1 Implementation in MFEM

The open source finite element library Modular Finite Elements Method (MFEM[1]) [?] was used to create the system of linear equations to be solved by a linear algebra solver. The user chooses various parameters to create the system of equations and passes them into MFEM as arguments (i.e. the order of finite elements, the mesh, the number of times to refine the mesh, the order of the mesh, any mesh transformations, the linear algebra solver method, the source iteration convergence criteria, the maximum number of source iterations to perform, the initial guess for the scalar flux, and, in diffusion limit problems, the scaling factor to be applied).

MFEM creates the matrices and a linear solver computes the angular flux. This research utilizes the serial version of MFEM (as opposed to the parallel version where the spatial domain is solved in parallel) and the direct solver UMFPack[2] [?, ?] to solve the equations using a LU decomposition. It is common for transport solvers to solve the local system of equations for an individual spatial cell and sweep through the problem domain, propagating information from one cell to the next. Instead, we use MFEM to create the system of equations for the entire problem domain and solve for all of the unknowns in all cells simultaneously. This is more computationally intensive because all of the degrees of freedom in the entire problem domain must be solved for simultaneously. However, mesh zone complications, such as cycles, may be present in the mesh. We avoid having to "break the cycles" by solving for the entire problem simultaneously.

1.1 Transport Operators

Shown in Table 2 are the functions within MFEM that integrate and assemble the various components of the transport equation (Equation **??**) to the linear algebraic system. The functions are displayed along with the general form of their equation and their translation to the applicable component of the transport equation. The last two entries of Table 2 are the interior boundaries using the upstream values (no. 5) and the problem boundary (no. 6). Several of the MFEM equations have coefficients, α and β, that are required input. For item number 1, using $\alpha = 1$ sets the MFEM equation equal to the discretized equation. Similarly, for item number 5 using $\alpha = -1$ and $\beta = 1/2$,

$$\alpha \int_{\partial \mathbb{V}} \boldsymbol{\Omega} \cdot (-\hat{n}) \, \{\psi\} \, w + \beta \int_{\partial \mathbb{V}} |\boldsymbol{\Omega} \cdot (-\hat{n})| \, [\![\psi]\!] \, w = \int_{\partial \mathbb{V}_k} (\boldsymbol{\Omega} \cdot \hat{n}) \, \psi_k \; w_{k,i} \tag{1}$$

[1] `mfem.org`

[2] `http://faculty.cse.tamu.edu/davis/suitesparse.html`

where $\{\psi\} = 1/2\,(\psi_u + \psi_k)$ and $[\![\psi]\!] = \psi_u - \psi_k$, where ψ_u is the upwind angular flux and ψ_k is the angular flux in cell k.

$$-1 \int_{\partial\mathbb{V}} [\mathbf{\Omega} \cdot (-\hat{n})] \left[\frac{1}{2}\,(\psi_u + \psi_k)\right] w + \frac{1}{2} \int_{\partial\mathbb{V}} |\mathbf{\Omega} \cdot (-\hat{n})|\,(\psi_u - \psi_k)\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w_{k,i} \tag{2}$$

$$\frac{1}{2} \int_{\partial\mathbb{V}} (\mathbf{\Omega} \cdot \hat{n})\,(\psi_u + \psi_k)\,w + \frac{1}{2} \int_{\partial\mathbb{V}} |\mathbf{\Omega} \cdot (-\hat{n})|\,(\psi_u - \psi_k)\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w_{k,i} \tag{3}$$

For $\mathbf{\Omega} \cdot \hat{n} < 0$ (incident to cell k),

$$\frac{1}{2} \int_{\partial\mathbb{V}} (\mathbf{\Omega} \cdot \hat{n})\,(\psi_u + \psi_k)\,w - \frac{1}{2} \int_{\partial\mathbb{V}} (\mathbf{\Omega} \cdot \hat{n})\,(\psi_u - \psi_k)\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w_{k,i} \tag{4}$$

$$\frac{1}{2} \int_{\partial\mathbb{V}} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w + \frac{1}{2} \int_{\partial\mathbb{V}} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_k\ w_{k,i} \tag{5}$$

The normal vector $\hat{n}$ in MFEM is outward of the upwind mesh surface so a negative was applied to the normal vector to convert it to be the outward normal of the surface of cell k like it has been previously defined in this thesis. Similarly, for item number 6, $\alpha = -1$ and $\beta = -1/2$,

$$\frac{\alpha}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\ \mathbf{\Omega} \cdot (-\hat{n})\ \ w - \beta \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,|\mathbf{\Omega} \cdot (-\hat{n})|\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_{\text{inc},k}\ w_{k,i} \tag{6}$$

$$-\frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\ \mathbf{\Omega} \cdot (-\hat{n})\ \ w + \frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,|\mathbf{\Omega} \cdot (-\hat{n})|\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_{\text{inc},k}\ w_{k,i} \tag{7}$$

$$\frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,(\mathbf{\Omega} \cdot \hat{n})\,w + \frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,[\mathbf{\Omega} \cdot (-\hat{n})]\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_{\text{inc},k}\ w_{k,i} \tag{8}$$

$$\frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,(\mathbf{\Omega} \cdot \hat{n})\,w + \frac{1}{2} \int_{\partial\mathbb{V}} \psi_{\text{inc}}\,(\mathbf{\Omega} \cdot \hat{n})\,w = \int_{\partial\mathbb{V}_k} (\mathbf{\Omega} \cdot \hat{n})\,\psi_{\text{inc},k}\ w_{k,i} \tag{9}$$

MFEM automatically determines the degree of numerical integration to integrate each of the integrals of Table 2. These default integration orders are shown in Table 3. It was discovered that integrating all of the terms consistently was important for numeric conservation. For simplicity, each of the integration orders were set to the largest of the default integration orders. Table 3 shows that the integration order is the same for all of the integrators except DomainLFIntegrator, which is the largest integration order only if $p = 0$ (piecewise constant). While the results presented in this thesis do not consider the circumstance of piecewise constant finite elements, this integration order was included in the code for future use.

MFEM is equipped to visualize data using various tools requiring additional user input. The images presented in this thesis were produced with VisIt, an open source visualization analysis tool [?].

no.	Discretized Equation	MFEM Equation	MFEM Integrator Function
1	$(\boldsymbol{\Omega} \cdot \boldsymbol{\nabla} \psi_j, v_i)_{\mathbb{D}_k}$	$(\alpha\ \boldsymbol{\Omega} \cdot \boldsymbol{\nabla} \psi, v)_{\mathbb{D}_k}$	ConvectionIntegrator($\boldsymbol{\Omega}, \alpha$)
2	$(\sigma_t\ \psi_j, v_i)_{\mathbb{D}_k}$	$(\sigma_t\ \psi, v)_{\mathbb{D}_k}$	MassIntegrator(σ_t)
3	$(\sigma_s\ \phi, v_i)_{\mathbb{D}_k}$	$(\varphi, v)_{\mathbb{D}_k}$	DomainLFIntegrator(φ)
4	$(S_0, v_i)_{\mathbb{D}_k}$	$(S_0, v)_{\mathbb{D}_k}$	DomainLFIntegrator(S_0)
5	$(\boldsymbol{\Omega} \cdot \hat{n}\ \psi_j, v_i)_{\partial\mathbb{D}_k}$	$\alpha\,(\boldsymbol{\Omega} \cdot \hat{n}\ \psi, v)_{\partial\mathbb{D}_k}$ $+\beta\,(\lvert\boldsymbol{\Omega} \cdot \hat{n}\rvert\ \psi, v)_{\partial\mathbb{D}_k}$	DGTraceIntegrator($\boldsymbol{\Omega}, \alpha, \beta$)
6	$(\boldsymbol{\Omega} \cdot \hat{n}\ \psi_{\text{inc}}, v_i)_{\partial\mathbb{D}_k}$	$\frac{\alpha}{2}\,(\psi_{inc}\ \boldsymbol{\Omega} \cdot \hat{n}, v)_{\partial\mathbb{D}_k}$ $-\beta\,(\psi_{inc}\,\lvert\boldsymbol{\Omega} \cdot \hat{n}\rvert\,, v)_{\partial\mathbb{D}_k}$	BoundaryFlowIntegrator($\psi_{inc}, \boldsymbol{\Omega}, \alpha, \beta$)

Table 1: MFEM PDE function calls where the arguments have been dropped (see Equations **??** and **??** for these details).

no.	Discretized Equation	MFEM Equation	MFEM Integrator Function
	$(r\ \mathbf{\Omega} \cdot \mathbf{\nabla} \psi_j, v_i)_{\mathbb{D}_k}$	$(r\ \alpha\ \mathbf{\Omega} \cdot \mathbf{\nabla} \psi, v)_{\mathbb{D}_k}$	RZConvectionIntegrator($\mathbf{\Omega}, \alpha$) †
	$(r\ \mathbf{\Omega} \cdot \hat{n}\ \psi_j, v_i)_{\partial\mathbb{D}_k}$	$\alpha\,(r\ \mathbf{\Omega} \cdot \hat{n}\ \psi, v)_{\partial\mathbb{D}_k}$ $+\beta\,(r\,\lvert\mathbf{\Omega} \cdot \hat{n}\rvert\,\psi, v)_{\partial\mathbb{D}_k}$	RZDGTraceIntegrator($\mathbf{\Omega}, \alpha, \beta$) †
	$(\mu_{n,m}\psi_j, v_i)_{\mathbb{D}_k}$		MassIntegrator($\mu_{n,m}$)
	$\left(\frac{\alpha_{m+1/2}n}{\tau_{n,m}w_{n,m}}\psi_j, v_i\right)_{\mathbb{D}_k}$		MassIntegrator($\frac{\alpha_{m+1/2}n}{\tau_{n,m}w_{n,m}}$)
	$(r\ \sigma_t\ \psi_j, v_i)_{\mathbb{D}_k}$	$(r\ \sigma_t\ \psi, v)_{\mathbb{D}_k}$	RZMassIntegrator(σ_t) †
	$\frac{1}{4\pi}\,(r\ \sigma_s\ \phi, v_i)_{\mathbb{D}_k}$	$(r\ \varphi, v)_{\mathbb{D}_k}$	RZDomainLFIntegrator$\left(\frac{1}{4\pi}\sigma_s\ \phi\right)$ †
	$\frac{1}{4\pi}\,(r\ S_0, v_i)_{\mathbb{D}_k}$	$(r\ S_0, v)_{\mathbb{D}_k}$	RZDomainLFIntegrator$\left(\frac{S_0}{4\pi}\right)$ †
	$(r\ \mathbf{\Omega} \cdot \hat{n}\ \psi_{\text{inc}}, v_i)_{\partial\mathbb{D}_k}$	$\frac{\alpha}{2}\,(r\ \psi_{inc}\ \mathbf{\Omega} \cdot \hat{n}, v)_{\partial\mathbb{D}_k}$ $-\beta\,(r\ \psi_{inc}\,\lvert\mathbf{\Omega} \cdot \hat{n}\rvert\,, v)_{\partial\mathbb{D}_k}$	RZBoundaryFlowIntegrator($\psi_{inc}, \mathbf{\Omega}, \alpha, \beta$) †

† Modified MFEM operator

Table 2: MFEM PDE function calls where the arguments have been dropped (see Equations **??** and **??** for these details).

Table 3: MFEM default integration orders for transport operators. The notation for the finite element order is p, mesh order is m, and problem dimension is d.

MFEM Integrator	**Default Integration Order**
DGTraceIntegrator	$m \cdot d + 2 \cdot p - 1$
ConvectionIntegrator	$m \cdot d + 2 \cdot p - 1$
MassIntegrator	$m \cdot d + 2 \cdot p - 1$
DomainLFIntegrator	$2 \cdot m$
BoundaryFlowIntegrator	$m \cdot d + 2 \cdot p - 1$

1.1.1 MIP DSA Operators

There are some specific function calls to MFEM for the diffusion equation that are listed in Table 4. Items 3 and 4 have a σ_D value that controls the DG method to be used, where $\sigma_D = -1$ is for the symmetric interior penalty method. Item 6 is for the Robin boundary condition described by Equation **??**. Item 4 is the function that will need to be modified to adapt Methods 1 and 2 (this function is not used for Method 3).

no.	FEM Equations **??** & **??**	MFEM Equation	User Input
1	$(\sigma_a \phi, w)_{\mathbb{V}}$	$(\sigma_a \phi, w)_{\mathbb{V}}$	MassIntegrator(σ_a)
2	$(D\boldsymbol{\nabla}\phi, \boldsymbol{\nabla}w)_{\mathbb{V}}$	$(D\boldsymbol{\nabla}\phi, \boldsymbol{\nabla}w)_{\mathbb{V}}$	DiffusionIntegrator(D)
3	$(\{\!\{D\partial_n \phi\}\!\}, [\![w]\!])_{\partial\mathbb{V}^i}$ $+ ([\![\phi]\!], \{\!\{D\partial_n w\}\!\})_{\partial\mathbb{V}^i}$ $+ (\kappa_e [\![\phi]\!], [\![w]\!])_{\partial\mathbb{V}^i}$	$(\{\!\{D\boldsymbol{\nabla}\phi \cdot \hat{n}\}\!\}, [\![w]\!])_{\partial\mathbb{V}^i}$ $+\sigma_D ([\![\phi]\!], \{\!\{D\boldsymbol{\nabla}w \cdot \hat{n}\}\!\})_{\partial\mathbb{V}^i}$ $+\kappa \left(\left\{\!\!\left\{\frac{D}{h_\perp}\right\}\!\!\right\} [\![\phi]\!], [\![w]\!]\right)_{\partial\mathbb{V}^i}$	DGDiffusionIntegrator(D, σ_D, κ)
4	$(\{\!\{D\partial_n \phi\}\!\}, [\![w]\!])_{\partial\mathbb{V}^d}$ $-\frac{1}{2} ([\![\phi]\!], \{\!\{D\partial_n w\}\!\})_{\partial\mathbb{V}^d}$ $-\frac{1}{2} (\kappa_e [\![\phi]\!], [\![w]\!])_{\partial\mathbb{V}^d}$	$(\{\!\{D\boldsymbol{\nabla}\phi \cdot \hat{n}\}\!\}, [\![w]\!])_{\partial\mathbb{V}^d}$ $+\sigma_D ([\![\phi]\!], \{\!\{D\boldsymbol{\nabla}w \cdot \hat{n}\}\!\})_{\partial\mathbb{V}^d}$ $+\kappa \left(\left\{\!\!\left\{\frac{D}{h_\perp}\right\}\!\!\right\} [\![\phi]\!], [\![w]\!]\right)_{\partial\mathbb{V}^d}$	DGDiffusionIntegrator(D, σ_D, κ)
5	$(Q_0, w)_{\mathbb{V}}$	$(Q_0, w)_{\mathbb{V}}$	DomainLFIntegrator(Q_0)
6	$\left(\frac{1}{2}\phi, w\right)_{\partial\mathbb{V}^d}$	$\left(\frac{1}{2}\phi, w\right)_{\partial\mathbb{V}^d}$	BoundaryMassIntegrator$\left(\frac{1}{2}\right)$

† $\partial\mathbb{V}^i$ denotes an internal edge
‡ $\partial\mathbb{V}^d$ denotes a boundary edge

Table 4: MFEM diffusion equation function calls.

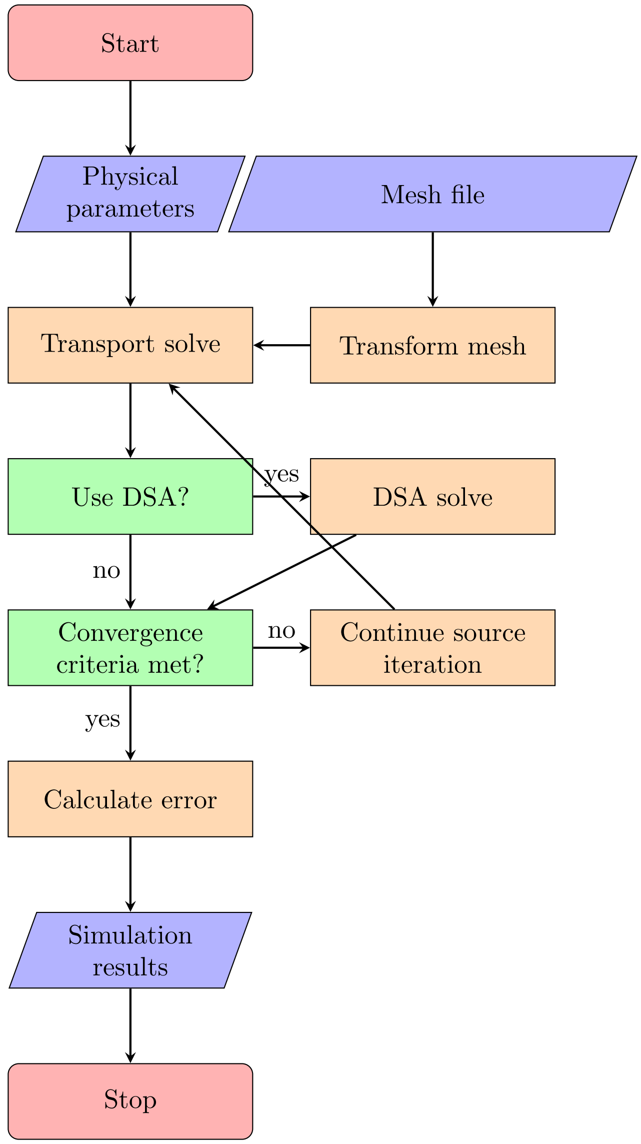

Figure 1: Flow diagram for solution process.