a) Error on Volume Integral. (b) Error on Surface Integral.

Fig. 12. Pincell results of log(DoF) vs. log(Error) of volume integral (left) and surface integral (right) for linear FEM methods.

5.2. One-group square supercell with burnable poison pin nuclear reactor physics benchmark verification test case

The second problem used to investigate the suitability of NEFEM is the one-group Supercell benchmark, a neutron transport problem created by Wood and Williams (1984) and later adapted for isotropic scatter for testing neutron diffusion codes by Owens et al. (2017a). The problem involves a 3 × 3 array of pincells of pitch 2 cm. The central pin is filled with two strongly absorbing materials of radii 0.55 cm and 0.71 cm, while the other 8 pins are filled with fuel of radius 0.71 cm. A schematic of this problem is seen in Figure Fig. 13(a), with material properties and reference flux integrals from Owens et al. (2017a) given in Table Table 2. The scalar flux for this problem is shown in Figure Fig. 13(b). This problem has been chosen as it involves an increased degree of geometric complexity compared to the pincell case, incorporating multiple annuli and extremely sharp flux gradients in the central absorbing pin. This should allow for more comprehensive testing of NEFEM's suitability in more complex cases where the leakage terms dominate in certain regions.

Convergence plots for the three linear FEM discretisations are very closely grouped, suggesting the inclusion of strong leakage. Therefore, the effects of surface errors results in both VPFEM and SPFEM failing to wholly mitigate geometric errors in the problem. The linear superparametric NEFEM achieves a lower error than its equivalent standard Lagrangian FEM counterpart across all refinements, as it is better able to quantify both the volumetric and surface leakage terms (see Figs. 13 and 14).

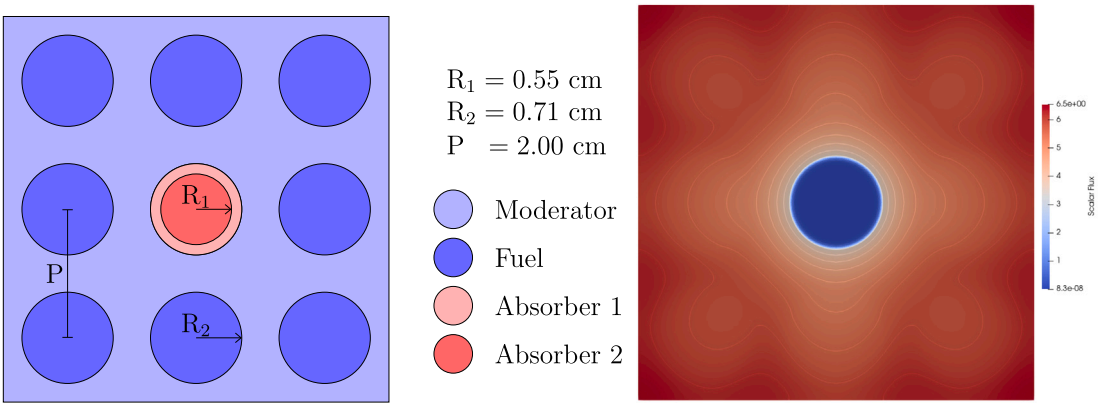
Results for the quadratic FEM methods show a more significant spread, as reductions in geometric errors associated with VPFEM and SPFEM allow for significant reductions in error compared to standard quadratic FEM. Quadratic isoparametric NEFEM achieves the lowest error across all refinements due to its geometric error mitigations, consistently achieving errors greater than an order of magnitude lower than FEM.

5.3. C5G7 nuclear reactor physics benchmark verification test case

The final problem that will be used to investigate the suitability of NEFEM for reactor physics applications is that of the NEA C5G7 quarter-core reactor physics benchmark. This is a seven-group colourset nuclear reactor physics benchmark verification test case. This problem is representative of larger whole-core type calculations performed within the nuclear industry. Note that while the C5G7 reactor physics benchmark problem is generally utilised with transport codes that solve

Table 2
Supercell nuclear reactor physics benchmark verification test case macroscopic neutron cross-section data and reference neutron scalar flux integrals.

	Σ_t [cm^{-1}]	Σ_a [cm^{-1}]	Q [neutrons cm^{-3} s^{-1}]	$\int_V \phi\, dV$ [neutrons s^{-1}]
Moderator	0.7557	0.00530	1.0	124.426023
Fuel	0.5145	0.17130	0.0	72.1396107
Absorber 1	21.630	21.2886	0.0	0.409144631
Absorber 2	14.270	13.9300	0.0	0.00142273066



(a) Schematic of the supercell problem.

(b) Neutron scalar flux [neutrons cm^{-2} s^{-1}] contour plot.

Fig. 13. Schematic (left) and neutron scalar flux contours (right) of the supercell nuclear reactor physics benchmark verification test case.

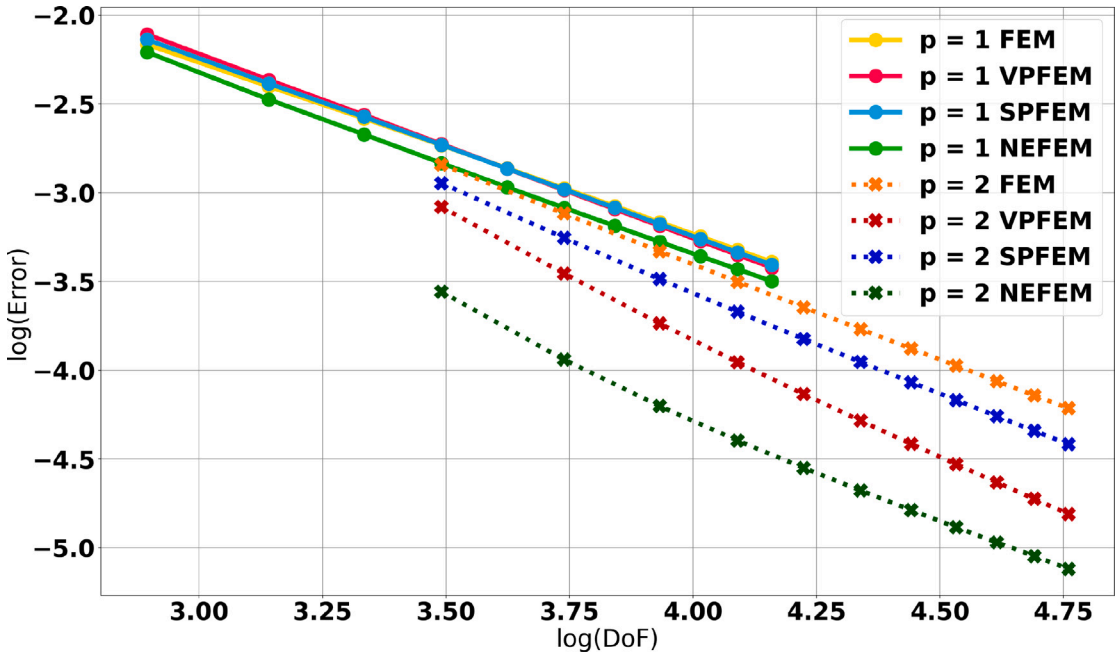


Fig. 14. Supercell results of log(DoF) vs. log(Error) in the CDF for linear and quadratic discretisation methods.

the NTE, diffusion forms of the benchmark have been widely used, in papers such as Welch et al. (2017b) and Ferguson et al. (2021) as well as the original documentation describing the benchmark (Lewis et al., 2001). This benchmark verification test case is often used to analyse and assess numerical methods used within nuclear reactor physics (Lewis et al., 2001). A schematic of this nuclear reactor physics problem is presented in Fig. 15. It is a quarter reactor core made up of two uranium dioxide (UO_2) and two mixed oxide (MOX) 17×17 pincell assemblies of width 21.42 cm arranged in a square lattice surrounded by a moderating region. Each of the four assemblies contains 24 guide tubes and a central fission chamber. The computational domain has reflective boundary conditions prescribed on the leftmost and top edges to allow the geometry to model a full core, with vacuum boundaries prescribed on the rightmost and lower edges. A zoomed-in view of an individual nuclear fuel pincell from the C5G7 geometry is presented in Fig. 16. Each nuclear fuel pin in the geometry is of radius 0.54 cm in a moderating region of width 1.26 cm. Demonstration of the ability of NEFEM to solve such a complex problem would provide evidence for its suitability in solving large-scale industrial nuclear reactor physics problems. As described by Lewis in the original benchmark, the neutron

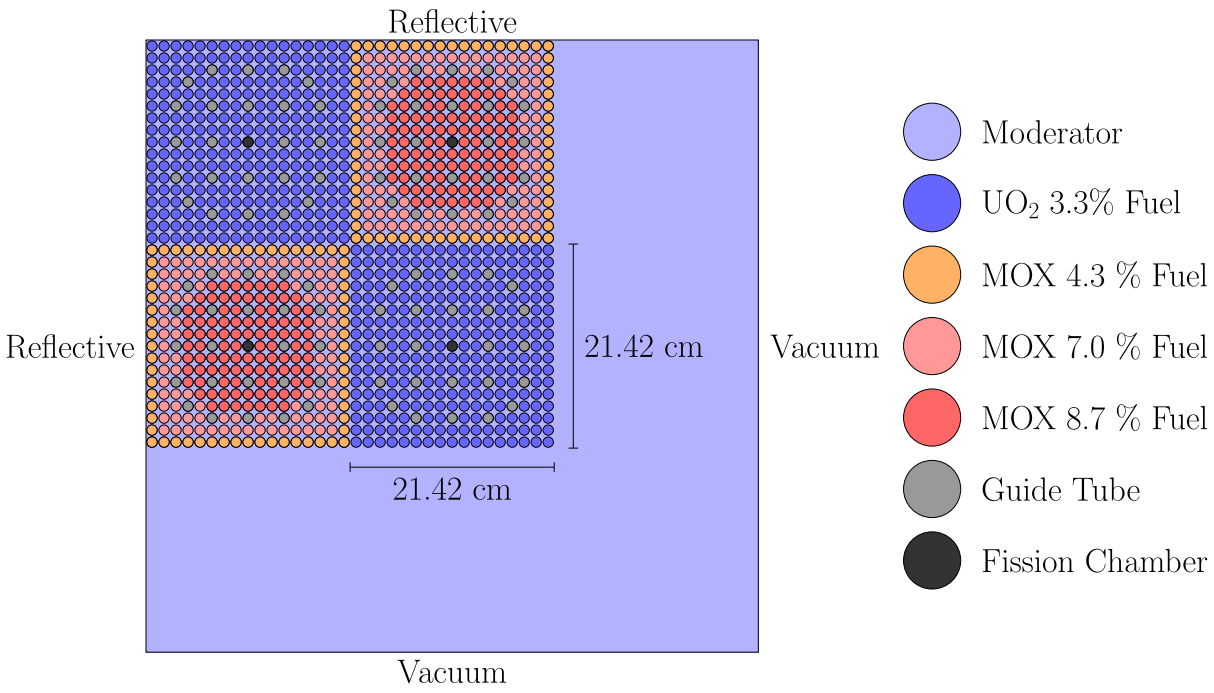


Fig. 15. Schematic of the C5G7 quarter core nuclear reactor physics benchmark verification test case.

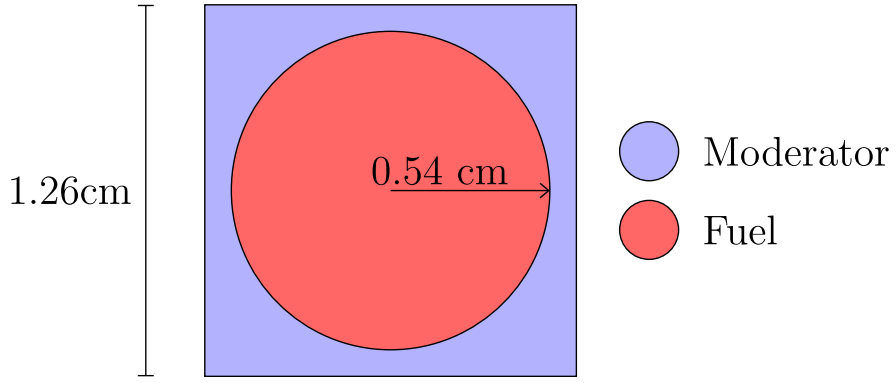


Fig. 16. Schematic of a single nuclear fuel pincell in the C5G7 quarter core nuclear reactor physics benchmark verification test case.

scalar flux is normalised such that the total power in the system is equal to the number of fuel pins (Lewis et al., 2001), such that:

$$\int_V \Sigma_f(\boldsymbol{x})\,\phi(\boldsymbol{x})\,dV = 1056. \tag{74}$$

Benchmark data for the C5G7 problem focuses on the k_{eff} of the system as well as the powers of the individual pins in the problem. Errors on the calculated pin power values are generally quantified through measures of the maximum and minimum pin powers in the system, the total power of the inner and outer UO_2 assemblies as well as the total power of the MOX assembly. These parameters will be referred to as k_{eff}, P_{Max},P_{Min}, $P_{UO_2\text{ - Outer}}$, $P_{UO_2\text{ - Inner}}$ and P_{MOX} respectively. For this convergence study, a benchmark case was created by performing a large $6.51E{+}06$ degree of freedom ($p = 1$) VPFEM solve, with results presented in Table 3. As presented in Table 4, the k_{eff} result was compared against an IGA solve from Welch et al. (2017b) and VEM (Virtual Element Method) solve from Ferguson et al. (2021), matching both to within 1 pcm. Convergence plots showing the numerical error in each of these values plotted against the degrees of freedom associated with the computational meshes is presented in Figs. 17 and 18. As with CDF values from Sections 5.1 and 5.2, error metrics in all cases are taken as the absolute difference between the measured and the reference value.

Fig. 17 presents the error on the k_{eff} of the system against spatial refinement. As seen in the previous test cases, NEFEM and VPFEM both perform much better than standard Lagrangian FEM and SPFEM, as they are better able to quantify the volumetric terms in the system. While VPFEM achieves a lower error for lower spatial refinement problems (3.86 and 4.46 $\log(DoF)$), the NEFEM is more accurate for higher spatial refinements. Pin power results from Figs. 17(a)–17(e) all generally show NEFEM outperforming standard Lagrangian FEM and SPFEM. However, in all cases VPFEM is a more accurate measure of pin power in the system across all refinements. Fig. 17(d) does show NEFEM performing poorer than SPFEM at high spatial refinements when quantifying the power of the inner UO_2 assembly. Overall, linear superparametric NEFEM has been shown to be a competitive alternative to standard Lagrangian FEM and the volume and surface-preserving corrections. It is shown to be more accurate in terms of calculating the k_{eff} value for the C5G7 problem at high refinements but is outperformed by VPFEM at low refinements and in the quantification of pin powers.

Convergence plots for quadratic ($p = 2$) FEM and NEFEM are presented in Figs. 18–(e). These results follow a generally similar trend to that of linear methods in terms of the relative accuracy of each discretisation type. NEFEM is shown to be more accurate at quantifying the k_{eff} in Fig. 18 at higher spatial refinement, while

Table 3
Neutron diffusion reference results for a 6.51E+06 degrees of freedom $p = 1$ VPFEM solution.

k_{eff}	P_{Max}	P_{Min}	P_{Inner}	P_{Outer}	P_{MOX}
1.1832413	2.5231256	0.2409044	496.4155635	138.5309895	210.5267236

Table 4
VPFEM neutron diffusion k_{eff} reference results of Table 3 compared to results from Welch et al. (2017b) and Ferguson et al. (2021).

	VPFEM reference result	Welch et al. (2017b)	Ferguson et al. (2021)
k_{eff}	1.1832413	1.1832383	1.1832408
Error (pcm)	–	0.257	0.046

VPFEM performs best at lower spatial refinements. Pin power results from Figs. 18(a)–18(e) show that the NEFEM and VPFEM methods generally achieve a very similar error to one another, with VPFEM only performing noticeably better than NEFEM in quantifying the value of P_{Min} and $P_{UO_2\ -\ Outer}$ in Figs. 18(b) and 18(c) respectively. Results for quadratic ($p = 2$) isoparametric NEFEM show that it is an even more competitive alternative to VPFEM for the C5G7 problem than its linear superparametric counterpart.

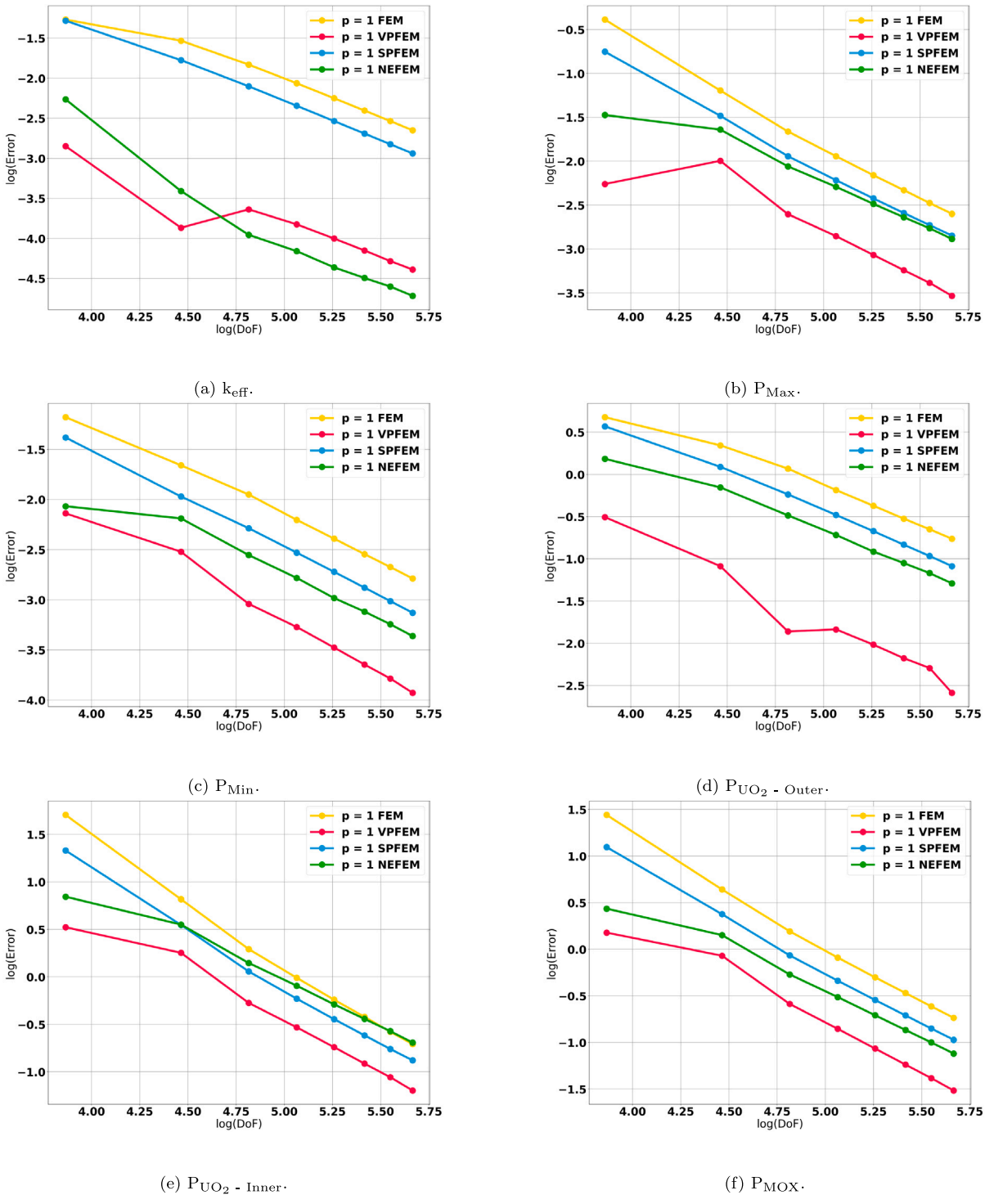
5.4. Solution times for the NEFEM

With the computational accuracy of the NEFEM compared against Lagrangian FEM over a range of different reactor physics benchmark verification tests cases, we now seek to investigate the relative solution times for each method. The solution times for linear ($p = 1$) and quadratic ($p = 2$) spatial discretisation methods are presented in Tables 5 and 6 respectively. For each spatial discretisation, a computational mesh with 2.91E+04 DoF for the NEA C5G7 reactor physics benchmark of Section 5.3 was utilised, with timings taken as the mean average of 10 successive runs. All problems were run serially on an AMD Ryzen 5 3600XT 3.8 GHz processor through a virtual machine running Linux Mint. The 'Hypre' preconditioner and PETSc toolkit was utilised to perform power iterations on the assembled system of algebraic equations to a tolerance of 1E−12. The computational solution time is given in seconds as a total, as well as being separated into pre-process, matrix solution, post-process and mesh generation times, along with the number of power iterations required for the solve. Pre-processing covered reading in material and geometry data and setting up the problem, while the matrix solution time included assembling the linear system of equations and iteratively solving them using the Portable, Extensible Toolkit for Scientific Computation (PETSc) software package. Post-processing included the computation of global and local quantities of interest (QoI) such as k_{eff} and pin powers, as well as the generation of data for visualisations using the Paraview software (Quammen, 2015). Mesh generation provides the total time taken by the python script to generate meshes for each relevant discretisation. As the meshing script was separate to the Fortran NDE code, this is not included in the 'Total' compute time. Note that in the case of the VPFEM and SPFEM, the 'Total' timings do not account for the additional mesh modification time required for these methods, whereas 'Total' timings for the NEFEM incorporate the additional setup required to link the CAGD description to the FEM mesh in the pre-processing stage. It should also be mentioned that the mesh generation times may not be wholly indicative of the timings that would be associated with a commercial NEFEM code. The python mesh generator used in this study spent additional time generating NURBS descriptions of each region of the pincell geometries and matching it to the relevant finite element. Optimisation of the timings associated with this process, and the writing of additional data to mesh input files, was deemed to be outside the scope of this investigation and a potential area for future study.

For linear methods, in Table 5, we observe relatively similar timings for each discretisation method, Lagrangian FEM and SPFEM being the joint slowest of the methods, which is understandable given the increased number of power iterations that each method needed to achieve convergence. A notable increase in pre-process time is seen for the NEFEM, which is expected given that more data must be read in for this method such that we may NURBS-enhance relevant elements. The mesh generation times for the VPFEM and SPFEM are only marginally longer than that of Lagrangian FEM, which is expected given the relative simplicity of volume and surface preservation for two-dimensional pincell geometries. A marked increase in mesh generation time is seen for the NEFEM, as a result of the additional steps taken to construct the mesh and the writing of additional data to file. Similar results are seen in Table 6, where Lagrangian FEM and SPFEM necessitate a greater number of iterations for convergence, and hence take longer to solve overall. The NEFEM once again demonstrates a marked increase in the overall mesh generation time for quadratic methods, however increase is an order of magnitude smaller than the overall solve times of the NDE code for each of the spatial discretisations.

As the comparison of solve times for each method is partially obfuscated by the fact that each take a different number of iterations to converge, we also present Tables 7 and 8, where the linear ($p = 1$) and quadratic ($p = 2$) spatial discretisation methods have been restricted to 100 power iterations. Linear results in Table 7 show a much narrower spread of overall solution times, as we would expect given the iteration restriction, with the SPFEM and the NEFEM taking slightly longer overall. In the case of the NEFEM, this may be attributed to the longer pre-processing time associated with the addition of the CAGD information, as gedwas seen in previous results. The solution time for the NEFEM is similar to each of the Lagrangian FEM methods, with an overall time between the VPFEM and the SPFEM. These results are broadly reflected in the quadratic results of Table 8, where the NEFEM achieves competitive solution times, but suffers from an increased pre-processing time. Despite this, the slightly faster solve time of the quadratic NEFEM allows it to achieve a faster overall compute time than its Lagrangian counterparts. Overall, these results suggest that any increase in solution time associated with the NEFEM is only found in the pre-processing steps, which account for a relatively small portion of the overall compute time. This fact, along with the faster convergence times seen for the NEFEM, demonstrates the computational efficiency of the method.

The similarity in solution times displayed in the previous tables of the Lagrangian FEM, VPFEM, SPFEM and the NEFEM suggests that the conditioning of the matrices for each method may display similar properties. This is expected, given that the same quadrature set is used for each discretisation, with meshes that are broadly similar. In addition, the NEFEM only modifies a small minority of elements that lie on curvilinear boundaries, so the vast majority of the global matrix will be unchanged. To verify this, the condition numbers of the global matrix for each spatial discretisation of the one-group nuclear fuel pincell problem of Section 5.1 over a range of refinements is presented in Figs. 19(a) and 19(b) for linear and quadratic discretisations respectively. Note that in both cases, condition numbers are evaluated for a global matrix with no preconditioning. Both plots show that the conditioning of each matrix is very similar to one another at any given degree or spatial refinement of the computational mesh, which in turn results in very similar compute times for each of the methods.



(a) k_{eff}. (b) P_{Max}.

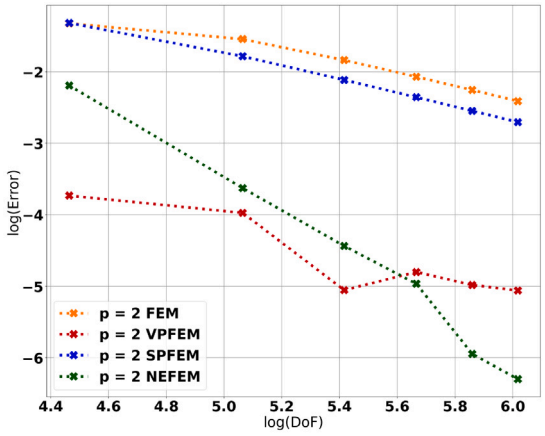
(c) P_{Min}. (d) $P_{UO_2\text{ - Outer}}$.

(e) $P_{UO_2\text{ - Inner}}$. (f) P_{MOX}.

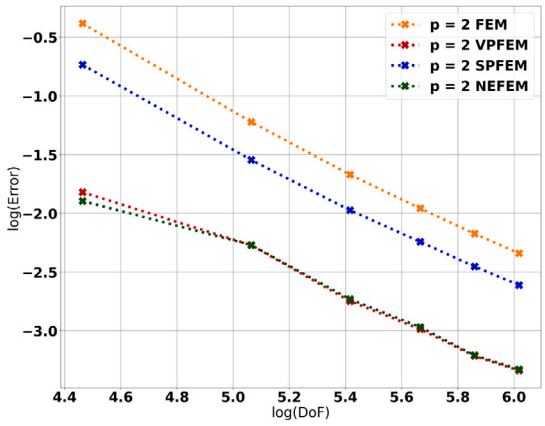
Fig. 17. Plots of relative refinement against (a) k_{eff}, (b) Maximum pin power, (c) Minimum pin power, (d) UO_2 outer power, (e) UO_2 inner power, (f) MOX power for the C5G7 problem for linear spatial discretisations.

Table 5
Computational solution times in seconds for linear spatial discretisations of the C5G7 benchmark verification test case with 2.91E+04 degrees of freedom for the Fortran NDE code and Python mesh generator.

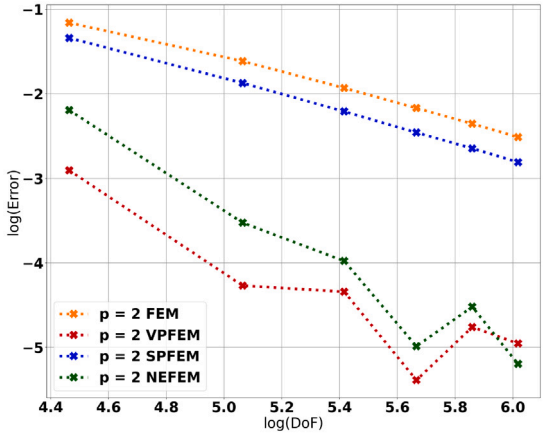
	p = 1 FEM	p = 1 VPFEM	p = 1 SPFEM	p = 1 NEFEM
Total	4.10E+01	3.94E+01	4.10E+01	4.01E+01
Pre-process	4.85E+00	4.79E+00	4.76E+00	5.50E+00
Solve	3.40E+01	3.24E+01	3.40E+01	3.26E+01
Post-process	2.23E+00	2.22E+00	2.23E+00	2.01E+00
Power iterations	131	123	127	122
Mesh generation	2.05E+00	2.09E+00	2.09E+00	3.97E+00



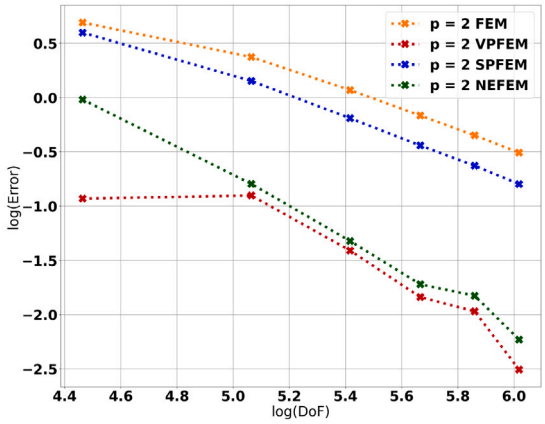
(a) k_{eff}.



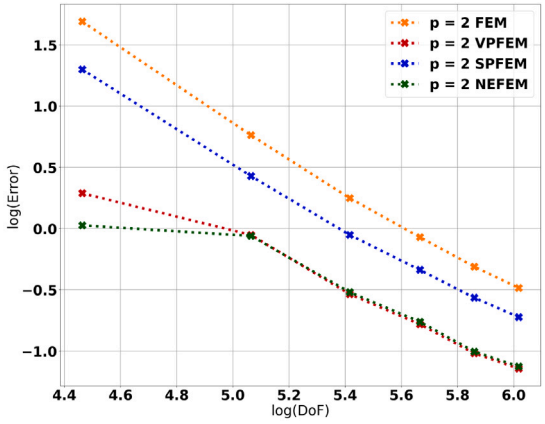
(b) P_{Max}.



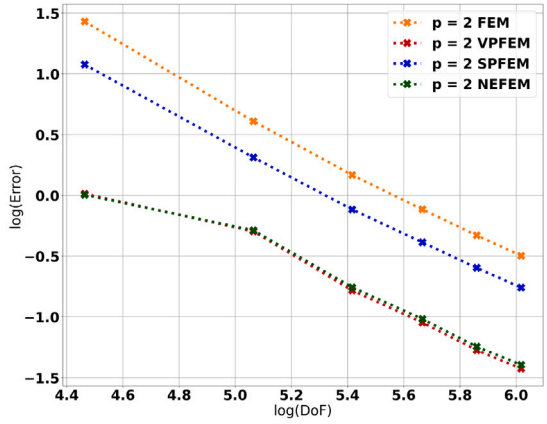
(c) P_{Min}.



(d) $P_{UO_2\text{ - Outer}}$.



(e) $P_{UO_2\text{ - Inner}}$.



(f) P_{MOX}.

Fig. 18. Plots of relative refinement against (a) k_{eff}, (b) Maximum pin power, (c) Minimum pin power, (d) UO_2 outer power, (e) UO_2 inner power, (f) MOX power for the C5G7 problem for quadratic spatial discretisations.

6. Conclusion

This paper has presented the NURBS-enhanced finite element method (NEFEM), which was applied to the steady-state multigroup neutron diffusion equation (NDE) for the first time. The NEFEM discretisation method was compared to FEM as well as volume and surface-preservation techniques that are currently used to mitigate the geometric errors associated with FEM. Linear ($p = 1$) superparametric NEFEM and quadratic ($p = 2$) isoparametric NEFEM discretisations were investigated. The numerical accuracy, and computational efficiency, of the NEFEM spatial discretisation method was analysed and assessed using a one-group nuclear fuel pincell problem, a one-group supercell problem and the NEA C5G7 NDE benchmark verification test case. The computational solution time was investigated for the NEFEM

Table 6
Computational solution times in seconds for quadratic spatial discretisations of the C5G7 benchmark verification test case with 2.91E+04 degrees of freedom for the Fortran NDE code and Python mesh generator.

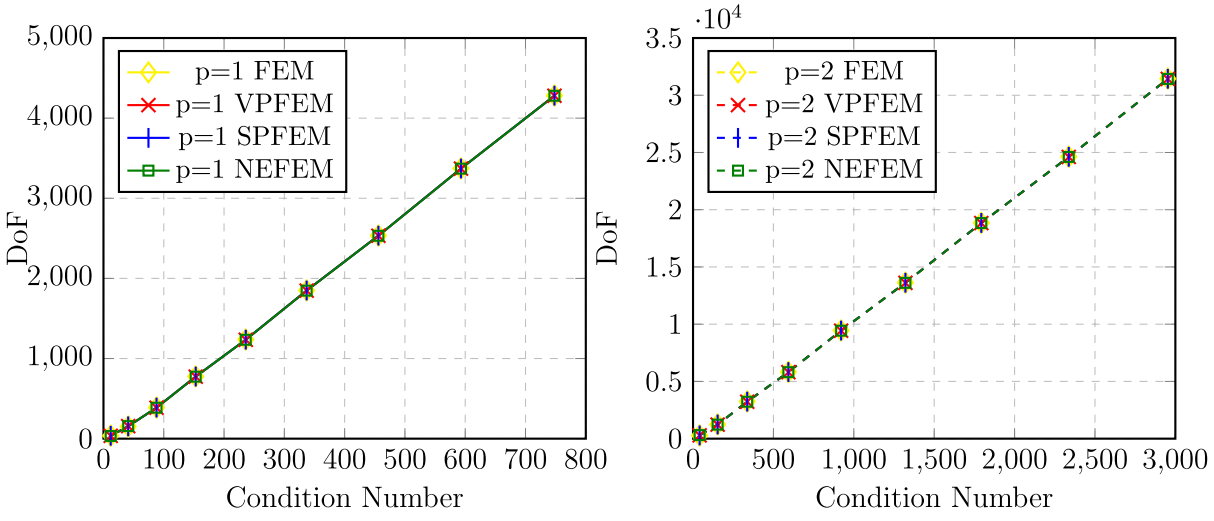
	p = 2 FEM	p = 2 VPFEM	p = 2 SPFEM	p = 2 NEFEM
Total	3.97E+01	3.87E+01	4.46E+01	3.68E+01
Pre-process	5.36E−01	5.40E−01	5.41E−01	7.28E−01
Solve	3.84E+01	3.74E+01	4.33E+01	3.53E+01
Post-process	7.62E−01	7.62E−01	7.62E−01	7.69E−01
Power iterations	134	122	142	122
Mesh generation	9.61E−01	9.74E−01	9.70E−01	2.06E+00

Table 7
Computational solution times in seconds for linear spatial discretisations of the C5G7 benchmark verification test case with 2.91E+04 degrees of freedom for the Fortran NDE code and Python mesh generator, limited to 100 power iterations.

	p = 1 FEM	p = 1 VPFEM	p = 1 SPFEM	p = 1 NEFEM
Total	3.47E+01	3.49E+01	3.59E+01	3.59E+01
Pre-process	4.47E+00	4.48E+00	4.48E+00	5.22E+00
Solve	2.79E+01	2.81E+01	2.91E+01	2.85E+01
Post-process	2.32E+00	2.33E+00	2.33E+00	2.14E+00
Mesh generation	2.05E+00	2.09E+00	2.09E+00	3.97E+00

Table 8
Computational solution times in seconds for quadratic spatial discretisations of the C5G7 benchmark verification test case with 2.91E+04 degrees of freedom for the Fortran NDE code and Python mesh generator, limited to 100 power iterations.

	p = 2 FEM	p = 2 VPFEM	p = 2 SPFEM	p = 2 NEFEM
Total	3.33E+01	3.46E+01	3.61E+01	3.29E+01
Pre-process	5.56E−01	5.39E−01	5.38E−01	7.25E−01
Solve	3.19E+01	3.33E+01	3.48E+01	3.13E+01
Post-process	8.09E−01	8.10E−01	8.09E−01	8.16E−01
Mesh generation	9.61E−01	9.74E−01	9.70E−01	2.06E+00



(a) Linear Spatial discretisation methods

(b) Quadratic Spatial discretisation methods

Fig. 19. Nuclear fuel pincell results of DoF vs. Condition number for linear and quadratic spatial discretisation methods. Note condition numbers are so similar that they are indistinguishable on the plot.

and the Lagrangian FEM for the NEA C5G7 benchmark, with global matrix conditioning investigated for the one-group nuclear fuel pincell benchmark. In addition, a method of manufactured solution (MMS) test case was used to rigorously determine the order of convergence of the NEFEM spatial discretisation method. Finally, an analytical Wigner–Seitz pincell problem is presented in Appendix B to provide further evidence of the accuracy of the method.

Results for the one-group nuclear fuel pincell problem showed that NEFEM consistently outperformed FEM for both linear and quadratic spatial discretisations. This accuracy improvement was particularly noticeable for quadratic NEFEM, which at lower spatial refinements was almost an order of magnitude more accurate. Volume-Preserved FEM (VPFEM) performed similarly to NEFEM, although it was slightly less accurate across all spatial refinements. This suggests that the numerical accuracy of the standard Lagrangian FEM was limited by volumetric errors associated with the linear finite element spatial discretisation. Surface-Preserved FEM performed better than FEM, but poorer than VPFEM. This is further evidence of a volumetric-dominated problem as, despite SPFEM accurately representing leakage terms, the correction

to the computational mesh in a nuclear fuel pincell case results in a slightly more accurate spatial discretisation. The relative geometric accuracies of each discretisation was then shown, where the volume and surface errors associated with each discretisation were quantified. It was shown that NEFEM could achieve similar accuracy to the analytically adjusted methods due to its exact CAGD representation.

The one-group supercell problem presented a benchmark with increased geometric complexity and regions where leakage terms dominate. Once again, NEFEM achieved higher levels of accuracy than FEM across all spatial refinements for both linear and quadratic spatial discretisations. For linear spatial discretisations, VPFEM and SPFEM failed to achieve significant error reductions and were actually less accurate at low spatial refinement levels than standard Lagrangian FEM. This shows that while SPFEM failed to accurately model the volumetric terms in the underlying geometry, VPFEM in turn failed to accurately model the leakage, particularly around the central absorber. NEFEM conversely, can model both accurately and as such achieves a lower error. This effect is exacerbated in the quadratic results, where NEFEM performs significantly better than FEM, VPFEM and SPFEM. Quadratic VPFEM achieves a much lower accuracy than the quadratic NEFEM, in contrast to the nuclear fuel pincell results, demonstrating that the VPFEM struggles to accurately model problems in the presence of high leakage terms.

Convergence plots for the NEA C5G7 LWR benchmark showed that the NEFEM once again consistently performed better than standard Lagrangian FEM. For linear spatial discretisations, the NEFEM achieved a lower accuracy than FEM across all six error measures for every level of mesh refinement. In terms of k_{eff} accuracy, the NEFEM was most accurate at higher refinements, only being beaten by the VPFEM at the lowest refinement levels. In all other pin power measures, the VPFEM was more accurate than the NEFEM. Results were similar for quadratic discretisations, with the NEFEM showing a consistent improvement in accuracy compared to FEM. Once again, it was only more accurate than the VPFEM at high refinements when performing error measures on the k_{eff} value. The discrepancy between the NEFEM and VPFEM for pin power errors was generally smaller than for the linear discretisations, particularly for P_{Max}, $P_{UO_2\ -\ Inner}$ and P_{MOX}.

An investigation into the computational solution times associated with the NEFEM showed that it was competitive with Lagrangian FEM for both linear and quadratic spatial discretisations of the NEA C5G7 LWR benchmark, performing particularly well in the quadratic case. While the NEFEM necessitates an increase in pre-processing time to read in the CAGD data and set up the NURBS-enhancement of relevant elements, this time is a relatively small portion of the overall compute time, with a value of ≈1–2 orders of magnitude smaller than the total solution time. In both cases, the reduced number of iterations and therefore time taken to solve the system of equations for the NEFEM allowed it to achieve lower overall compute times than the Lagrangian FEM. The mesh generation times for each spatial discretisation was also investigated. This demonstrated that the time required for the mesh corrections associated with the VPFEM and SPFEM was negligible, which one would expect for simple geometries such as the pincell assemblies found in the NEA C5G7 LWR Benchmark. A notable increase in mesh generation time was noted for the NEFEM, due to the additional steps of matching each finite element to an associated CAGD NURBS description and writing this data to file. This additional meshing time was significantly smaller than the overall solve time. It should be noted that the optimisation of the mesh generator was not prioritised as this was outside the scope of the investigation. An optimised mesh generator written in a compiled language such as C++ or Fortran would significantly improve upon these times and as such the authors do not believe that this is not representative of a significant issue if the NEFEM were to be employed in a commercial or industrial code. This investigation also verified that the conditioning of NEFEM matrices was similar to that of Lagrangian FEM over a range of spatial refinements for linear and quadratic discretisations of the one-group nuclear fuel pincell benchmark.

Overall, the NEFEM is highly effective at mitigating geometric errors, consistently improving upon the accuracy of standard Lagrangian FEM for solving the multigroup NDE. It has also been shown to be a competitive alternative to the volume and surface-preserving methods that are widely used to alleviate issues associated with the meshing of complex geometric domains. For two of the three test cases, NEFEM achieved a lower error across all mesh refinements for both linear superparametric and quadratic isoparametric formulations. For the final C5G7 quarter core nuclear reactor physics benchmark test case, it was shown to outperform both standard Lagrangian FEM and SPFEM, and was competitive with VPFEM. The NEFEM achieved a lower error for the k_{eff} value at higher spatial refinements. However, this came at the cost of a slight increase in the overall computational solution time. Additionally, the geometric accuracy of NEFEM was achieved without the mesh modification processes required for the VPFEM and SPFEM. This potentially presents an opportunity for reducing the time required to produce computational meshes for complex multidimensional curvilinear nuclear reactor physics and shielding geometries. While the NEFEM does necessitate increased pre-processing compute times, this is negligible compared to the overall solution time, and in some cases the faster convergence seen for the NEFEM compared to Lagrangian FEM allows it to achieve faster computational solution times. The relatively small pre-processing time required, combined with the numerical accuracy improvements achieved by the NEFEM and the streamlining of the computer aided geometric design (CAGD) to computer aided engineering (CAE) modelling and simulation (M&S) pipeline suggest that this is a very worthwhile trade-off (Ferguson et al., 2022; Zou et al., 2024a; Otoguro et al., 2018; Sevilla et al., 2011b; Montanari et al., 2024; Kirilov et al., 2024; Zou et al., 2024b).

Further work on NEFEM for nuclear reactor physics and shielding problems should focus on the development of curvilinear mesh generation software to streamline the CAGD to CAE M&S pipeline. This is a challenging area of research for three-dimensional (3D) complex geometry domains but it is important in terms of realising the full potential of the NEFEM spatial discretisation method (Ferguson et al., 2022; Zou et al., 2024a; Otoguro et al., 2018; Sevilla et al., 2011b; Montanari et al., 2024; Kirilov et al., 2024; Zou et al., 2024b). In addition, research into both continuous and discontinuous adaptive mesh refinement (AMR) algorithms for the NEFEM should also be explored. The NEFEM spatial discretisation should also be applied to both first and second-order forms of the neutron transport equation. Finally, an investigation into the optimisation of a purpose built mesh generator for the NEFEM should be performed to fully investigate the additional time required to set up the NURBS-Enhanced mesh information.

CRediT authorship contribution statement

J. Trainor: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **M.D. Eaton:** Conceptualization, Funding acquisition, Methodology, Project administration, Resources, Supervision, Writing – review & editing. **J. Kópházi:** Writing – review & editing. **S.G. Wilson:** Writing – review & editing. **C. Latimer:** Writing – review & editing. **L. Smith:** Funding acquisition, Writing – review & editing. **D. Baker:** Writing – review & editing. **I. Jordan:** Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: J. Trainor reports financial support was provided by Engineering and Physical Sciences Research Council. J. Trainor reports financial support was provided by Rolls-Royce plc. S. Wilson reports a relationship

with Rolls-Royce plc that includes: employment. C. Latimer reports a relationship with Rolls-Royce plc that includes: employment. L. Smith reports a relationship with Rolls-Royce plc that includes: employment. D. Baker reports a relationship with Rolls-Royce plc that includes: employment. I. Jordan reports a relationship with Rolls-Royce plc that includes: employment. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Mr J. Trainor gratefully acknowledges the financial support of the Engineering and Physical Science Research Council (EPSRC) through the Nuclear Energy Futures (NEF) Centre for Doctoral Training (CDT) (EPSRC Grant No.: EP/S023844/1). Mr J. Trainor and Dr M.D. Eaton would like to acknowledge the financial support of Rolls-Royce. Mr. J. Trainor would like to gratefully acknowledge the assistance of C. Jones in generating the Wigner–Seitz pincell problem.

Appendix A. Method of manufactured solution

This section will describe the methodologies utilised in the journal paper to generate and solve Method of Manufactured Solution (MMS) problems for the NDE. MMS aims to test the basic functionality of NEFEM, where convergence plots will aim to demonstrate that the method is stable and can reproduce the required theoretical order of accuracy of $\mathcal{O}(h^{p+1})$ in the L^2 Error. The MMS sees widespread use in assessing the numerical accuracy and convergence of spatial discretisation methods, with applications to neutron diffusion problems found in publications such as Ferguson (2022). The mono-energetic NDE is:

$$-\nabla \cdot D(\boldsymbol{x})\nabla\phi(\boldsymbol{x}) + \Sigma(\boldsymbol{x})\phi(\boldsymbol{x}) = S(\boldsymbol{x}) \tag{A.1}$$

Let ψ be a chosen MMS neutron scalar flux solution. This gives the equation:

$$-\nabla \cdot D(\boldsymbol{x})\nabla\psi(\boldsymbol{x}) + \Sigma_r(\boldsymbol{x})\psi(\boldsymbol{x}) = Q_\psi(\boldsymbol{x}) \tag{A.2}$$

where $Q_\psi(\boldsymbol{x})$ is the MMS source. This can easily be calculated by generating the linear system of equations A, equal to the global mass matrix M + the global stiffness matrix K multiplied by the MMS flux ψ. We can then solve the following system using any neutron diffusion solution method:

$$-\nabla \cdot D(\boldsymbol{x})\nabla\phi(\boldsymbol{x}) + \Sigma(\boldsymbol{x})\phi(\boldsymbol{x}) = Q_\psi(\boldsymbol{x}) \tag{A.3}$$

The solution ϕ to this equation should naturally be ψ to some level of accuracy associated with the neutron diffusion solution method (mesh refinement etc.). This accuracy must then be evaluated to check for convergence in the code. Where the MMS neutron scalar flux can be analytically integrated over the domain Γ, the total neutron scalar flux integral can be used as a measure of integration accuracy. For example, for a unit square ($[0,1]^2$), a manufactured solution of $\psi = \sin(\pi x)\cdot\sin(\pi y)$ can be integrated to give:

$$\int_0^1\int_0^1 \sin(\pi x)\cdot\sin(\pi y)dxdy = \frac{2}{\pi}\int_0^1 \sin(\pi y)dy = \frac{4}{\pi^2} \tag{A.4}$$

This is then used as a benchmark value, against which the resulting neutron scalar flux generated by the neutron diffusion solution method is evaluated. To calculate this integral, the neutron scalar flux is integrated over each element, approximated with Gaussian quadrature. Gauss–Legendre quadrature of order $p+1$ was used throughout, maintaining consistency with all other benchmarks included in this journal paper.

$$\sum_{E\in V}\int_E \phi dV \approx \sum_{E\in V}\sum_{i,j}^{n,m} \phi\left(\xi_i,\eta_j\right)\cdot\left|\boldsymbol{J}\left(\xi_i,\eta_j\right)\right|\cdot w_{i,j} \tag{A.5}$$

Table A.9
MMS problem neutron cross-section data.

	Σ_t	D
R1	1.00	0.333
R2	2.00	0.167

Table A.10
Gradients of standard Lagrangian FEM and NEFEM convergence for the MMS problem.

	FEM	NEFEM
Linear	2.00090734	2.00090734
Quadratic	2.99323768	2.99323768

where i,j are the quadrature points in each parametric dimension, n,m are the associated total number of quadrature points in each parametric dimension, $\phi\left(\xi_i,\eta_j\right)$ is the neutron scalar flux at quadrature point (i,j), $\left|\boldsymbol{J}\left(\xi_i,\eta_j\right)\right|$ is the determinant of the Jacobian at the quadrature point and $w_{i,j}$ is the quadrature weighting factor at the quadrature point. The L_2 error norm is used to define the error in this study, defined by:

$$\|\phi-\psi\|_{L_2} = \sqrt{(\phi-\psi)^2}, \tag{A.6}$$

The stability of the neutron diffusion solution method was tested using the MMS method described above. A Schematic of the problem can be seen in Fig. A.20(a). The computational domain for the MMS benchmark verification test case is a unit square $R = [0,1]^2$, split into two regions R_1 and R_2, as presented in Fig. A.20(a). The prescribed MMS solution is:

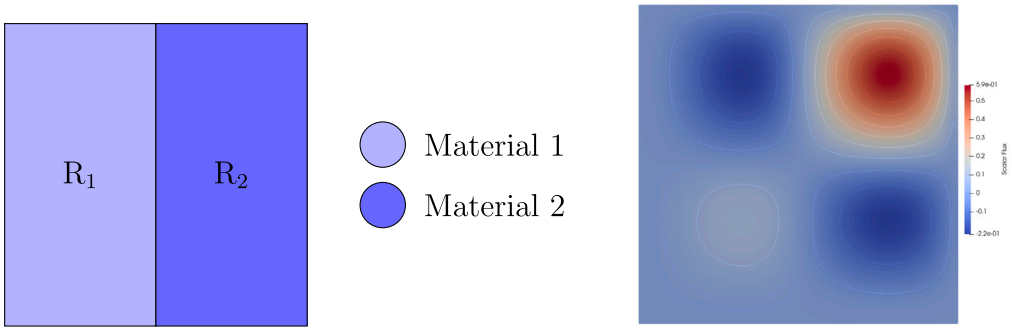
$$\psi = x\cdot y\cdot\sin(2\pi x)\cdot\sin(2\pi y) \tag{A.7}$$

This neutron scalar flux solution is presented in the contour plot in Fig. A.20(b). Dirichlet boundary conditions are prescribed at the top and right edges of the domain while Neumann boundaries are prescribed at the bottom and right edges of the domain. The Dirichlet boundaries are 'zero', enforcing a neutron scalar flux value of zero ($\psi_\Gamma = 0$), while the Neumann boundaries are reflective, ensuring that the derivative of the neutron scalar flux is zero ($\nabla\psi_\Gamma = 0$). The domain, V, is meshed into a simple orthogonal, structured cartesian grid as seen in Fig. A.21. As no curved surfaces exist on any of the boundaries, Γ, in the problem, we would expect NEFEM to produce results that exactly match FEM. Material properties utilised for the test case are presented in Table A.9.

Fig. A.21 presents the computational mesh for the MMS problem, progressively refined such that convergence plots can be generated. Plots of the log of the error in the MMS neutron scalar flux against the log of the relative refinement of the computational mesh (h) for the FEM and NEFEM are presented in Fig. A.22 for both first ($p=1$) and second ($p=2$) order elements. Gradients of each convergence line are given for FEM and NEFEM in Table A.10. Both FEM and NEFEM match the theoretical $\mathcal{O}(h^{p+1})$ rate of convergence to within two decimal places, demonstrating their stability.

Appendix B. Analytical Wigner–Seitz pincell

In this section, we seek to further investigate the performance of the NEFEM in the presence of curvilinear boundaries for an analytical problem. The purpose of this one-dimensional (1D) infinite cylindrical analytical Wigner–Seitz pincell benchmark is to verify the numerical accuracy, and convergence rates, of two-dimensional (2D) straight sided FEM, SPFEM, VPFEM, curvilinear Lagrangian FEM and NEFEM neutron diffusion solutions. This investigation aims to supplement the results presented in Section 5, as well as Appendix A, which have demonstrated the numerical accuracy and convergence rates of the NEFEM when compared to standard Lagrangian FEM. For this, we



(a) Schematic of the MMS problem.

(b) Neutron scalar flux [neutrons cm^{-2} s^{-1}] contour plot.

Fig. A.20. Schematic (left) and neutron scalar flux contours (right) of the two-region MMS problem.

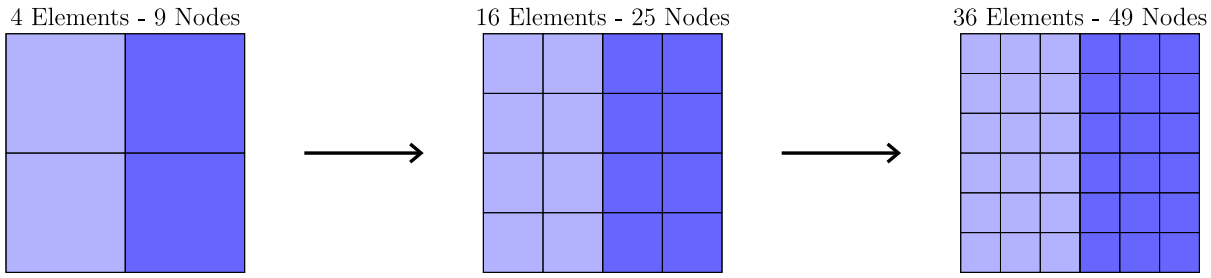


Fig. A.21. Progressively refined MMS computational meshes for convergence calculations.

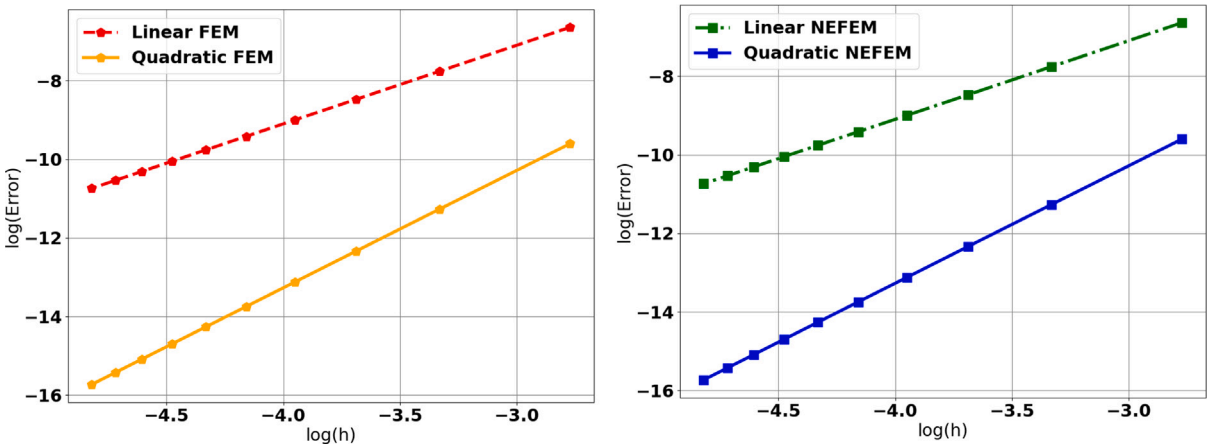


Fig. A.22. Convergence plots for the MMS problem for standard Lagrangian FEM (left) and NEFEM (right).

construct a two-region circular fuel element consisting of a central circular region surrounded by a larger ring of another material.

This type of problem is known as a Wigner–Seitz pincell approximation (Lewis and Miller, 1984). This approximation reduces a two-dimensional (2D) pincell problem with prescribed reflective boundary conditions, into an approximate 1D infinite cylindrical problem, with a prescribed white boundary condition, for neutron transport problems. However, instead of a neutron transport solution a neutron diffusion approximation is utilised, with a prescribed reflective boundary condition (zero net neutron current) on the outer surface of the pincell, in order to obtain an analytical solution to the 1D infinite cylindrical pincell problem.

This is shown in Fig. B.23, where the outer square region of length W is replaced with a circular region of radius R_2. The inner radius, R_1, remains unchanged. The replacement of the square outer region of the geometry with an annuli allows for a one-dimensional flux solution to be constructed in cylindrical co-ordinates. The material properties for the problem are defined in Table B.11.

Table B.11
Wigner–Seitz pincell benchmark verification test case macroscopic neutron cross-section data.

	Σ_t [cm^{-1}]	Σ_s [cm^{-1}]	Q [neutrons cm^{-3} s^{-1}]
Region 1	0.764	0.397	0.0
Region 2	0.38468	0.385	1.0

The derivation begins by defining $L = \sqrt{D/\Sigma_a}$ as well as I_n and K_n as the nth order modified Bessel functions of the first and second kind respectively. While the full derivation of the analytical solution has been omitted for brevity, the final analytical solution for each region is

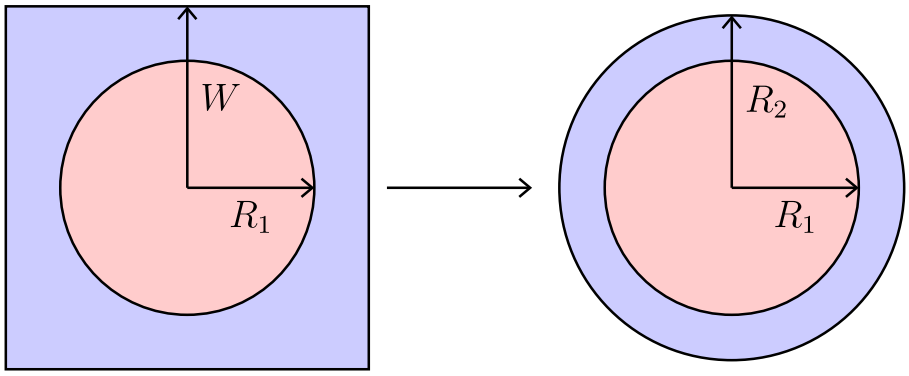


Fig. B.23. Wigner–Seitz pincell approximation. The square outer region of length W is replaced by an outer ring of radius R_2.

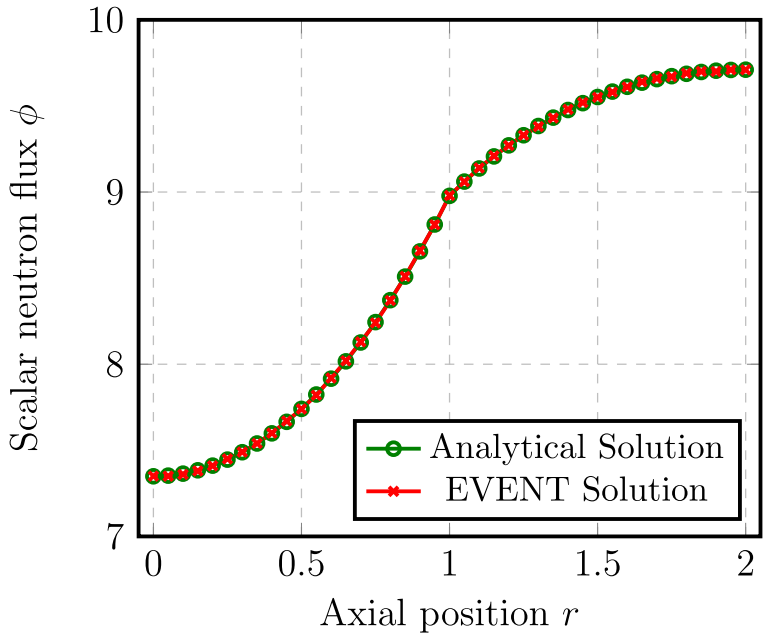


Fig. B.24. Neutron scalar flux of the Wigner–Seitz pincell problem with increasing axial position r for the analytical solution and from the EVENT code (de Oliveira, 1986).

provided here as:

$$\phi(r) = \frac{L_2^2 S}{D_2}\beta I_0\left(\frac{r}{L_1}\right), \quad 0 \le r \le R_1, \tag{B.1}$$

$$\phi(r) = \alpha\beta S_2 \frac{D_1}{D_2}\frac{L_2^2}{L_1} I_1\left(\frac{R_1}{L_1}\right)\left[I_0\left(\frac{r}{L_2}\right) + \frac{I_1\left(\frac{R_2}{L_2}\right)}{K_1\left(\frac{R_2}{L_2}\right)} K_0\left(\frac{r}{L_2}\right)\right] + \frac{L_2^2 S}{D_2}, \quad R_1 \le r \le R_2, \tag{B.2}$$

where:

$$\alpha = \left[\frac{D_2}{L_2} I_1\left(\frac{R_1}{L_2}\right) - \frac{D_2}{L_2}\frac{I_1\left(\frac{R_2}{L_2}\right)}{K_1\left(\frac{R_2}{L_2}\right)} K_1\left(\frac{R_1}{L_2}\right)\right]^{-1}, \tag{B.3}$$

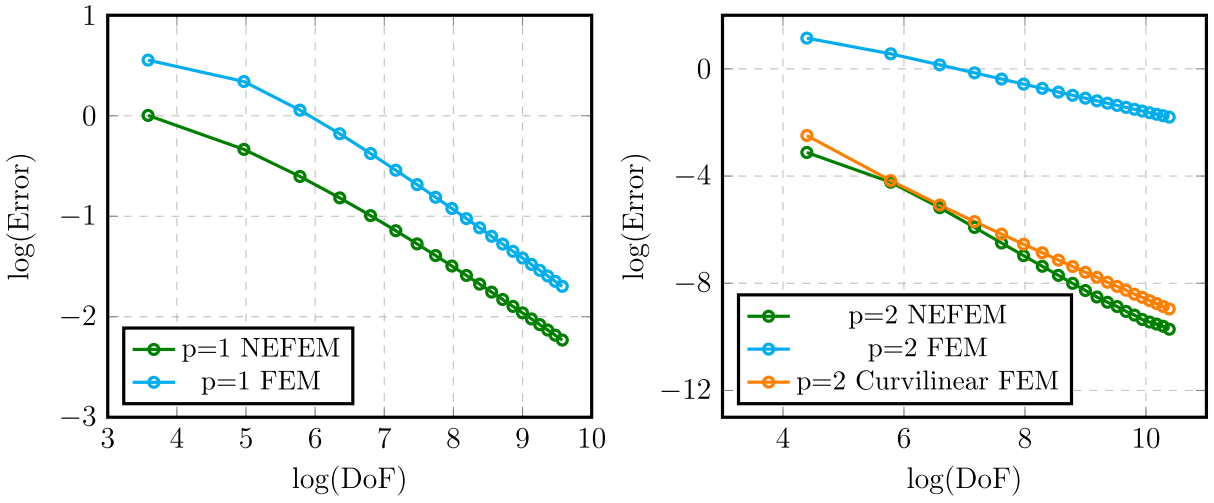
$$\beta = \left[I_0\left(\frac{R_1}{L_1}\right) - \alpha\frac{D_1}{L_1} I_0\left(\frac{R_1}{L_2}\right) I_1\left(\frac{R_1}{L_1}\right) - \alpha\frac{D_1}{L_1} I_1\left(\frac{R_1}{L_1}\right) K_0\left(\frac{R_1}{L_2}\right)\frac{I_1\left(\frac{R_2}{L_2}\right)}{K_1\left(\frac{R_2}{L_2}\right)}\right]^{-1}. \tag{B.4}$$

Fig. B.24 presents the analytical neutron scalar flux described in Eqs. (B.1)–(B.4) with respect to the axial radius r, alongside a result generated by a 4E+03 DoF linear FEM solve of the EVENT code (de Oliveira, 1986). This quantity may be analytically integrated over the volume of each region, producing the results:

$$\int_0^{2\pi}\int_0^{R_1} \phi(r) r dr d\phi = 2\pi R_1 L_1 \frac{L_2^2 S_2 \beta}{D_2} I_1\left(\frac{R_1}{L_1}\right), \tag{B.5}$$

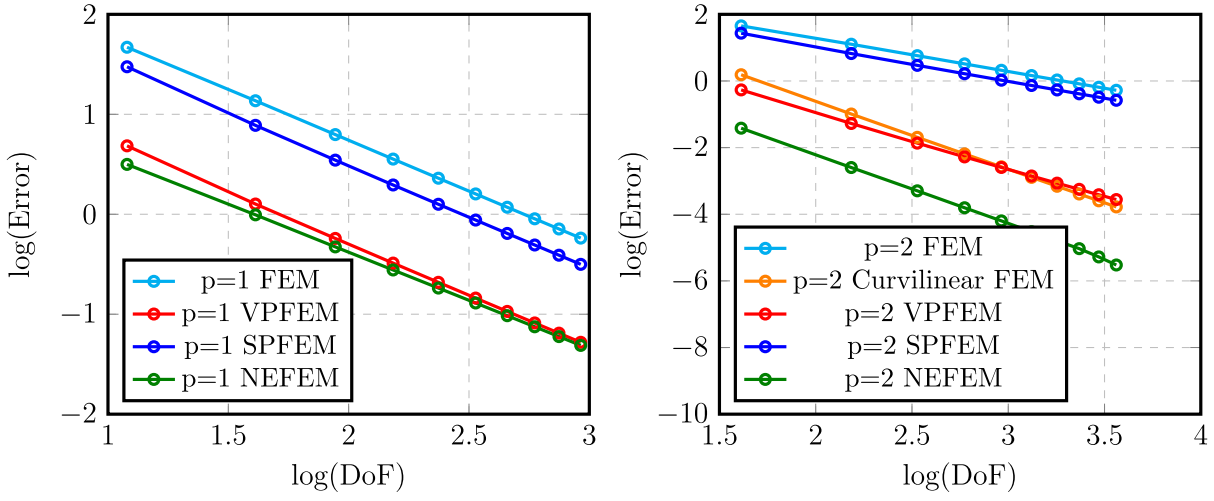
$$\int_0^{2\pi}\int_{R_1}^{R_2} \phi(r) r dr d\theta = 2\pi\alpha\beta S_2 \frac{D_1}{D_2}\frac{L_2^3}{L_1} I_1\left(\frac{R_1}{L_1}\right) \times \left[R_2 I_1\left(\frac{R_2}{L_2}\right) - R_1 I_1\left(\frac{R_1}{L_2}\right) + \frac{I_1\left(\frac{R_2}{L_2}\right)}{K_1\left(\frac{R_2}{L_2}\right)}\left(R_1 K_1\left(\frac{R_1}{L_2}\right) - R_2 K_1\left(\frac{R_2}{L_2}\right)\right)\right] + \pi\frac{L_2^2 S_2}{D_2}\left(R_2^2 - R_1^2\right). \tag{B.6}$$

Note that the subscript on each material property in Eqs. (B.1)–(B.6) denotes if it is from region 1 or 2. The numerical values of these integrals, produced by Eqs. (B.5) and (B.6) as well the result from the EVENT code are presented in Table B.12.



(a) Linear ($p = 1$) FEM and NEFEM (b) Quadratic ($p = 2$) FEM and NEFEM

Fig. B.25. Convergence plots of the local error on the neutron scalar flux for the Wigner–Seitz Pincell Problem for the FEM and NEFEM spatial discretisations of linear ($p = 1$) and quadratic ($p = 2$) polynomial orders.



(a) Linear ($p = 1$) FEM and NEFEM (b) Quadratic ($p = 2$) FEM and NEFEM

Fig. B.26. Convergence plots of the global error on the total integrated volumetric neutron scalar flux integral for the Wigner–Seitz Pincell Problem for the FEM and NEFEM spatial discretisations of linear ($p = 1$) and quadratic ($p = 2$) polynomial orders.

Table B.12
Wigner–Seitz pincell benchmark verification test case neutron scalar flux integrals in each region from the analytical solution and EVENT code (de Oliveira, 1986).

	Analytical	EVENT
Region 1	25.60234	25.60234
Region 2	89.75131	89.75132

A comprehensive overview of the approximation and associated methods for deriving the analytical solution may be found in reference texts such as Lewis and Miller (1984) and Okajima et al. (2024). The analytical flux profile, shown in Eqs. (B.1)–(B.4) and Fig. B.24 as well as the total neutron scalar flux of Table B.12 has been verified with the EVENT neutron transport code. One-dimensional (1D) infinite cylindrical (exact geometry) pointwise and volume integrated neutron scalar fluxes were generated using the EVENT (EVEn parity Neutral particle Transport) radiation transport modelling and simulation (M&S) software. The EVENT software is based upon the second order even-parity form of the neutron transport equation. It utilises a Lagrangian finite element (FE) representation of the spatial variables and a spherical harmonic (P_N) expansion in the angular variables (de Oliveira, 1986). The 1D cylindrical computational mesh for EVENT was generated using the GEM mesh generator and data pre-processing software (Wood and Williams, 1984). In order to produce a neutron diffusion solution a P_1 expansion is used within EVENT along with a prescribed reflective boundary condition (zero net neutron current) on the outer surface of the pincell.

Similarly to the MMS problem of Appendix A, a local error metric is quantified by comparing the neutron scalar flux, ϕ, of the solution to the analytical result, as described in Eq. (A.6) over each numerical quadrature point when constructing the integral. In addition to linear and quadratic spatial discretisations of the standard Lagrangian FEM and NEFEM, a curvilinear quadratic spatial discretisation has also been incorporated. This should provide a competitive method to quadratic NEFEM as it will achieve much lower geometric errors than the standard straight-sided quadratic Lagrangian FEM. It has been included as the VPFEM and SPFEM are not compatible with the local analytical flux solution due to their modification of the underlying mesh geometry.

Convergence plots of the local error on the neutron scalar flux against the DoF are presented for linear and quadratic spatial discretisations in Figs. B.25(a) and B.25(b) respectively. Fig. B.25(a) demonstrates that the linear NEFEM consistently outperforms Lagrangian FEM over all spatial discretisations investigated, likely due to the effective mitigation of geometric errors. Similarly, the quadratic NEFEM achieves significant error reductions over the standard Lagrangian quadratic FEM of Fig. B.25(b). The quadratic curvilinear FEM implementation is much more competitive due to its improved geometric accuracy, but still fails to outperform the quadratic NEFEM over all spatial discretisations investigated.

Figs. B.26 instead uses the error on the total volumetric scalar flux integral when compared to the analytical values of Table B.12. This global error metric is included to supplement the analysis performed in Fig. B.25. As the total integrated volumetric neutron scalar flux integral for the Wigner–Seitz pincell is analysed, the numerical values produced by surface and volume preserving spatial discretisations of the FEM are re-introduced into the analysis. Figs. B.26(a) and B.26(b) present convergence plots for the error on the total integrated volumetric neutron scalar flux integral for linear and quadratic spatial discretisations respectively. Fig. B.26(a) demonstrates the effectiveness of the NEFEM in mitigating geometric errors for curvilinear problems, outperforming standard Lagrangian FEM, SPFEM and VPFEM across each spatial discretisation investigated. The VPFEM once again demonstrates that is has greater numerical accuracy than SPFEM for volumetric quantities of interest (QoI) as a total integrated volumetric neutron scalar flux for nuclear reactor physics problems, as shown extensively in Section 5. Results from Fig. B.26(b) show similar results, with the NEFEM achieving the lowest error of all spatial discretisations at all but the lowest refinement investigated. Notably, the curvilinear Lagrangian FEM implementation achieves a lower numerical error than the standard Lagrangian FEM and SPFEM. This is likely due to its more accurate spatial representation. However, this still fails to achieve the numerical accuracy of the NEFEM or VPFEM. This result may be due to the fact that the curvilinear Lagrangian FEM does not exactly represent the curvilinear geometry of the pincell problem (i.e., it does not preserve the volume or surface geometry of the pincell). The numerical results presented in Figs. B.25 and B.26 further demonstrate the numerical accuracy, and convergence rates, of the NEFEM for curvilinear nuclear reactor physics problems.

Data availability

In accordance with EPSRC funding requirements, all supporting data used to create figures in this paper may be accessed at the following DOI: https://zenodo.org/records/15607929.

References

Abdelfattah, A., Barra, V., Beams, N., Bleile, R., Brown, J., Camier, J.-S., Carson, R., Chalmers, N., Dobrev, V., Dudouit, Y., et al., 2021. GPU algorithms for efficient exascale discretizations. Parallel Comput. 108, 102841. http://dx.doi.org/10.1016/j.parco.2021.102841.

Amaldi, E., 1984. From the discovery of the neutron to the discovery of nuclear fission. Phys. Rep. 111 (1), 1–331. http://dx.doi.org/10.1016/0370-1573(84)90214-X.

Barbarino, A., Dulla, S., Mund, E., Ravetto, P., 2013. The spectral element method for static neutron transport in A_N approximation. Part I. Ann. Nucl. Energy 53, 372–380. http://dx.doi.org/10.1016/j.anucene.2012.10.013.

Barbarino, A., Dulla, S., Mund, E., Ravetto, P., 2015. The spectral element method for static neutron transport in A_N approximation. Part II. Ann. Nucl. Energy 86, 108–115. http://dx.doi.org/10.1016/j.anucene.2015.04.015.

Bazilevs, Y., Calo, V.M., Cottrell, J.A., Evans, J.A., Hughes, T.J.R., Lipton, S., Scott, M.A., Sederberg, T.W., 2010. Isogeometric analysis using T-splines. Comput. Methods Appl. Mech. Engrg. 199 (5–8), 229–263. http://dx.doi.org/10.1016/j.cma.2009.02.036.

Betzler, B.R., Ade, B.J., Wysocki, A.J., Jain, P.K., Chesser, P.C., Greenwood, M.S., Terrani, K.A., 2020. Transformational challenge reactor preconceptual core design studies. Nucl. Eng. Des. 367, 110781. http://dx.doi.org/10.1016/j.nucengdes.2020.110781.

Boltzmann, L., 1872. Weitere studien über das wärmegleichgewicht unter gasmolekülen. Sitzungsberichte Akad. Wiss. 66, 275–370. http://dx.doi.org/10.1007/978-3-322-84986-1_3.

Boyd, W., Gibson, N., Forget, B., Smith, K., 2018. An analysis of condensation errors in multi-group cross section generation for fine-mesh neutron transport calculations. Ann. Nucl. Energy 112, 267–276. http://dx.doi.org/10.1016/j.anucene.2017.09.052.

Brough, M.D., Chudley, C.T., 1980. Characteristic ray solutions of the transport equation. Adv. Nucl. Sci. Technol. 12, 1–31. http://dx.doi.org/10.1007/978-1-4613-9916-2_1.

Cacuci, D.G., 2010. Handbook of Nuclear Engineering. Springer, http://dx.doi.org/10.1007/978-0-387-98149-9.

Chen, L., Bu, N., Jin, Y., Xu, G., Li, B., 2022. Construction of IGA-suitable volume parametric models by the segmentation–mapping–merging mechanism of design features. Comput.-Aided Des. 146, 103228. http://dx.doi.org/10.1016/j.cad.2022.103228.

de Oliveira, C.R.E., 1986. An arbitrary geometry finite element method for multigroup neutron transport with anisotropic scattering. Prog. Nucl. Energy 18 (1), 227–236. http://dx.doi.org/10.1016/0149-1970(86)90029-6, Finite Element and Allied Methods.

Dokken, T., Lyche, T., Pettersen, K.F., 2013. Polynomial splines over locally refined box-partitions. Comput. Aided Geom. Design 30 (3), 331–356. http://dx.doi.org/10.1016/j.cagd.2012.12.005.

Erlich, R., Hurwitz, Jr., H., Stehn, J.R., 1947. A Multi-Group Method for Computing Critical Masses of Intermediate Piles. Technical Report A-4315; GE-RE-1, General Electric Co., Schenectady, NY, URL: https://www.osti.gov/servlets/purl/4387012.

Ferguson, J., 2022. Virtual Element Methods for the Neutron Diffusion and Neutron Transport Equations with Applicationsin Nuclear Reactor Physics (Ph.D. thesis). Imperial College London.

Ferguson, J.A., Kópházi, J., Eaton, M.D., 2021. Virtual element methods for the spatial discretisation of the multigroup neutron diffusion equation on polygonal meshes with applications to nuclear reactor physics. Ann. Nucl. Energy 151, 107884. http://dx.doi.org/10.1016/j.anucene.2020.107884.

Ferguson, J.A., Kophazi, J., Eaton, M.D., 2022. NURBS enhanced virtual element methods for the spatial discretization of the multigroup neutron diffusion equation on curvilinear polygonal meshes. J. Comput. Theor. Transp. 51 (4), 145–204. http://dx.doi.org/10.1080/23324309.2022.2103150, arXiv:https://doi.org/10.1080/23324309.2022.2103150.

Fermi, E., 1936. Sul moto dei neutroni nelle sostanze idrogenate. Ric. Sci. 7 (2), 13–52.

Feynman, R.P., Welton, T.A., Ashkin, J., Ehrlich, R., Peshkin, M., Reines, F., 1947. The Calculation of Critical Masses Including the Effects of the Distribution of Neutron Energies. Technical Report LA-524, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), http://dx.doi.org/10.2172/4387012.

Gaston, D.R., Permann, C.J., Peterson, J.W., Slaughter, A.E., Andrš, D., Wang, Y., Short, M.P., Perez, D.M., Tonks, M.R., Ortensi, J., et al., 2015. Physics-based multiscale coupling for full core nuclear reactor simulation. Ann. Nucl. Energy 84, 45–54. http://dx.doi.org/10.1016/j.anucene.2014.09.060.

Ge, J., Zhang, D., Tian, W., Wang, K., Qiu, S., Su, G., 2015. Steady and transient solutions of neutronics problems based on finite volume method (FVM) with a CFD code. Prog. Nucl. Energy 85, 366–374. http://dx.doi.org/10.1016/j.pnucene.2015.07.012.

Gordon, W.J., Hall, C.A., 1973. Construction of curvilinear co-ordinate systems and applications to mesh generation. Internat. J. Numer. Methods Engrg. 7, 461–477. http://dx.doi.org/10.1002/nme.1620070405.

Gould, C.R., Sharapov, E.I., 2024. Gian-Carlo Wick and neutron physics in the 1930s. Eur. Phys. J. H 49 (9), http://dx.doi.org/10.1140/epjh/s13129-024-00072-9.

Habetler, G.J., Martino, M.A., 1958. The Multigroup Diffusion Equations of Reactor Physics. Technical Report KAPL-1886, Knolls Atomic Power Lab.(KAPL), Niskayuna, NY (United States), http://dx.doi.org/10.2172/4310968.

Hales, J., Tonks, M., Gleicher, F., Spencer, B., Novascone, S., Williamson, R., Pastore, G., Perez, D., 2015. Advanced multiphysics coupling for LWR fuel performance analysis. Ann. Nucl. Energy 84, 98–110. http://dx.doi.org/10.1016/j.anucene.2014.11.003.

Hall, S.K., Eaton, M.D., Williams, M.M.R., 2012. The application of isogeometric analysis to the neutron diffusion equation for a pincell problem with an analytic benchmark. Ann. Nucl. Energy 49, 160–169. http://dx.doi.org/10.1016/j.anucene.2012.05.030.

Halpern, O., Lueneburg, R., Clark, O., 1938. On multiple scattering of neutrons I. Theory of the albedo of a plane boundary. Phys. Rev. 53 (2), 173. http://dx.doi.org/10.1103/PhysRev.53.173.

Hansen, K., Kang, C., 1975. Finite element methods in reactor physics analysis. In: Advances in Nuclear Science and Technology. Elsevier, pp. 173–253. http://dx.doi.org/10.1016/B978-0-12-029308-7.50012-9.

Hébert, A., Benoist, P., 1991. A consistent technique for the global homogenization of a pressurized water reactor assembly. Nucl. Sci. Eng. 109 (4), 360–372. http://dx.doi.org/10.13182/NSE109-360.

Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput. Methods Appl. Mech. Engrg. 194 (39), 4135–4195. http://dx.doi.org/10.1016/j.cma.2004.10.008.

Hughes, T.J., Engel, G., Mazzei, L., Larson, M.G., 2000. The continuous galerkin method is locally conservative. J. Comput. Phys. 163 (2), 467–488. http://dx.doi.org/10.1006/jcph.2000.6577.

Hurwitz, Jr., H., Ehrlich, R., 1948. Further Developments Of Multi-group Method For Intermediate Piles. Technical Report KAPL-39, Knolls Atomic Power Lab., Schenectady, NY, URL: https://www.osti.gov/biblio/4374393.

Hurwitz, Jr., H., Ehrlich, R., 1953. Multigroup Methods for Neutron Diffusion Problems. Technical Report AECD-3595, US Atomic Energy Commission. Technical Information Service, URL: https://www.osti.gov/biblio/4418873.

International Organization for Standardization (ISO), T.I.E.C.I., 2023. Programming Languages - Fortran - Part 1: Base Language. Technical Report 1539-1, ISO/IEC, URL: https://www.iso.org/standard/82170.html.

Kang, C.M., Hansen, K.F., 1973. Finite element methods for reactor analysis. Nucl. Sci. Eng. 51 (4), 456–495. http://dx.doi.org/10.13182/NSE73-A23278, arXiv:https://doi.org/10.13182/NSE73-A23278.

Kim, H.-J., Seo, Y.-D., Youn, S.-K., 2009. Isogeometric analysis for trimmed CAD surfaces. Comput. Methods Appl. Mech. Engrg. 198 (37), 2982–2995. http://dx.doi.org/10.1016/j.cma.2009.05.004.

Kirilov, K., Peiró, J., Green, M., Moxey, D., Beirão da Veiga, L., Dassi, F., Russo, A., 2024. Curvilinear mesh generation for the high-order virtual element method (VEM). In: Ruiz-Gironés, E., Sevilla, R., Moxey, D. (Eds.), SIAM International Meshing Roundtable 2023. Springer International Publishing AG, Cham, pp. 419–439. http://dx.doi.org/10.1007/978-3-031-40594-5_19.

Latimer, C., Kópházi, J., Eaton, M.D., McClarren, R.G., 2020a. A geometry conforming isogeometric method for the self-adjoint angular flux (SAAF) form of the neutron transport equation with a discrete ordinate (S_N) angular discretisation. Ann. Nucl. Energy 136, 107049. http://dx.doi.org/10.1016/j.anucene.2019.107049.

Latimer, C., Kópházi, J., Eaton, M.D., McClarren, R.G., 2020b. A geometry conforming, isogeometric, weighted least squares (WLS) method for the neutron transport equation with discrete ordinate (S_N) angular discretisation. Prog. Nucl. Energy 121, 103238. http://dx.doi.org/10.1016/j.pnucene.2019.103238.

Latimer, C., Kópházi, J., Eaton, M.D., McClarren, R.G., 2021. Spatial adaptivity of the SAAF and Weighted Least Squares (WLS) forms of the neutron transport equation using constraint based, locally refined, isogeometric analysis (IGA) with dual weighted residual (DWR) error measures. J. Comput. Phys. 426, 109941. http://dx.doi.org/10.1016/j.jcp.2020.109941.

Lawrence, R.D., 1986. Progress in nodal methods for the solution of the neutron diffusion and transport equations. Prog. Nucl. Energy 17 (3), 271–301. http://dx.doi.org/10.1016/0149-1970(86)90034-X.

Leslie, M., 2023. First supercomputer breaks exascale barrier, with more expected soon. Eng. 23, 10–12. http://dx.doi.org/10.1016/j.eng.2023.02.004.

Lewis, E.E., Miller, W.F., 1984. Computational Methods of Neutron Transport. American Nuclear Society.

Lewis, E.E., Smith, M.A., Tsoulfanidis, N., Palmiotti, G., Taiwo, T.A., Blomquist, R.N., 2001. Benchmark Specification for Deterministic 2-D/3-D MOX Fuel Assembly Transport Calculations Without Spatial Homogenisation (C5G7 MOX). Technical Report NEA/NSC/DOC(2001)4, Nuclear Energy Authority, URL: https://www.oecd-nea.org/science/wprs/eg3drtb/NEA-C5G7MOX.PDF.

Li, X., Sederberg, T.W., 2019. S-splines: a simple surface solution for IGA and CAD. Comput. Methods Appl. Mech. Engrg. 350, 664–678. http://dx.doi.org/10.1016/j.cma.2019.03.035.

Löhner, R., 2010. Cache-efficient renumbering for vectorization. Int. J. Numer. Methods Biomed. Eng. 26 (5), 628–636. http://dx.doi.org/10.1002/cnm.1160.

Löhner, R., Galle, M., 2002. Minimization of indirect addressing for edge-based field solvers. Commun. Numer. Methods Eng. 18 (5), 335–343. http://dx.doi.org/10.1002/cnm.494.

Lu, X., Li, Y., Chen, D., Chu, G., Wang, A., 2023. Challenges of high-fidelity virtual reactor for exascale computing and research progress of China virtual reactor. Nucl. Eng. Des. 413, 112566. http://dx.doi.org/10.1016/j.nucengdes.2023.112566.

Marchandise, E., Piret, C., Remacle, J.-F., 2012. CAD and mesh repair with radial basis functions. J. Comput. Phys. 231 (5), 2376–2387. http://dx.doi.org/10.1016/j.jcp.2011.11.033.

Martin, R.C., Knauer, J.B., Balo, P.A., 2000. Production, distribution and applications of californium-252 neutron sources. Appl. Radiat. Isot. 53 (4), 785–792. http://dx.doi.org/10.1016/S0969-8043(00)00214-1.

Martineau, R., Andrs, D., Carlsen, R., Gaston, D., Hansel, J., Kong, F., Lindsay, A., Permann, C., Slaughter, A., Merzari, E., et al., 2020. Multiphysics for nuclear energy applications using a cohesive computational framework. Nucl. Eng. Des. 367, 110751. http://dx.doi.org/10.1016/j.nucengdes.2020.110751.

Mengyan, H., Xueyan, Z., Cuiting, P., Yixuan, Z., Jun, Y., 2024. Current status of digital twin architecture and application in nuclear energy field. Ann. Nucl. Energy 202, 110491. http://dx.doi.org/10.1016/j.anucene.2024.110491.

Montanari, M., Santi, G.M., Sevilla, R., Alfredo, L., Petrinic, N., 2024. NURBS-enhanced finite element method (NEFEM) on quadrilateral meshes. Finite Elem. Anal. Des. 231, 104099. http://dx.doi.org/10.1016/j.finel.2023.104099.

Moxey, D., Cantwell, C.D., Kirby, R.M., Sherwin, S.J., 2016. Optimising the performance of the spectral/hp element method with collective linear algebra operations. Comput. Methods Appl. Mech. Engrg. 310, 628–645. http://dx.doi.org/10.1016/j.cma.2016.07.001.

Mund, E.H., 2011. Spectral element solutions for the P_N neutron transport equations. Comput. & Fluids 43 (1), 102–106. http://dx.doi.org/10.1016/j.compfluid.2010.06.010.

Nelson, A.T., 2023. Prospects for additive manufacturing of nuclear fuel forms. Prog. Nucl. Energy 155, 104493. http://dx.doi.org/10.1016/j.pnucene.2022.104493.

Nguyen, V.P., Anitescu, C., Bordas, S.P.A., Rabczuk, T., 2015. Isogeometric analysis: An overview and computer implementation aspects. Math. Comput. Simulation 117, 89–116. http://dx.doi.org/10.1016/j.matcom.2015.05.008.

Okajima, S., Kugo, T., Mori, T., 2024. In: Oka, Y., Madarame, H., Uesaka, M. (Eds.), Nuclear Reactor Physics. Springer, http://dx.doi.org/10.1007/978-4-431-55600-8.

Ornstein, L.S., Uhlenbeck, G.E., 1937. Some kinetic problems regarding the motion of neutrons through paraffine. Phys. 4 (6), 478–486. http://dx.doi.org/10.1016/S0031-8914(37)80080-4, First derivation of the NTE.

Otoguro, Y., Takizawa, K., Tezduyar, T., 2018. A General-Purpose NURBS Mesh Generation Method for Complex Geometries. Springer Nature Switzerland AG, pp. 399–434. http://dx.doi.org/10.1007/978-3-319-96469-0_10.

Owens, A.R., Kópházi, J., Welch, J.A., Eaton, M.D., 2017a. Energy dependent mesh adaptivity of discontinuous isogeometric discrete ordinate methods with dual weighted residual error estimators. J. Comput. Phys. 335, 352–386. http://dx.doi.org/10.1016/j.jcp.2017.01.035, URL: https://www.sciencedirect.com/science/article/pii/S0021999117300517.

Owens, A.R., Welch, J.A., Kópházi, J., Eaton, M.D., 2016. Discontinuous isogeometric analysis methods for the first-order form of the neutron transport equation with discrete ordinate (S_N) angular discretisation. J. Comput. Phys. 315, 501–535. http://dx.doi.org/10.1016/j.jcp.2016.03.060.

Owens, A.R., Welch, J.A., Kópházi, J., Eaton, M.D., 2017b. An adaptive, hanging-node, discontinuous isogeometric analysis method for the first-order form of the neutron transport equation with discrete ordinate (S_N) angular discretisation. Comput. Methods Appl. Mech. Engrg. 318, 215–241. http://dx.doi.org/10.1016/j.cma.2017.01.036.

Prince, Z.M., Hanophy, J.T., Labouré, V.M., Wang, Y., Harbour, L.H., Choi, N., 2024. Neutron transport methods for multiphysics heterogeneous reactor core simulation in Griffin. Ann. Nucl. Energy 200, 110365. http://dx.doi.org/10.1016/j.anucene.2024.110365.

Quammen, C., 2015. Scientific data analysis and visualization with Python, VTK, and ParaView. In: Proceedings of the 14th Python in Science Conference. SCIPY 2015, URL: https://api.semanticscholar.org/CorpusID:5631793.

Sanchez, R., 2009. Assembly homogenization techniques for core calculations. Prog. Nucl. Energy 51 (1), 14–31. http://dx.doi.org/10.1016/j.pnucene.2008.01.009.

Schmidt, R., Belcourt, K., Hooper, R., Pawlowski, R., Clarno, K., Simunovic, S., Slattery, S., Turner, J., Palmtag, S., 2015. An approach for coupled-code multiphysics core simulations from a common input. Ann. Nucl. Energy 84, 140–152. http://dx.doi.org/10.1016/j.anucene.2014.11.015.

Semenza, L.A., Lewis, E.E., Rossow, E.C., 1972. The application of the finite element method to the multigroup neutron diffusion equation. Nucl. Sci. Eng. 47 (3), 302–310. http://dx.doi.org/10.13182/NSE72-A22416.

Sevilla, R., Fernández-Méndez, S., 2011. Numerical integration over 2D NURBS-shaped domains with applications to NURBS-enhanced FEM. Finite Elem. Anal. Des. 47 (10), 1209–1220. http://dx.doi.org/10.1016/j.finel.2011.05.011.

Sevilla, R., Fernández-Méndez, S., Huerta, A., 2008. NURBS-enhanced finite element method (NEFEM) for Euler equations. Internat. J. Numer. Methods Fluids 57, 1051–1069. http://dx.doi.org/10.1002/fld.1711.

Sevilla, R., Fernandez-Mendez, S., Huerta, A., 2011a. 3D NURBS-enhanced finite element method (NEFEM). Internat. J. Numer. Methods Engrg. 88, 103–125. http://dx.doi.org/10.1002/nme.3164.

Sevilla, R., Fernández-Méndez, S., Huerta, A., 2011b. NURBS-enhanced finite element method (NEFEM). Arch. Comput. Methods Eng. 18 (4), 441–484. http://dx.doi.org/10.1007/s11831-011-9066-5.

Smith, K.S., 1986. Assembly homogenization techniques for light water reactor analysis. Prog. Nucl. Energy 17 (3), 303–335. http://dx.doi.org/10.1016/0149-1970(86)90035-1.

Smith, K.S., 2017. Nodal diffusion methods and lattice physics data in LWR analyses: Understanding numerous subtle details. Prog. Nucl. Energy 101, 360–369. http://dx.doi.org/10.1016/j.pnucene.2017.06.013.

Sobes, V., Hiscox, B., Popov, E., Archibald, R., Hauck, C., Betzler, B., Terrani, K., 2021. AI-based design of a nuclear reactor core. Sci. Rep. 11 (1), 19646. http://dx.doi.org/10.1038/s41598-021-98037-1.

Thomas, D.C., Engvall, L., Schmidt, S.K., Tew, K., Scott, M.A., 2022. U-splines: Splines over unstructured meshes. Comput. Methods Appl. Mech. Engrg. 401, 115515. http://dx.doi.org/10.1016/j.cma.2022.115515.

Tomov, S., Dongarra, J., Baboulin, M., 2010. Towards dense linear algebra for hybrid GPU accelerated manycore systems. Parallel Comput. 36 (5–6), 232–240. http://dx.doi.org/10.1016/j.parco.2009.12.005.

Turcksin, B., Ragusa, J.C., 2014. Discontinuous diffusion synthetic acceleration for S_N transport on 2D arbitrary polygonal meshes. J. Comput. Phys. 274, 356–369. http://dx.doi.org/10.1016/j.jcp.2014.05.044.

Wachspress, E.L., 1992. Historical Review of Group Diffusion Computation. Trans. Am. Nucl. Soc. 66, 240–241, https://www.osti.gov/biblio/6708585.

Wang, D., Zhu, Z., 2021. A revisit to CMFD schemes: Fourier analysis and enhancement. Energies 14 (2), 424. http://dx.doi.org/10.3390/en14020424.

Welch, J.A., Kópházi, J., Owens, A.R., Eaton, M.D., 2017a. A geometry preserving, conservative, mesh-to-mesh isogeometric interpolation algorithm for spatial adaptivity of the multigroup, second-order even-parity form of the neutron transport equation. J. Comput. Phys. 347, 129–146. http://dx.doi.org/10.1016/j.jcp.2017.06.015.

Welch, J.A., Kópházi, J., Owens, A.R., Eaton, M.D., 2017b. Isogeometric analysis for the multigroup neutron diffusion equation with applications in reactor physics. Ann. Nucl. Energy 101, 465–480. http://dx.doi.org/10.1016/j.anucene.2016.11.015.

Wick, G., 1936. Sulla diffusione dei neutroni lenti. Atti della Accademia Nazionale dei Lincei Rendiconti 23, 774–782.

Wilson, S.G., Eaton, M.D., Kópházi, J., 2024a. Energy-dependent, self-adaptive mesh h(p)-refinement of a constraint-based continuous bubnov-Galerkin isogeometric analysis spatial discretization of the multi-group neutron diffusion equation with dual-weighted residual error measures. J. Comput. Theor. Transp. 53 (2), 89–152. http://dx.doi.org/10.1080/23324309.2024.2313460.

Wilson, S.G., Eaton, M.D., Kópházi, J., 2024b. Energy-dependent, self-adaptive mesh h(p)-refinement of an interior-penalty scheme for a discontinuous Galerkin isogeometric analysis spatial discretization of the multi-group neutron diffusion equation with dual-weighted residual error measures. J. Comput. Theor. Transp. 53 (4), 223–278. http://dx.doi.org/10.1080/23324309.2024.2334277.

Wilson, S.G., Eaton, M.D., Kópházi, J., 2024c. A symmetric interior-penalty discontinuous Galerkin isogeometric analysis spatial discretization of the self-adjoint angular flux form of the neutron transport equation. Comput. Methods Appl. Mech. Engrg. 432, 117414. http://dx.doi.org/10.1016/j.cma.2024.117414.

Wilson, W.B., Perry, R.T., Charlton, W.S., Parish, T.A., 2009. Sources: A code for calculating (alpha, n), spontaneous fission, and delayed neutron sources and spectra. Prog. Nucl. Energy 51 (4), 608–613. http://dx.doi.org/10.1016/j.pnucene.2008.11.007.

Wood, J., Williams, M.M.R., 1984. Recent progress in the application of the finite element method to the neutron transport equation. Prog. Nucl. Energy 14 (1), 21–40. http://dx.doi.org/10.1016/0149-1970(84)90010-6.

Xu, X., 2021. Implicit discrete ordinates discontinuous Galerkin method for radiation problems on shared-memory multicore CPU/many-core GPU computation architecture. Numer. Heat Transfer B 79 (4), 165–188. http://dx.doi.org/10.1080/10407790.2020.1819708.

Zou, X., Lo, S.B., Sevilla, R., Hassan, O., Morgan, K., 2024a. The generation of 3D Surface Meshes for NURBS-Enhanced FEM. Comput.-Aided Des. 168, 103653. http://dx.doi.org/10.1016/j.cad.2023.103653.

Zou, X., Lo, S.B., Sevilla, R., Hassan, O., Morgan, K., 2024b. The generation of tetrahedral meshes for NURBS-enhanced FEM. Eng. Comput. 40 (6), 3949–3977. http://dx.doi.org/10.1007/s00366-024-02004-z.