IMPLEMENTATION OF THE $P_N$-APPROXIMATION FOR RADIATIVE HEAT TRANSFER ON OPENFOAM

Ricardo Marquez
Michael Modest∗
Mechanical Engineering and Applied Mechanics
School of Engineering
University of California Merced
Merced, California, 95343
Email: mmodest@ucmerced.edu

ABSTRACT
This work presents an OpenFOAM implementation of the $P_N$ approximation for radiative heat transfer, including higher orders $P_3$, $P_5$, and $P_7$. Also described is a procedure which enables the sequential numerical computations of the coupled partial differential equations (PDEs) by re-expressing the boundary conditions in matrix form so that individual boundary conditions can be associated with each PDE. The implementation of the software programs are verified with derived analytical solutions for 1-D slabs with constant and variable properties, and are also tested with various orientations in order to demonstrate the geometric invariance properties of the 3-dimensional $P_N$ formulation. A few examples taken from the literature are also considered in this work and could be taken as benchmark solutions for the $P_N$ approximations.

INTRODUCTION
The radiative transfer equation (RTE) is an integro-differential equation in seven independent variables (3 space and 2 directional, temperature, and wavenumber) [1], which is exceedingly difficult to solve. As a result, approximate solution methods to the RTEs such as the spherical harmonics method (SHM), discrete ordinates method (DOM), the finite volume method (FVM), or Monte Carlo method are frequently employed to solve radiation problems. Each of these approximating methods have their well-known advantages and disadvantages. The DOM/FVM method has the advantage that it is easy to implement and can use the finite volume discretizations which are used in computational fluid dynamics (CFD) computations. Higher accuracy can be obtained using the DOM/FVM by increasing angular discretizations. The disadvantages of DOM/FVM is that results can become inaccurate when there are large aspect ratios, geometries with sharp corners, or if the extinction coefficient varies by orders of magnitude in the computational domain [2,3]. The PMC method is an exact method with statistical error that decreases with larger number of photon bundles. The PMC is often used to benchmark other RTE solvers when exact solutions are unavailable. However, PMC can be much slower than other RTE solvers.

The SHM, which is the focus of this paper, offers an approximate solution to the radiative transfer equation (RTE) by transforming the RTE to a system of elliptic PDEs. This method approximates the radiative intensity as a truncated series of spherical harmonics that decouple the directional and spatial variation of the intensity field. For optically thick cases, the lowest order of the SHM, $P_1$, can provide acceptable accuracy for the intensity field. However, for optically thin to intermediate cases, the accuracy of $P_1$ is known to diminish [1]. Despite the shortcomings of $P_1$, applications of the higher order SHM have been rather limited, due to exponentially increasing mathematical complexity.

Modest and Yang [4] formulated a methodology to reduce the $(N+1)^2$ first-order PDEs to a set of $N(N+1)/2$ second-order, elliptic PDEs. Their work also included the formulation of the $N(N+1)/2$ Marshak boundary conditions. The methodology was designed to be applicable for arbitrary three dimensional geometries. Further development of the $P_N$-approximation for isotropic scattering and its Marshak boundary conditions was recently presented in [5]. The newer formulation for isotropic scattering, reduces the complexity of the SHM formulation because many of the coefficient tensors involved can be combined to obtain all-
symmetric operators in the set of $P_N$ equations.

There are only a few literature examples with applications of high order SHM. Yang and Modest [6] demonstrated $P_3$ on a two dimensional rectangular enclosure with an isotropically scattering medium with a hot wall segment, and also a two-dimensional triangular enclosure with an absorbing-emit-ting medium with variable absorption coefficient. Ravishankar et al. [3] demonstrated $P_3$ in a steady state homogeneous combustion of a methane-air mixture under laminar flow conditions for optical thicknesses $\tau = 0.25$ and 0.5. Modest [5] applied $P_3$ to a square enclosure with an absorbing-emitting medium with strongly variable temperature and absorption coefficients. The absorption coefficients were adjusted in order to analyze optically thin, intermediate, and thick situations.

The work presented here employs the full three-dimensional $P_N$ approximation formulated in [5], where the only simplification is to neglect anisotropic scattering. The set of governing differential equations consist of $N(N+1)/2$ unknowns coupled by a set of elliptical PDEs that are complex because of several second-order mixed derivatives that are involved. These mixed derivatives make it challenging to obtain analytical solutions for simple cases in 2- and 3-D. However, for simple 1-D cases, analytical solutions are readily obtained. The 1-D problems are important as they serve to verify that the $P_N$ implementation accurately employs each of the terms that are involved. Additionally, one can verify that the resulting incident radiation profiles computed by $P_N$ are invariant with respect to coordinate rotations. The work presented here also tests 2- and 3-D examples. The 2-D example includes the sample problem provided in [5], which is further extended to demonstrate $P_7$. A version of this 2-D example is also applied to a 3-D cylinder case.

With these example problems, the purpose of this work is to
1. Outline useful procedures, such as manipulation of the boundary conditions, in order to implement the SHM in the OpenFOAM CFD software package [7],
2. verify consistency of solutions of $P_N$ solutions for geometrically invariant problems,
3. demonstrate the relative accuracy of higher order $P_N$-approximations on 1-, 2-, and 3-D examples,
4. and illustrate $P_N$ solutions for axisymmetric cases.

The procedures introduced here take advantage of the SHM formulation for isotropic scattering [5]. Similar procedures are more involved for the anisotropic scattering formulation and are not discussed here.

$P_N$ FORMULATION

Governing Differential Equations

The $P_N$ approximation is based on representing the intensity field $I(\bar{r}, s)$ as a series products of intensity coefficients functions and of spherical harmonics $Y_n^m$, whereby the spatial $I(\bar{r} = \int \beta d\bar{x}, \beta$ is the extinction coefficient) and the directional ($s$) dependencies are decoupled,

$$I(\bar{r}, s) = \sum_{n=0}^{N} \sum_{m=-n}^{n} I_n^m(\bar{r})Y_n^m(s).$$

Spherical harmonics satisfy Laplace’s equation in spherical coordinates and are defined here as,

$$Y_n^m = \begin{cases} \cos(m\phi)P_n^m(\cos \theta), & \text{for } m \geq 0, \\ \sin(m\phi)P_n^m(\cos \theta), & \text{for } m < 0, \end{cases}$$

and $P_n^m(\cos \theta)$ are associated Legendre polynomials (see, [5]).

As formulated in [4] and simplified for isotropic scattering in [5], a set of $N(N+1)/2$ second-order elliptic PDEs result after applying Eqn. (1) to the RTE with the same number of dependent variables $I_n^m$. Each PDE is associated with a set of the spherical harmonics $Y_n^m$ defined for a global coordinate system. From [5], the equations associated with each $Y_n^m : n = 0, 2, ..., N - 1, 0 \leq m \leq n$ are 1,

$$\begin{align*}
\sum_{k=1}^{3} (\mathcal{L}_{xy} - \mathcal{L}_{yy}) & \left[ (1 + \delta_{m2})a_k^{n-2} I_{n-4}^{m-2} + \frac{\delta_{m1}}{2} c_k^{n-2} I_{n-4}^{m-2} \\
+ b_k^{n+2} I_{n+4}^{m+2} \right] \\
+ (\mathcal{L}_{xz} + \mathcal{L}_{zy}) & \left[ (1 + \delta_{m1})b_k^{n-1} I_{n-4}^{m-1} + d_k^{n+1} I_{n+4}^{m+1} \right] \\
+ (\mathcal{L}_{xx} + \mathcal{L}_{yy}) & \left[ -(1 - \delta_{m2})a_k^{n-2} I_{n-4}^{m-2} - \frac{\delta_{m1}}{2} c_k^{n-2} I_{n-4}^{m-2} \right] \\
+ (\mathcal{L}_{yz} + \mathcal{L}_{xy}) & \left[ -(1 - \delta_{m1})b_k^{n-1} I_{n-4}^{m-1} + e_k^{n+1} I_{n+4}^{m+1} \right] \\
+ (\mathcal{L}_{zz} - (1 - \omega \delta_{0n})c_k) & I_n^m = -(1 - \omega)I_{0} \delta_{0n} \\
\end{align*}$$

and for each $Y_n^m : n = 0, 2, ..., N - 1, 1 \leq m \leq n :$

$$\begin{align*}
\sum_{k=1}^{3} (\mathcal{L}_{xy} + \mathcal{L}_{yy}) & \left[ (1 + \delta_{m2})a_k^{n-2} I_{n+4}^{m+2} + \frac{\delta_{m1}}{2} c_k^{n+2} I_{n+4}^{m+2} \\
- b_k^{n-2} I_{n-4}^{m-2} \right] \\
+ (\mathcal{L}_{yz} + \mathcal{L}_{xy}) & \left[ (1 + \delta_{m1})b_k^{n-1} I_{n+4}^{m+1} - d_k^{n+1} I_{n+4}^{m+1} \right] \\
+ (\mathcal{L}_{xx} - \mathcal{L}_{yy}) & \left[ (1 - \delta_{m2})a_k^{n-2} I_{n+4}^{m+2} - \frac{\delta_{m1}}{2} c_k^{n+2} I_{n+4}^{m+2} \right] \\
+ (\mathcal{L}_{yz} + \mathcal{L}_{xy}) & \left[ -(1 - \delta_{m1})b_k^{n+1} I_{n+4}^{m+1} + e_k^{n+1} I_{n+4}^{m+1} \right] \\
+ (\mathcal{L}_{xx} + \mathcal{L}_{yy}) & \left[ -(1 - \delta_{m1})b_k^{n+1} I_{n+4}^{m+1} + e_k^{n+1} I_{n+4}^{m+1} \right] \\
+ (\mathcal{L}_{zz} - (1 - \omega \delta_{0n})c_k) & I_n^m = 0 \\
\end{align*}$$

1The formulation of the SHM equations presented in [5] contained an error for $f_{s},$ which should be identically equal to 1 ($f_s = 1$). The correction was applied in this paper.
where \( \omega \) is the scattering albedo, the coefficients \( a, b, c, d, \) and \( e \) are given in [5], and \( \delta_{ij} \) is the Kronecker delta function. The \( \mathcal{L} \) operators denote mixed derivatives, e.g.,

\[
L_{xy} = \frac{1}{\beta} \frac{\partial}{\partial y} \left( \frac{1}{\beta} \frac{\partial}{\partial x} \right). \tag{4}
\]

Although the combination of operators in Eqn. (3) are not identical in \( x, y, \) and \( z \) due to the fact that the global directional angles are tied to the choice of coordinate system [5], the results for \( I_0^m \) (and \( G = 4\pi I_0^0 \)) are nevertheless invariant to rotations for geometrically similar problems, as is verified with numerical examples in Section 4.

The \( N(N+1)/2 \) equations can be solved with the OpenFOAM package by either using a sequential or a block-coupling approach [8], with the latter being more involved and under-developed in OpenFOAM. Therefore, this first implementation of the \( P_N \)-approximation focuses on the development of a sequential approach where each PDE is used to solve for a single intensity coefficient at a time. Since the above equations each contain only one dependent variable without a derivative \( I_0^m \), and because a Laplacian operator can be formed for the corresponding \( I_0^m \) variable, it is clear that the equation associated with \( Y_n^m \) should be used to solve for \( I_0^m \). When solving for each \( I_0^m \), boundary conditions must be specified for it. The next section discusses how this can be done efficiently.

**Boundary Conditions**

Consistent Marshak boundary conditions for the SHM were also developed in [5]. The ‘relevant’ boundary conditions are associated with local spherical harmonics \( Y_n^m \), with certain combinations of \( m \) and \( n \) indices. The relevant \((m,n)\) pairs are for \( m = 0, 1, \ldots, N - 1 \), and for even \( m: n = m+1, m+3, \ldots, N \); for odd \( m: n = m, m+2, \ldots, N - 1 \). These combinations of \((m,n)\) indices lead to \( N(N+1)/2 \) distinct boundary equations. For example, the boundary conditions for \( P_3 \) (where \( N(N+1)/2 = 6 \)) are associated with local spherical harmonics: \( Y_1^0, Y_1^{-1}, Y_1^1, Y_3^{-2}, \) and \( Y_3^0 \).

Letting \( i = (n+1)/2 \), the boundary conditions for isotropic scattering are [5],

\[
I_n^m p_{0,n} = \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \label{Y_n^0} p_{l,m,m'} I_{2l}^m \frac{\partial}{\partial \tau_x} + \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( v_{li}^0 \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_y} - \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( w_{li}^0 \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_z}
\]

for \( Y_n^0 \):

\[
0 = \sum_{l=1}^{N+1} \sum_{m'=-2l}^{2l} p_{l,m}^m \Delta_{l,m,m'}^2  I_{2l}^m \frac{\partial}{\partial \tau_x} + \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( v_{li}^m \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_y} - \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( w_{li}^m \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_z} \tag{5a}
\]

and for \( Y_n^m \):

\[
0 = \sum_{l=1}^{N+1} \sum_{m'=-2l}^{2l} p_{l,m}^m \Delta_{l,m,m'}^2  I_{2l}^m \frac{\partial}{\partial \tau_x} + \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( v_{li}^m \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_y} - \sum_{l=0}^{N+1} \sum_{m'=-2l}^{2l} \left( w_{li}^m \Delta_{l,m,m'}^2 \right) I_{2l}^m \frac{\partial}{\partial \tau_z} \tag{5c}
\]

These boundary conditions are expressed in terms of normal and tangential derivatives. Again, the coefficients \( u, v, w, p, \) and \( \Delta \) are computed based on expressions given in [5]. The \( \Delta \) coefficients depend on the Euler angles \( \alpha, \beta, \) and \( \gamma \) which are related to the local coordinate system defined by the vectors \( \mathbf{i}, \mathbf{j}, \) and \( \mathbf{k} \). These angles and vectors can be calculated based on the equations given in the following section.

As noted in the previous section, each \( P_N \) equation is used to numerically solve for each \( I_n^m \). With boundary conditions expressed in the form above, it is not clear how to assign a single boundary condition to each \( I_n^m \). The choice of which boundary condition to assign for a given \( I_n^m \) can not be arbitrary either because, depending on the geometry, the coefficients for \( I_n^m \) can vanish. In order to assign a boundary condition to each \( P_N \) equation for arbitrary configurations, the boundary conditions should be combined in order to solve for either all \( I_n^m \) or \( \partial I_n^m \)/\( \partial \tau \) depending on the type of boundary condition requirements of the numerical software package. In the case of OpenFOAM, it is straightforward to implement Robin-type boundary conditions, therefore, either \( I_n^m \) or \( \partial I_n^m \)/\( \partial \tau \) can be isolated, but here \( I_n^m \) will be isolated to get the boundary conditions in the form,

\[
I_n^m + k \frac{\partial I_n^m}{\partial \tau_z} = f \left( \frac{\partial I_n^m}{\partial \tau_x}, \frac{\partial I_n^m}{\partial \tau_y}, \frac{\partial I_n^m}{\partial \tau_z} \right) \tag{6}
\]
where \( k \) is a scalar constant and \( f \) is a function of partial derivatives of other intensity coefficients, including the tangential derivatives of \( I^m_n \). The boundary conditions are readily manipulated by reexpressing them in matrix form,

\[
Q \cdot I + Q_x \frac{\partial I}{\partial x} + Q_y \frac{\partial I}{\partial y} + Q_z \frac{\partial I}{\partial z} = I_w p,
\]

(7)

where the intensity coefficient vector \( I \), the Marshak boundary coefficient matrices \( Q, Q_x, Q_y, Q_z \), and the coefficient vector for the wall intensity \( p \) can be compactly defined. The intensity coefficient vector is defined as,

\[
I = \begin{bmatrix} I_0^0 \\ I_1^2 \\ \vdots \\ I_N^{-1} \end{bmatrix}, \quad \text{where } \quad I_n = \begin{bmatrix} I_n^0 \\ \vdots \\ I_n^{N-1} \end{bmatrix}.
\]

(8)

For example, for \( P_3 \) the intensity coefficient vector and the coefficient matrix is

\[
I = \begin{bmatrix} I_0^0 \\ I_1^2 \\ I_1^2 \\ I_1^2 \\ I_1^2 \\ I_1^2 \end{bmatrix}, \quad Q = \begin{bmatrix} q(0,1) \\ q(0,3) \\ q(-1,1) \\ q(1,1) \\ q(-2,3) \\ q(2,3) \end{bmatrix}.
\]

(9)

where \( q(\pm m,n) \) are row vectors corresponding to the same \((\pm m,n)\) combinations as for the \( P_3 \) local spherical harmonics \((\vec{Y}_{n}^{m})\), as described earlier in this section. These coefficient vectors can be computed as

\[
q(\pm m,n) = [p_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0}, p_{2,m}^m \tilde{\Delta}_2^m, \ldots, p_{N-1,-m}^m \tilde{\Delta}_{N-1}^{-m}]
\]

(10)

where \( q(\pm m,n) \) is a row of the matrix \( Q \) in Eqn. (7) and,

\[
\tilde{\Delta}_m^m = [\tilde{\Delta}_m^{m,-n} \cdots \tilde{\Delta}_m^{m,m}].
\]

(11)

Similarly, for the rows of the matrices \( Q_x, Q_y, \text{and } Q_z \), let

\[
q_x(\pm m,n) = \pm (1 \mp \delta_m,m) \left(1 - \delta_{m,0}\right) \times \left[u_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0}, u_{1,1}^m \tilde{\Delta}_1^1, \ldots, u_{N-1,2}^m \tilde{\Delta}_{N-1}^{2,-1}\right]
\]

(12)

\[
q_y(\pm m,n) = \mp (1 \mp \delta_m,m) \left(1 - \delta_{m,0}\right) \times \left[u_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0}, u_{1,1}^m \tilde{\Delta}_1^1, \ldots, u_{N-1,2}^m \tilde{\Delta}_{N-1}^{2,-1}\right]
\]

(13)

and,

\[
q_z(\pm m,n) = \pm (1 \mp \delta_m,m) \left(1 - \delta_{m,0}\right) \times \left[u_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0}, u_{1,1}^m \tilde{\Delta}_1^1, \ldots, u_{N-1,2}^m \tilde{\Delta}_{N-1}^{2,-1}\right]
\]

(14)

where \( H_x = 1 \) for \( +m > 0 \) and \( H_x = 0 \) otherwise, and \( H_y = 1 \) for \( -m \leq 0 \) and \( H_y = 0 \) otherwise. The vector \( p \) can be defined simply by specifying the \((\pm m,n)\) element as,

\[
p(\pm m,n) = p_{0,0}^m \delta(m,0).
\]

(15)

To illustrate how the first terms of the Marshak boundary conditions can be computed for a given \((m,n)\), consider the following vector product,

\[
q(\pm m,n) \cdot I = p_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0} p_0^0 + p_{2,m}^m \tilde{\Delta}_2^m I_2 + \cdots + p_{N-1,-m}^m \tilde{\Delta}_{N-1}^{-m} I_{N-1}
\]

(16)

\[
= p_{0,0}^m \tilde{\Delta}_0^0 \delta_{m,0} p_0^0 + \sum_{l=1}^{(N-1)/2} p_{0,2}^m \tilde{\Delta}_{2,2l}^2 I_{2l} + \sum_{l=1}^{(N-1)/2} \sum_{m'=-2l}^{2l} p_{m,2l}^m \tilde{\Delta}_{2,m',2l}^m p_{m',2l}^m,
\]

which reproduces the expressions of the first terms of the Marshak BCS. Isolating the vector of intensity coefficients \( I \) from Eqn. (7) by pre-multiplying with \( Q^{-1} \), the boundary conditions become

\[
I + Q^{-1} \cdot Q_x \frac{\partial I}{\partial x} = I_w Q^{-1} \cdot p - Q^{-1} \left( Q_x \cdot \frac{\partial I}{\partial x} + Q_y \cdot \frac{\partial I}{\partial y} \right).
\]

(17)

The form in Eqn. (6) is obtained by defining \( D_x \) as the diagonal matrix of \( Q^{-1} \cdot Q_x \) and \( UL_x \) as the off-diagonals (or upper + lower triangular matrices) to obtain,

\[
I + D_x \cdot \frac{\partial I}{\partial x} = I_w Q^{-1} \cdot p - UL_x \cdot \frac{\partial I}{\partial x} - Q^{-1} \left( Q_x \cdot \frac{\partial I}{\partial x} + Q_y \cdot \frac{\partial I}{\partial y} \right).
\]

(18)

By using this representation of the boundary conditions, there is no concern of important terms vanishing since the leading term \((I)\) does not contain any geometry-dependent coefficients.

It is worth noting that the first term on the right side of Eqn. (18) is trivially computed by recognizing that

\[
Q^{-1} \cdot p = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^T.
\]

(19)

This identity is easily verified by letting \( x = [1,0,\ldots,0]^T \), then taking the product with \( q(\pm m,n) \) (from Eqn. (10)) to obtain,

\[
q(\pm m,n) \cdot x = p_{0,0}^m \delta(m,0).
\]

(20)
As suggested in [9], the following rotation matrix should be used for cases where \( \sin \beta = 0 \) (or \( n_z = \pm 1 \)),

\[
R = \begin{bmatrix}
\pm \cos(\alpha \pm \gamma) & -\sin(\alpha \pm \gamma) & 0 \\
\pm \sin(\alpha \pm \gamma) & \cos(\alpha \pm \gamma) & 0 \\
0 & 0 & \pm 1
\end{bmatrix}.
\] (23)

The corresponding tangential directions are obtained by applying the above rotations to the unit global coordinate vectors:

\[
t_x = R \cdot \hat{i} \quad \text{and} \quad t_y = R \cdot \hat{j}.
\] (24)

Equivalently, \( t_x \) is equal to the first column of \( R \), and \( t_y \) is equal to the second column of \( R \). The normal and tangential derivatives are then computed using the vector products,

\[
\frac{\partial}{\partial t_x} = n \cdot \nabla, \quad \frac{\partial}{\partial t_y} = t_x \cdot \nabla, \quad \text{and} \quad \frac{\partial}{\partial t_y} = t_y \cdot \nabla.
\] (25)

**IMPLEMENTATION DETAILS**

**Segregated solver**

The most direct implementation of the SHM in OpenFOAM involves solving the system of equations sequentially. A blocked-coupled solution strategy [8] of the \( N(N+1)/2 \) PDEs is much more involved and memory intensive, but will be considered in future developments.

In OpenFOAM, the PDE solvers make use of finite volume based discretizations of certain terms that are expressed as standard differential operators (\( \nabla, \nabla \times, \nabla \cdot, \nabla^2, \partial / \partial t, \partial^2 / \partial \theta^2 \)). Depending on whether \( \text{fvm} \) (finite volume method) or \( \text{fvc} \) (finite volume calculus) is selected to compute an operator, OpenFOAM returns either a matrix or geometric field. In OpenFOAM, a returned matrix means that the terms are discretized so that a linear algebraic system of equations can be solved. From a numerical standpoint, the difference between terms computed with \( \text{fvm} \) or \( \text{fvc} \) is that the terms computed with \( \text{fvm} \) are implicit calculations where the discretization of the terms are used to determine new values of the dependent variable, and the \( \text{fvc} \) calculated terms are explicit terms held fixed during a sequence of an inner iteration.

For the \( P_N \)-approximation, the only operator that is useful for implicit calculations is the \( \nabla^2 \) operator in Cartesian coordinates. Therefore, the last two terms with \( \mathcal{L} \) operators for \( k = 2 \) in Eqn. (3) are combined to get the following expression,

\[
(L_{xx} + L_{yy} - 2L_{zz})c_2^{nm}I_n^{mn} + L_{zz}c_2^{mn}I_n^{nm}
= c_2^{nm} \nabla^2 I_n^{nm} + (1 - 3c_2^{nm})L_{zz}I_n^{nm}
\] (26)

The \( P_N \) system of equations are solved iteratively, by denoting \( I_j = I_n^{km} \), where \( j \) denotes the sequence number and goes for \( j = 1, \ldots, N(N+1)/2 \),

\[
\nabla \cdot \left( \frac{1}{\beta} \nabla I_j \right) + \beta I_j = S_j.
\] (27)
In this expression, $S_j$ is the summation of all explicit terms not included in Eqn. (26), and each computed using the OpenFOAM fvc option, and the Laplacian operator is computed by the OpenFOAM fvm option, with $1/\beta$ passed as an argument. The sequence of numerical computations is repeated until convergence, which typically requires about a dozen or more inner iterations.

Compared to $P_1$, which does not require inner iterations, the number of numerical computations for higher order $P_N$ is approximately $N(N + 1) \times M$, where $M$ is the number of inner iterations. For instance, if applying $P_3$ and if $M = 10$, then $P_3$ requires approximately $6 \times 10 = 60$ more numerical calculations than $P_1$. Since the computational time is proportional to number of additional numerical calculations, the extra computational time can be estimated a priori to be at least an order of magnitude more expensive than $P_1$.

Boundary conditions

In OpenFOAM, Robin-Type boundary conditions are applied by specifying a reference value $R$ for the right hand side of the BC and a value fraction $F$ that accounts for the ratio between the derivative and non-derivative term in the BC. Boundary face values are updated according to

$$I_{j,\text{face}} = F \cdot R + (1 - F) \cdot I_{j,\text{center}}$$

(28)

where $I_{j,\text{face}}$ is the value of $I_j$ at the boundary face and $I_{j,\text{center}}$ is the value at the center of the neighboring cell. This equation conforms to the boundary condition Eqn. (6) if $R$ is equal to the $j$th row of Eqn. (18), and if

$$F = \frac{\beta}{\beta + D_{\delta,j}/\delta},$$

(29)

where $D_{\delta,j}$ is the $j$th diagonal element of $D_\delta$ and $\delta$ is the distance between the boundary face and the center of the neighboring cell value. Eqn. (6) can be discretized and reexpressed as,

$$F \cdot I_{j,\text{face}} + (1 - F) \cdot (I_{j,\text{face}} - I_{j,\text{center}}) = F \cdot R$$

(30)

so that Eqn. (28) is obtained after reordering.

### APPLICATIONS

Three example problems are shown in this section. Descriptions of the problems are provided in Table 1. Each of these problems considers nonscattering mediums with variable emission and absorption coefficient fields.

#### 1-D slab with variable medium properties

In the first verification examples, the $P_N$-approximations are applied to a 1-D slab that encloses a medium with variable properties. For this problem, the walls of the enclosure are assumed cold and black and the length of the slab is $L = 1\text{m}$. The optical thickness is $\tau_L = 0.5$. In optical coordinates the temperature field (Eqn. (32)) is linear, which is convenient for finding a particular solution for the analytical solution (see Appendix A). For the numerical computations, a lower limit of $\kappa_{\text{min}} = 10^{-4}\text{m}^{-1}$ was set for $\kappa$ in order to avoid division by $0$ at the lower wall.

Although this example is only 1-D, having the analytical solutions is useful for the purposes of verifying certain aspects of the $P_N$ approximation and the program implementation itself. For instance, by orienting the 1-D slab at $\phi$ angles within each of the three coordinate planes (see Fig. 2), the 1-D slab can be used to verify that all partial derivatives are accounted for correctly in the implementation. For example, setting the configuration angle to $\phi = 0$ with the slab positioned in the $xy$ plane, one can test the $L_{xx}$ derivatives, neglecting all mixed derivative, $L_{yy}$, and $L_{zz}$ terms from the PDEs. Similarly, setting $\phi = 90^\circ$, the combination of $L_{yy}$ derivatives can be singled out, and for $\phi = 45^\circ$, the combination of $L_{xx}$, $L_{yy}$, $L_{xy}$, and $L_{yx}$ terms can be verified. In terms of the radiation profile within the slab, the solutions for $I_0^j$ should

---

**TABLE 1.** Example problem descriptions.

<table>
<thead>
<tr>
<th>Problem 1</th>
<th>Problem 2</th>
<th>Problem 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometry</td>
<td>1-D slab</td>
<td>2-D enclosure</td>
</tr>
<tr>
<td>Spatial variable, $r$</td>
<td>distance from lower wall</td>
<td>distance from center</td>
</tr>
<tr>
<td>Absorption coefficient $\kappa = r$</td>
<td>$\kappa = C_\kappa \left(1 + \frac{15}{4} (2 - r^2)^2\right)$</td>
<td>$\kappa = C_\kappa \left(1 + 240 (\frac{1}{4} - r^2)^2\right)$</td>
</tr>
<tr>
<td>Black-body Intensity $I_b = 10(1 + \frac{1}{2} r^2)$</td>
<td>$I_b = 1 + 5r^2(2 - r^2)$</td>
<td>$I_b = 1 + 320r^2(2 - r^2)$</td>
</tr>
<tr>
<td>Optical thickness $\tau_L = 0.5$</td>
<td>$\tau_D = 1.6$</td>
<td>$\tau_D = 4.5$</td>
</tr>
</tbody>
</table>

---

3See lines 00143-00162 in code located at http://foam.sourceforge.net/docs/cpp/a04020_source.html.

---

**FIGURE 2.** Rotations of 1D slab at angles $\phi = 0, 45^\circ, -45^\circ$ in the $xy$-plane. The medium properties increase with low values at the lower wall to higher values at the upper wall.
be found to be identical (and matching the analytical profiles) for arbitrary orientations of the 1-D slab.

The numerical and analytical solution of \( G \) and \( \nabla \cdot \mathbf{q} \) for the case of the slab oriented in the \( y \)-direction is shown in Figs. 3(a) and 3(d). The numerical results were found to perfectly overlap the analytical results for various orientations which demonstrates that the \( P_N \) implementation correctly employs all the terms in the PDEs. Comparing with the exact solution for this 1-D problem [1], higher order \( P_N \) generally produce more accurate results over the entire slab, except near the lower wall where it seems that the \( P_N \)-approximations incur larger errors. This is probably due to very low values of \( \kappa \) at the lower wall. Nevertheless, the overall error is reduced by approximately 50% every time the order is increased for this example.

2-D enclosure with variable medium properties

The 2-D enclosure problem that is considered here was adopted from [5, 10]. This problem is another example of a non-scattering medium, but with variable absorption coefficient and temperature field. These properties vary according to the what is listed in the second column of Table 1. The enclosure walls are assumed to be cold and black. The optical thickness taken along the diagonal is \( \tau_D = 18 \sqrt{2} C \). The maximum value of \( \kappa \) occurs at \( r = 0 \) with a value of \( \kappa_{\text{max}} = 16C \). For the results presented here, \( C \) was set to \( C = 0.1 \) (optically intermediate).

Calculated profiles of \( G \) and \( \nabla \cdot \mathbf{q} \) using \( P_1 \) through \( P_7 \) along the diagonal \( x = y \) are plotted in Fig. 3(b) and 3(e). The results of the higher order \( P_N \) are consistently more accurate as the order increases, with \( P_7 \) nearly identical to the exact solution obtained directly from the RTE. For this optically intermediate case, increasing the order \( N \) produces more accurate results for \( G \), about as significantly as for the previous 1-D optically thin case.

The coordinate invariance properties of the \( P_N \)-approximation were also tested by applying the approximations for the 2-D enclosure oriented on each of the \( xy \), \( yz \) and \( xz \)-planes. The resulting surface plots of \( G \) computed with \( P_3 \) are displayed in Fig. 4. Each of these plots give identical results for \( P_0 \), and thus \( G \), which is to be expected. However, for each orientation, different intensity coefficients are relevant. By examining the PDEs and Marshak boundary conditions for the \( xy \)-configuration (where all terms with \( y \)-derivatives vanish), and for the \( yz \)-configuration (where all terms with \( x \)-derivatives

![Figure 3](image-url)
The optical thickness along the diameter is and has cold-black walls. The parameter $C_D$ is used to adjust the optical thickness along the diameter is calculated as $\tau_D = 9C_e$. Profiles of $G$ and $\nabla \cdot q$ across the cylinder at $z = 1m$ for $C_e = 0.5$ ($\tau_D = 4.5$) are shown in Figs. 3(c) and 3(f).

An exact solution for this problem is calculated directly from the integral solution from the RTE. The solutions for $P_5$ and $P_7$ are close to the exact solution, however, increasing the order of $P_N$ beyond $N = 5$ changes very little the accuracy for this problem.

Surface plots of the computed intensity coefficients are shown in Fig. 5. Values of the intensity coefficients $I_1^1$ and $I_2^2$ near the center of the cylinder are negligible because of the absent variations in the $z$-direction. Therefore, these intensity coefficients are not shown. As this figure shows, the intensity coefficients $I_2^1$ and $I_2^2$ have variations in both the radial and polar directions, whereas, $I_0^0$, has only $r$ dependence which is to be expected. If this problem were to be applied on a wedge, then it appears that boundary conditions for $I_2^1$ and $I_2^2$ such as Marshak, zero-gradient, or periodic on the wedge faces will not be consistent with the full-cylinder solution. Remedies for this wedge surface BC problem have not yet been addressed elsewhere and is beyond the scope of the current presentation.

**Cylindrical enclosure with variable medium properties**

A cylindrical case is presented here to demonstrate $P_N$ solutions for axisymmetric problems. In these types of problems, physical quantities such as temperature, heat flux, intensity, chemical species concentrations, etc., vary only radially and axially, and are considered as 2-D. As a result, for many of these applications, the transport equations are solved over a wedge in order to reduce the computation effort in solving these classes of problems. Here, a full cylinder rather than a wedge is considered in order to demonstrate that the resulting intensity coefficients have azimuthal dependence and, therefore, special care must be taken regarding boundary conditions for general axisymmetric cases.

For this example problem, the medium in the cylinder has the variable properties according to the third column of Table 1. The cylinder has a diameter $D = 1m$, a length of $L = 2.8m$, and has cold-black walls. The parameter $C_e$ is used to adjust the optical thickness. The optical thickness along the diameter is

$$
\begin{bmatrix}
I_0^0 \\
I_2^1 \\
I_2^2 \\
I_2^2_{(xz)} \\
I_2^2_{(yz)}
\end{bmatrix} =
\begin{bmatrix}
I_0^0 \\
I_2^1 \\
I_2^2 \\
I_2^2_{(xz)} \\
I_2^2_{(yz)}
\end{bmatrix},
$$

where the subscripts denote the plane of the orientations of the 2-D slab. These identities make for useful criteria to verify consistency of the numerical solutions.

**Computational expense**

While the higher order $P_N$-approximations have been shown here to be more accurate, they typically require more cpu time. Details regarding the computational effort encountered on the example problems that were presented here are displayed in Table 2. This table includes the number of interior cells $N_c$ for each mesh, the cpu time, and the number of inner iterations ($N_t$). Compared with the experiments from [6], where a 2-D enclosure problem is solved with $N_c = 242$ and the cpu time was reported...
as \(<1\) s, the application here of \(P_3\) on the 2-D example took 6.22 s (an order of magnitude greater) for a much larger \(N_c\) (also an order of magnitude greater). According to the cpu time results, it appears that \(P_3\) takes at least 2 orders of magnitude longer than \(P_1\), which is more than what we estimated in Section 3. It is possible that most of the extra cpu time is due to new computations of the terms in the right hand side of Eqs. (18) and (27) for every sequence \(j\). A thorough profiling of the module will be needed to verify this, however.

### CONCLUSIONS

The implementation of the spherical harmonics method (SHM) in OpenFOAM confirms that higher order \(P_N\) generally produce higher accuracy as the order \(N\) increases. This work demonstrated higher orders of \(P_N\) up to \(P_7\). Details regarding the implementation of SHM in OpenFOAM were presented including a representation of the boundary conditions in matrix form so that Robin-type BCs can be used from OpenFOAM. The OpenFOAM implementation has only included a segregated solution method where the set of PDEs are solved sequentially and iteratively. For the applications considered here, this approach converges satisfactorily. However, it may be possible to improve computational time and numerical stability of \(P_N\) by pursuing a block-coupling approach [8]. Also, the results for a cylindrical case warrant that special care should be taken for axisymmetric problems. Strategies to enforce axisymmetric conditions are still in development.

### REFERENCES


### APPENDIX

The solution method for the one-dimensional enclosure problem is to express the system of PDEs resulting after neglecting variations in the two other dimensions. For 1-D problems, Eqn. (3) results in a linear system of ODEs which are expressed in matrix form as,

\[
A \cdot I'' = I - I_b
\]  

where \(A\) is a coefficient matrix, \(I\) is the vector of intensity coefficients, \(I''\) is the second derivative of \(I\) with respect to optical coordinates, and \(I_b\) has all zero elements except for the first row which corresponds to \(I_0\) where the element is equal to \(I_b(\tau)\). For 1-D in the \(x\)- or \(y\)-direction, the intensity coefficients corresponding to odd or negative \(m\) are zero, and in the \(z\)-direction all terms are zero except for the intensity coefficients corresponding to \(m = 0\). Thus, after neglecting terms that vanish, the dimensions of these vectors are reduced to \(N\) if the \(x\) or \(y\) direction is chosen, or \((N + 1)/2\) if the \(z\) direction is chosen.

For convenience, the resulting system of ODEs may be re-expressed as,

\[
I'' = A^{-1} \cdot I - A^{-1} \cdot I_b.
\]  

The boundary conditions are given by the Marshak boundary conditions. For 1-D, the terms involving tangential derivatives vanish, thus the Marshak boundary conditions are

\[
I(\tau^+) = \mathbf{Q} \cdot \mathbf{I}(\tau^-) = \mathbf{Q} \cdot \mathbf{p} I_{in}(\tau^-),
\]

where \(\mathbf{Q}\) is the vector of boundary intensities, and \(\mathbf{p}\) is the normal vector at the boundary.

---

**Copyright © 2013 by ASME**
where the + is applies to the upper wall and the − applies to the lower wall. The solution for this system is obtained by adding together the homogeneous solution and the particular solution \( I = \mathbf{I}_H + \mathbf{I}_P \), then determining the constants of integration through the boundary conditions. The homogeneous solution is the solution to the eigenvalue problem,

\[
\mathbf{I}_H'' = \mathbf{A}^{-1} \cdot \mathbf{I}_H \tag{35}
\]

which leads to the solution,

\[
\mathbf{I}_H(\tau) = \Phi \cdot \cosh(\Lambda \tau) \cdot \mathbf{c}_1 - \Phi \cdot \sinh(\Lambda \tau) \cdot \mathbf{c}_2 \tag{36}
\]

where \( \Lambda \) is a diagonal matrix with entries corresponding to the square root of the eigenvalues of \( \mathbf{A}^{-1} \) and \( \Phi \) is the set of associated eigenvectors. The particular solution is obtained for simple situations. Assuming that \( \mathbf{I}_H \) varies linearly with \( \tau \) so that \( \mathbf{I}_H'' = 0 \), the particular solution is easily found,

\[
\mathbf{I}_P(\tau) = \mathbf{I}_P(\tau) \tag{37}
\]

After applying \( \mathbf{I} = \mathbf{I}_H + \mathbf{I}_P \) into the boundary conditions, the following system of equations are obtained,

\[
\begin{bmatrix}
\Phi \cdot \cosh(\Lambda \tau^+) \pm \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \Phi \cdot \Lambda \cdot \sinh(\Lambda \tau^+) \cdot \mathbf{c}_1 \\
\Phi \cdot \sinh(\Lambda \tau^+) \pm \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \Phi \cdot \Lambda \cdot \cosh(\Lambda \tau^+) \cdot \mathbf{c}_2
\end{bmatrix}
= \mathbf{Q}^{-1} \cdot \mathbf{p}_{bw} + \mathbf{I}_b \mp \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \mathbf{I}_b \tag{38}
\]

To abbreviate notation, let

\[
\begin{align*}
\mathbf{V}_1^+ &= \Phi \cdot \cosh(\Lambda \tau^+) \pm \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \Phi \cdot \Lambda \cdot \sinh(\Lambda \tau^+), \\
\mathbf{V}_2^+ &= \Phi \cdot \sinh(\Lambda \tau^+) \pm \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \Phi \cdot \Lambda \cdot \cosh(\Lambda \tau^+), \\
\mathbf{b}^+ &= \mathbf{Q}^{-1} \cdot \mathbf{p}_{bw} + \mathbf{I}_b(\tau^+) \mp \mathbf{Q}^{-1} \cdot \mathbf{Q}_2 \cdot \mathbf{I}_b 
\end{align*}
\]

and from Eqn. (40),

\[
\begin{bmatrix}
\mathbf{b}^- \\
\mathbf{b}^+
\end{bmatrix} = \frac{1}{3} \begin{bmatrix}
-14 & -8 & 8 & 4
\end{bmatrix}^T. \tag{46}
\]

Then from Eqn. (41), the constants of integration are

\[
\begin{bmatrix}
\mathbf{c}_1 \\
\mathbf{c}_2
\end{bmatrix} = \begin{bmatrix}
-4.3387 & -5.1268 & 0 & 0.7846 & -4.0544 & 0
\end{bmatrix}^T. \tag{47}
\]

Therefore, the solution for \( \mathbf{I}_b^0 \) is

\[
\mathbf{I}_b^0 = 10(1 + \frac{1}{2} y^2) \\
+ 0.7385(-4.3387 \cosh(2.9413 + \frac{1}{2} y^2)) \\
+ 0.5045(-5.1268 \cosh(1.1613 + \frac{1}{2} y^2)) \\
+ 0.7385(+0.7846 \sinh(2.9413 + \frac{1}{2} y^2)) \\
+ 0.5045(-0.40544 \sinh(1.1613 + \frac{1}{2} y^2))
\]

where \( \tau \) was replaced for \( \frac{1}{2} y^2 \) to represent the solution in geometric coordinates. The solutions presented in the paper are for \( G = 4\pi \mathbf{I}_b^0 \) using \( P_3 \) up to \( P_7 \) and were compared to the numerical results.