3 Boundary Value Problems
Larry C. Young, tildentechnologies.com

Table of Contents
3 Boundary Value Problems ........................................................................................................... 1
3.1 Diffusion/Conduction with Source ........................................................................................ 1
3.1.1 Orthogonal Collocation Method ..................................................................................... 2
   Linear Source, Constant Coefficients, Dirichlet B.C. ......................................................... 4
   Linear Source, Constant Coefficients, Third Kind B.C. ..................................................... 8
3.1.2 Moments Method ............................................................................................................ 9
3.1.3 Galerkin Method ........................................................................................................... 11
   Linear Source, Constant Coefficients, Third Kind B.C., Galerkin/Moments. ..................... 14
3.1.4 Mass Conservation and Fluxes ...................................................................................... 15
   Linear Source, Variable Coefficients, Dirichlet B.C. ........................................................ 17
3.1.5 Symmetric Problems .................................................................................................... 22
   kth Order Source, Dirichlet B.C., Various Geometry ......................................................... 25
   Autocatalytic Source, Third Kind B.C. .............................................................................. 28
3.1.6 Orthogonal Polynomial Trial Functions ....................................................................... 33
   Moments Method ............................................................................................................. 35
   Galerkin Method ............................................................................................................. 36
   Variable Coefficients ..................................................................................................... 38
3.2 Chemical Reactor with Axial Dispersion ............................................................................ 39

3.1 Diffusion/Conduction with Source
Consider reaction and diffusion in a porous slab. This differential equation is one of the most ubiquitous ones in engineering and physics. It is analogous to heat conduction in a slab with a heat source which is dependent on temperature and position. If the source, \( r \), is constant, the equations describe laminar flow between parallel plates. It also describes the deflection of a beam, where \( r \) characterizes the load. The governing equations are:

\[
\frac{\partial^2 y}{\partial x^2} + r(x, y) = 0
\]

(3.1)

where \( r(x, y) = 4\varphi^2 \hat{r}(x, y) \) and \( y \) is the fractional conversion, 0 for no reaction and 1 for complete reaction. \( \varphi \) is called the Thiele modulus. The average value of \( \hat{r}(x,0) \) is 1 by definition. The boundary conditions are either first kind, Dirichlet, boundary conditions:

\[
y(0) = y(1) = 0
\]

(3.1a)

or third kind, Robin, boundary conditions:

\[
\left. \frac{dy}{dx} \right|_{x=0} = 2Bi y(0) \quad \text{and} \quad \left. \frac{dy}{dx} \right|_{x=1} = 2Bi y(1)
\]

(3.1b)
In Eq. (3.1b), the Biot number, $Bi$, accounts for an external transfer resistance. As $Bi$ tends to infinity, Eq. (3.1b) reduces to Eq. (3.1a). The Thiele modulus and Biot numbers are normally defined using half the thickness of the slab, so the factors of 4 and 2 are required when the full slab is considered.

This problem has been heavily studied in chemical engineering, including various geometries and highly nonlinear source functions which lead to multiple solutions. For a more thorough discussion see Villadsen and Michelsen (1978), Rawlings and Ekerdt (2015) and references therein.

For a simple $k^{th}$ order reaction with the reactivity independent of position, the source term is:

$$
\dot{r}(x, y) = (1 - y)^k
$$

(3.2)

For a first order reaction, $k = 1$, the analytical solution is:

$$
y = 1 - \frac{\cosh[\varphi(2x - 1)]}{\cosh(\varphi) + \varphi \sinh(\varphi)/Bi}
$$

(3.3)

The quantity of interest from the solution is called the effectiveness factor, $\eta$, which gives the overall rate of reaction relative to that with no diffusional resistance:

$$
\eta = \frac{\int_0^1 r(x, y)dx}{\int_0^1 r(x, 0)dx} = \int_0^1 \dot{r}(x, y)dx
$$

(3.4)

The effectiveness factor is basically a normalized boundary flux, since the divergence theorem in one dimension gives:

$$
-\frac{dy}{dx}igg|_0^1 = 4\varphi^2 \int_0^1 \dot{r}(x, y)dx = 4\varphi^2 \eta
$$

(3.5)

Eq. (3.5) is given the generic name average energy equation, since in a heat transfer setting it is an overall energy and the right-hand side is the average energy generated. Using the analytical solution, Eq. (3.3), this normalized flux is:

$$
\eta = \frac{1}{\varphi[\coth(\varphi) + \varphi/Bi]}
$$

(3.6)

$\eta$ is approximately one for $\varphi < \frac{1}{2}$ and becomes asymptotic when $\varphi > 2$. The asymptotic state corresponds to a condition where all the reactants are consumed and the conversion, $y$, approaches one at the center.

### 3.1.1 Orthogonal Collocation Method

To solve the problem using a Method of Weighted residuals (MWR), we start with a trial solution following Eqs. (1.6) and (2.2). Some developments of the method state that the trial functions are orthogonal polynomials, but then use monomials and through transformations formulate the problems in terms of nodal values. Although the approximation turns out the
same, we prefer to dispense with the transformations and develop the method directly as follows:

\[ y = \sum_{i=0}^{n+1} y(x_i) \ell_i(x) \]  \hspace{1cm} (3.7)

The interpolation points include the endpoints, \( x_0 = 0 \) and \( x_{n+1} = 1 \). As explained in Chapter 1, the interior points are the roots of an orthogonal polynomial, usually either Chebyshev polynomials of the 2\(^{nd}\) kind, Legendre polynomials (Gauss points) or the base points of Lobatto quadrature. These choices are all roots of specific types of Jacobi polynomials.

The residual is formed by substitution of the approximate solution into the equation:

\[ \sum_{i=0}^{n+1} y(x_i) \frac{d^2 \ell_i}{dx^2} + r \left( x, \sum_{i=0}^{n+1} \ell_i(x) y(x_i) \right) = R(x,y) \]  \hspace{1cm} (3.8)

Where \( y \) is the vector of nodal values \( y(x_i) \). With the collocation method the residual is set to zero at the interior collocation points, \( x_j, j = 1, \ldots, n \). Since \( \ell_i(x_j) = \delta_{ij} \) (the dirac delta function), the resulting equation simplifies to:

\[ \sum_{i=0}^{n+1} y(x_i) \frac{d^2 \ell_i}{dx^2} \bigg|_{x_j} + r \left( x_j, y(x_j) \right) = 0 \]  \hspace{1cm} (3.9)

By defining \( B \) as indicated below and letting \( y_i = y(x_i) \) the equation simplifies to:

\[ \sum_{i=0}^{n+1} B_{ji} y_i + r(x_j, y_j) = 0 \]  \hspace{1cm} (3.10)

The boundary conditions provide two additional conditions, either first kind, Dirichlet, boundary conditions:

\[ y_0 = y_{n+1} = 0 \]

or third kind, Robin, boundary conditions:

\[ \sum_{i=0}^{n+1} A_{0,i} y_i = 2Bi y_0 \quad \text{and} \quad -\sum_{i=0}^{n+1} A_{n+1,i} y_i = 2Bi y_{n+1} \]  \hspace{1cm} (3.11)

where:

\[ A_{ji} = \frac{d \ell_i}{dx} \bigg|_{x_j} \quad \text{and} \quad B_{ji} = \frac{d^2 \ell_i}{dx^2} \bigg|_{x_j} \]

For Dirichlet conditions, Eq. (3.1a), the boundary values (whether zero or finite) are simply substituted into Eq. (3.10). For the third kind conditions, Eq. (3.1b), most texts [e.g. Finlayson (1972), p. 101; Villadsen and Michelsen (1978), p. 137; Trefethen (2000), p. 137; Boyd (2000), p. 111, Peyret (2002), p. 59] recommend that the boundary condition be satisfied exactly as in Eq. (3.11). We call this boundary collocation. We will have much more discussion on issues related to boundary derivatives or fluxes and will demonstrate methods that are superior to Eq. (3.11) in many cases.
To solve the problem, we need only the collocation points, i.e. the roots of the orthogonal polynomial, and the $A$ and $B$ matrices, the derivatives of the Lagrange interpolating polynomials. We also need the quadrature weights, $W$, for approximating integrals, e.g. Eq. (3.4). As described in chapter 2, all of these quantities can be calculated from the collocation points.

Eqs. (3.10) together with the boundary conditions are a set of algebraic equations. After substitution of the boundary values for Dirichlet conditions, we are left with $n$ equations which are usually nonlinear. A Newton-Raphson procedure works well for the nonlinear solution. If the reaction is linear, i.e. first order or 0th order, the solution is obtained after one Newton iteration. Given some estimate, $y^s$, a single iteration to solve for the change to get an improved estimate requires solution of:

$$\sum_{i=1}^{n} \left( B_{ji} + \delta_{ji} \frac{\partial r}{\partial y} \right) (y_i^{s+1} - y_i^{s}) = - \sum_{i=1}^{n} B_{ji} y_i^{s} - r(y_i^{s}, x_i) \quad (3.12)$$

To reduce roundoff errors, it is better to formulate the iterations with the residual and the change in $y$.

The nonlinear reaction terms appear only on the diagonal, which simplifies the calculations. We shall see that for a full Moments or Galerkin method, the reaction terms are distributed throughout the matrix.

**Linear Source, Constant Coefficients, Dirichlet B.C.** Fig. 3.1 shows solutions of Eqs. (3.1) and (3.1a) for a first order reaction and $\phi = 5$, i.e. Eq. (3.2) with $k = 1$. Approximate solutions are shown with $n = 4$ for collocation at Lobatto, Gauss and Chebyshev points. With this relatively high reaction rate most of the reaction occurs near the boundary. The fifth order polynomial can only approximate the sharp profile by oscillating about the exact solution. Actually, since this problem is symmetric about $x = 0.5$, the coefficient of the fifth order term is zero. Later, we will look at efficient methods to exploit symmetry. Clearly, for this example,
Lobatto points produce a more accurate solution followed in order by Chebyshev and Gauss points.

Fig. 3.2 shows the $L_2$ error norms versus $n$ for the three choices of points. A plot of the $L_1$ error norm looks very similar. All methods show the typical exponential convergence with Lobatto points giving slightly better results. The error with Chebyshev and Gauss points averages 1.3 and 1.9 times that with Lobatto points, which is relatively small since the error decreases almost an order of magnitude with each increment of $n$.

Using Eq. (3.6), the normalized boundary flux, $\eta = 0.199983$. Numerical quadrature can be used to calculate this value from the numerical solutions using Eq. (3.4):

$$\eta = \sum_{i=0}^{n+1} W_i \hat{r}(x_i, y_i)$$

(3.13)

For the cases in Fig. 3.1 with $n = 4$, the calculated fluxes are in error by 0.8, -3.1 and -4.3 percent for Lobatto, Chebyshev and Gauss points when the flux is calculated by Eq. (3.13). The same quantity can be approximated using Eq. (3.5). The derivatives are given by:

$$\left. \frac{dy}{dx} \right|_{x=0} = \sum_{i=0}^{n+1} A_{0,i} y_i \quad \text{and} \quad \left. \frac{dy}{dx} \right|_{x=1} = \sum_{i=0}^{n+1} A_{n+1,i} y_i$$

(3.14)

Using Eq. (3.14), the derivatives of the solution at the boundaries are in error by -14.3, -10.2 and -4.3 percent for Lobatto, Chebyshev and Gauss points. These errors are much larger and the relative accuracy of the methods is a complete reversal of that found with Eq. (3.13). Only Gauss points give the same result with either method of flux calculation. Shortly, we will explain why this is so. For the other two methods, integration gives a far more accurate result.

Fig. 3.3 shows the flux errors for increasing $n$. Relative to Fig. 3.2, this graph shows a much greater difference between the methods. For $n < 4$, the error with all three methods is relatively large, while for $n > 14$, some of the results are affected by rounding errors. Engineering accuracy is obtained with 4 to 8 points depending on the method and accuracy required. In this range, the errors with Gauss and Chebyshev points are similar, while Lobatto points give somewhat greater accuracy.

Since Figs. 3.2 and 3.3 are so different let us take a closer look at these measures of the error. Given the exact solution, $y^*$, the $L_p$ error norm is a measure of the error in the internal profile:
\[ \epsilon_p = \left[ \int_0^1 |y - y^*|^p \, dx \right]^{1/p} \]  

(3.15)

The error in the normalized flux for this linear source function is:

\[ \epsilon_\eta = \left| \int_0^1 (y - y^*) \, dx \right| \]  

(3.16)

These two error measures look similar, especially when \( p = 1 \). However, upon closer examination, it is clearly possible to achieve an exact flux with an imperfect solution which oscillates about the exact solution but in a way that gives the correct solution on average. To gain a better understanding of the nature of the error, Figs. 3.4 - 3.7 show the error, i.e. \( y - y^* \), and the residual, \( R \) in Eq. (3.7) for several cases.

In Figs. 3.4 and 3.6 the residuals are, of course, zero at the collocation points, but we note that the errors shown in Figs. 3.5 and 3.7 are very nearly zero at the collocation points and the positive and negative deviations from zero are nearly balanced, so it appears reasonable that the error calculated by Eq. (3.16) should be small. However, all of the points appear to exhibit this behavior, so why do the solutions converge more slowly with Chebyshev points? A simple
experiment reveals a possible explanation. If values from the analytical solution, Eq. (3.3), are integrated using the quadrature formulas, Eq. (3.13), the error due exclusively to the quadrature is obtained. The quadrature errors are shown together with the errors from the approximate solutions in Fig. 3.8. By examination of Fig. 3.8 we see that the errors with Chebyshev points (Clenshaw-Curtis quadrature) do not improve as much as the others. Even with the exact solution for the profile, the Chebyshev convergence rate and accuracy is worse than the results for the overall approximate solutions with the other two methods.

Villadsen and Michelsen (1978) (p. 85) have noted that other methods may produce similar profile errors as we find here (see Fig. 3.2), but the Moments and Galerkin methods do an especially good job of balancing the error, so that the flux calculation is very accurate. It is fortunate that Gauss and Lobatto points have this property, because the flux is the most important result of the calculation. As we shall see, these two methods have equivalence to Moments and Galerkin methods, respectively. The Chebyshev points may also achieve this improvement, but the Clenshaw-Curtis quadrature is not accurate enough to take advantage of the more accurate solutions at the points.

It is likely that the greater observed convergence rate of Galerkin and Moments methods is related to the long known superconvergence property of some finite element methods [see Kržek and Neittaanmäki (1998)]. The superconvergence phenomenon is one where the solution at certain points converges at a faster rate than the solution as a whole. Some collocation finite elements methods also display this property at the collocation points. Since the global methods considered here are but a single element of a finite element procedure, it is likely the exceptional accuracy of the flux calculations in Fig. 3.3 is a related phenomenon.

The error curves in Figs. 3.2 and 3.3 converge at a supergeometric rate, i.e. with \((n)\log(n)\), but are plotted versus \(\log(n)\) as is customary. Lobatto points give the best convergence rate. Gauss points converge at the same rate, but the errors are about 10 to 15 times larger. The convergence rate with Chebyshev points is roughly half that with Gauss or Lobatto points. If derivatives are used to calculate the flux, Eq. (3.14), the results are poor with Lobatto or Chebyshev points. Gauss points produce the same result regardless of calculation method.

Since all the methods give exponential convergence and the \(L_p\) error norms are similar, one could easily be misled by an incomplete comparison. Significant differences show up only when fluxes are compared. One of the advantages of orthogonal collocation or pseudospectral
methods is that virtually exact solutions are feasible. The method with Lobatto points is superior for achieving high accuracy for this example, but only if fluxes are accurately calculated.

**Linear Source, Constant Coefficients, Third Kind B.C.** Now, consider the same problem as above, but with the third kind boundary conditions, Eq. (3.1b). It is clear from Eqs. (3.3) and (3.5) that $\phi/\text{Bi}$ gives the relative importance of internal and external resistance. The shape of the profile is not changed, but the boundary value is scaled up to account for the external resistance. We present calculations here with $\phi = 5$ as above and $\text{Bi} = 10$. For these conditions both are important, but the external resistance is less important than the internal resistance. Other values of Bi are considered to determine its effect on the results.

Most texts recommend that boundary collocation, Eq. (3.11), or an equivalent method, be used to approximate the boundary condition. Given that Fig. 3.3 shows poor accuracy of derivatives (Lobatto and Chebyshev) for calculating fluxes, one might question the suitability of this approach for all but Gauss points.

Fig. 3.9 compares the profiles for solutions calculated with $n = 4$. Note that the boundary value of $y$ is about 0.3, indicating roughly 30% of the resistance is external. The relative accuracy of the methods appears similar to that shown in Fig. 3.1 with Dirichlet conditions. Fig. 3.10 shows the $L_2$ error norms are almost the same for Chebyshev and Gauss.
points and averages about 50 percent greater for Lobatto points. However, Fig. 3.11 shows that the disparity in fluxes (calculated using Eq. (3.13)) is significant. Although, the differences are not large for \( n < 6 \), the differences in convergence rate are significant. As discussed below, similar flux errors persist even when \( Bi \) is so large that a Dirichlet condition is approached. Since Eq. (3.3) shows that \( Bi \) does not change the shape of the profiles, there can be no reason for this slower convergence rate other than a flaw in the implementation of the method.

I first became aware of this problem with flux boundary conditions from Ferguson (1971). It was pointed out at roughly the same time by Elnashaie and Cresswell (1973). The problem is clearly evident for two different problems in Finlayson (1972) (see Table 5.7 and Fig. 5.7). For those aware of the problem, one solution is to use Gauss points whenever a flux boundary condition is encountered. Unfortunately, texts have perpetuated the use of boundary collocation, so many are unaware of the problem. A superior alternative method can be found by examining other Methods of Weighted Residuals (MWR) and the foundation of orthogonal collocation.

### 3.1.2 Moments Method

To solve Eq. (3.1) by the method of moments, the residual is weighted by \( x^k \) for \( k = 0, \ldots, n - 1 \); however, weighting by any linearly independent set of \( n \) polynomials through degree \( n - 1 \) will give identical results. For example, we could use the first \( n \) Legendre polynomials as is common with the Tau method of Lanczos (1956). The results with Legendre polynomials would be identical except possibly for rounding errors. Another suitable set of linearly independent polynomials are the Lagrange interpolating polynomials for only the \( n \) interior points. These polynomials are related to those in Eq. (3.7) by:

\[
\ell_i^*(x) = \ell_i(x) \frac{x_i(1-x_i)}{x(1-x)}
\]

(3.17)

where the asterisk indicates the reduced polynomial. Using these weight functions in Eq. (1.4) together with the residual function, Eq. (3.7), and integrating numerically, the problem becomes:

\[
\sum_{k=1}^{m} \left[ \left( \sum_{i=0}^{n+1} y_i \frac{d^2 \ell_i}{dx^2} \right) x_k \right] + r \left( x_k, \sum_{i=0}^{n+1} \ell_i(x_k)y_i \right) W_k \ell_j^*(x_k) = 0
\]

(3.18)

for \( j = 1, \ldots, n \), where \( x_k \) and \( W_k \) designate quadrature base points and weights, respectively, which are not yet specified. The method of Moments requires that all boundary conditions be satisfied exactly. For Dirichlet conditions, Eq. (3.3), the boundary values are substituted. For third kind conditions, Eq. (3.1b), Eq. (3.11) provides the two extra equations.

For \( m > n \) the quadrature base points are naturally different from the nodal interpolation points used to define the trial functions, Eq. (3.7). If \( m = n \), and the interpolation points correspond to the quadrature points, some wonderful simplifications occur. For this case, Eq. (3.18) simplifies to:
\[
\sum_{i=0}^{n+1} W_j B_{ji} y_i + W_j r(x_j, y_j) = 0
\]  

(3.19)

Eq. (3.19) is equal to Eq.(3.10) when each row is multiplied by the quadrature weight, \(W_j\), so the equations are equivalent. To make this approach work, we need a quadrature formula that will give an exact or accurate approximate integration of Eq. (3.18) with \(m = n\) and \(0 < x_i < 1\). The most accurate quadrature of this type is Gaussian quadrature. We now examine the accuracy of Gaussian quadrature for integration of Eq. (3.18).

The trial functions, \(\ell_i(x)\), are polynomials of degree \(n + 1\), so the second derivative is of degree \(n - 1\). The weight functions, \(\ell_j^*(x)\) are of degree \(n - 1\). Combining the two terms, the diffusion term is of order \(2n - 2\) and the first order reaction term is of order \(2n\). Since Gaussian quadrature is exact for polynomials through degree \(2n - 1\), integration of the diffusion terms is exact, while the source term misses exact integration by one degree. To achieve an exact representation of the Moments method, the following integration must be performed more accurately:

\[
D_{ji} = \int_0^1 \ell_j^*(x) \ell_i(x) \, dx \approx \delta_{ji} W_j
\]  

(3.20)

We call \(D\) the mass matrix and define \(D_{0,i} = D_{n+1,i} = 0\). With Gauss points it is approximated by the diagonal matrix of quadrature weights shown at the far right. In finite element methods, the reduction of the mass matrix to a diagonal one is called lumping. Lumping is achieved automatically here due to the approximate integration.

To list the complete set of equations, it is convenient to combine the boundary and interior equations by defining:

\[
C_{ji} = \delta_{j,n+1} A_{n+1,i} - \delta_{j,0} A_{0,i} - W_j B_{ji}
\]  

(3.21)

With this definition, the complete set of equation for the first order reaction with constant coefficients and third kind boundary conditions is:

\[
\delta_{j,0} 2By_0 + \delta_{j,n+1} 2By_{n+1} + \sum_{i=0}^{n+1} (C_{ji} + 4\varphi^2 D_{ji}) y_i = 4\varphi^2 \sum_{i=0}^{n+1} D_{ji} = 4\varphi^2 W_j
\]  

(3.22)

\(C\) is a symmetric matrix which we call the stiffness matrix. For the full Moments method the complete matrix problem is not symmetric because of \(D\), but with its diagonal approximation the complete system of equations is symmetric and positive definite. One deficiency of orthogonal collocation or pseudospectral methods is that self adjoint operators do no lead to symmetric matrix problems. This development shows that by a simple rescaling of the equations this desirable feature is achieved with Gauss points. A symmetric matrix problem cuts the calculations for solution almost in half (Fadeeva, 1959). For a full Moments method, more calculations are required because the mass matrix is not symmetric. Also, for nonlinear reaction terms, the mass matrix must be recalculated every Newton iteration. These extra calculations are rarely worth the effort. We discuss more complicated source terms below.
Lobatto quadrature with \( m = n \) has sufficient accuracy to integrate all the terms in Eq. (3.18) exactly for a first order reaction. However, it does not reduce to a collocation method because end points terms would appear in Eq. (3.19). These terms are zero with Gauss points, because the quadrature weights are zero on the boundaries. Orthogonal collocation at Lobatto quadrature base points bears no direct relationship to the Moments method. With Chebyshev points, the associated Clenshaw-Curtis quadrature also has nonzero quadrature weights at the endpoints. In addition, Clenshaw-Curtis quadrature is not accurate enough to produce a good approximation to the Moments method. For these reasons, it also bears no direct relationship to the Moments method.

The mass and stiffness matrices, \( D \) and \( C \) in Eqs. (3.20) and (3.21), are available from the computer codes described in Chapter 2.

### 3.1.3 Galerkin Method

To solve the problem with the Galerkin method, the residual is weighted by the trial functions \( \ell_j(x) \). Since the Robin boundary conditions reduce to Dirichlet conditions for large Bi number, we will consider only the more general conditions, Eq. (3.1b). The Galerkin method permits two different methods to treat boundary conditions involving derivatives. The boundary conditions may be satisfied exactly or they may be treated as *natural* boundary conditions (see for example Finlayson (1972)). The normal procedure with the orthogonal collocation or pseudospectral method is to satisfy the boundary conditions exactly, i.e. boundary collocation, Eq. (3.11). The short development in Appendix B shows the difficulties with this approach. For this reason, we will use the *natural* boundary condition treatment here. With this approach, the boundary condition is satisfied approximately along with the rest of the differential equation.

To solve the problem with the Galerkin method, the residual Eq. (3.7) is weighted by the trial functions \( \ell_j(x) \), for \( j = 0, \ldots, n + 1 \). The equations are converted to the *weak formulation* by integrating the second derivative term by parts:

\[
\sum_{i=0}^{n+1} \ell_j \left. \frac{d\ell_i}{dx} \right|_0^1 y_i - \int_0^1 \left( \sum_{i=0}^{n+1} \frac{d\ell_j}{dx} \frac{d\ell_i}{dx} y_i - \ell_j r(x,y) \right) dx = 0
\]

(3.23)

The first term contains the two boundary derivatives, so we substitute the boundary conditions and integrate the other terms using a suitable quadrature formula:

\[
\delta_{j,0} 2Bi y_0 + \delta_{j,n+1} 2Bi y_{n+1} + \sum_{k=1}^{m} W_k \left( \sum_{i=0}^{n+1} \frac{d\ell_j}{dx} \left. \frac{d\ell_i}{dx} \right|_{x_k} y_i - \ell_j(x_k) r(x_k,y(x_k)) \right) = 0
\]

(3.24)

The \( x_k \) in Eq. (3.24) designate the quadrature base points, which differ from the nodal interpolation points for \( m > n \). Since the trial functions are polynomials of degree \( n+1 \), the diffusion term is of degree \( 2n \). For a first order reaction the source term is of degree \( 2n+2 \). An \( n \) point Gaussian quadrature is exact through degree \( 2n-1 \), so neither term would be integrated exactly. On the other hand, Lobatto quadrature with \( n \) interior points gives exact
integration through degree $2n+1$, so it gives exact integration for the diffusion term, but misses exact integration of the reaction term by one degree. This is the same level of discrepancy found when the Moments method is approximated with Gaussian quadrature. Using Lobatto quadrature with $n$ interior points, Eq. (3.24) reduces to:

$$\delta_{j,0}2Bi y_0 + \delta_{j,n+1}2Bi y_{n+1} + \sum_{l=0}^{n+1} C_{jl} y_l - W_j r(x_j, y_j) = 0$$  \hspace{1cm} (3.25)$$

where:

$$C_{jl} = \sum_{k=0}^{n+1} W_k A_{kj} A_{kl} = \delta_{j,n+1} A_{n+1,l} - \delta_{j,0} A_{0,l} - W_j B_{jl}$$  \hspace{1cm} (3.26)$$

Since Lobatto quadrature can perform the integration by parts exactly, it follows that the stiffness matrix, $C$, can be calculated by either expression above. The right expression is identical to Eq. (3.21). Given this relationship, it is clear that at the interior points, Eq. (3.25) is identical to Eq. (3.19) and equivalent to Eq. (3.10), i.e. collocation. From this development and the development in Appendix B, we conclude collocation at Lobatto points is a close approximation of the Galerkin method, but only when a natural boundary condition treatment is used.

To examine the different treatment of the boundary conditions, we compare Eqs. (3.11) and (3.25) at the boundaries, $j = 0$ and $j = n + 1$. Using the left expression of Eq. (3.26) for $C$, the boundary equations are:

$$2Bi y_0 - \sum_{i=0}^{n+1} A_{0i} y_i - W_0 \left( \sum_{i=0}^{n+1} B_{0i} y_i + r(0, y_0) \right) = 0$$  \hspace{1cm} (3.27)$$

$$2Bi y_{n+1} + \sum_{i=0}^{n+1} A_{n+1,i} y_i - W_{n+1} \left( \sum_{i=0}^{n+1} B_{n+1,i} y_i + r(1, y_{n+1}) \right) = 0$$

Eq. (3.27) differs from (3.11) by the extra term on the right. It is the boundary quadrature weight multiplying the interior residual, Eq. (3.8), evaluated at the boundaries, $R(0,y)$ and $R(1,y)$. This procedure sets the combination of the boundary condition residual and the interior residual at the boundary to zero. Unlike boundary collocation, the boundary condition is not satisfied exactly, but both residuals will converge to zero at an exponential rate.

Note also that the right expression of Eq. (3.26) and Eq. (3.21) are identical for the calculation $C$, so Eqs. (3.25) and (3.27) are equally valid for Gauss points, since the quadrature weights are zero at the boundaries. The left expression of Eq. (3.26) is not valid for Gauss or Chebyshev points, because the quadrature is not accurate enough to perform the integration by parts exactly. For an infinite $Bi$ number, Eqs. (3.25) and (3.27) reduce to the Dirichlet conditions, $y_0 = y_{n+1} = 0$. 

[12]
Lobatto quadrature calculates the second derivative term exactly, but misses exact integration of the source term by one degree. For the first order reaction, the full Galerkin method requires a more accurate calculation of the mass matrix:

\[
D_{ji} = \int_0^1 \ell'_j(x) \ell'_i(x) \, dx \approx \delta_{ji}W_j
\]  

(3.28)

Collocation at Lobatto points approximates this matrix by a diagonal one composed of the quadrature weights as shown above on the right. Substituting a more accurate integration of the source term into the Galerkin method, Eq. (3.24), produces an equation identical in form to Eq. (3.22). For the Galerkin method, both the stiffness and mass matrices are symmetric. The mass matrix is full for the Galerkin method and is diagonal or lumped for collocation at Lobatto points. For more complex rate expressions, a full mass matrix adds complexity and calculations which do not normally improve the accuracy enough to warrant the extra calculations.

In summary, collocation with Lobatto quadrature base points and a natural treatment of flux boundary conditions accurately approximates the Galerkin method. In Section 3.1.2 shows collocation at Gauss points is an accurate approximation of the Moments method. Collocation at Radau points is in between and reduces to boundary collocation at one boundary and a natural treatment at the other.

The Clenshaw-Curtis quadrature is not accurate enough to give a good approximation to either the Galerkin or Moments method. However, the Chebyshev points and quadrature weights are between Gauss and Lobatto points and weights (see Fig. 1.1) which gives some confidence that they should work. Their use is supported primarily by approximation theory and not integrated MWR. It would be beneficial if the Chebyshev flux errors shown in Fig. 3.11 could be reduced. We propose that Chebyshev points also use Eq. (3.25) with \( c \) calculated by Eq. (3.21) or the right expression of Eq. (3.26). We will call this a natural boundary condition treatment also even though the quadrature does not produce an accurate approximation of the Galerkin method.

Formulations using the stiffness matrix and a natural treatment of flux boundary conditions are referred to as weak formulations. Unlike with Gauss and Lobatto points, the stiffness matrix for Chebyshev points is not symmetric (except for \( n < 4 \)). Consequently, almost twice the computations are required to solve matrix problems. Also note the natural boundary condition treatment causes boundary rate terms to appear in the approximation, Eq. (3.27). For a nonlinear rate expression, all \( n + 2 \) equations are nonlinear, whereas with Gauss points the two boundary equations are linear. These linear equations can be eliminated initially so only \( n \) nonlinear equations must be solved iteratively.

The relationship between collocation at Lobatto points and the Galerkin method is what motivated Villadsen and Stewart (1967) to select Lobatto points. However, they considered only Dirichlet conditions. The natural boundary condition treatment for the Galerkin method is
advocated by Finlayson and Scriven (1966). They point out this treatment sets a combination of boundary and interior residuals to zero, as in Eq. (3.27). The natural treatment is standard in most finite element applications. The use of this procedure with collocation at Lobatto points was outlined by Young (1977) and described more fully by Funaro (1988). Canuto (1986) discussed an apparently similar procedure for Chebyshev and Lobatto points. This early work notwithstanding, the natural treatment has not found wide spread use and more recent texts continue to recommend boundary collocation.

**Linear Source, Constant Coefficients, Third Kind B.C., Galerkin/Moments.** Fig. 3.12 shows the flux errors from Fig. 3.11 updated with results from full Galerkin and Moments methods and Lobatto and Chebyshev collocation with a natural treatment of the boundary conditions. These results are labeled “Nat.” while the boundary collocation results are labeled “Coll.”.

The improvement by using natural boundary conditions is impressive, especially with Lobatto points. With the natural boundary condition treatment, the convergence rates for the three choices of point are similar to those in Fig. 3.3. Since the problem reduces to the Dirichlet problem for large $Bi$, the effect of this parameter was investigated by solving the problem with $Bi = 2, 5, 10$ and 50 for comparison. With the largest value, $7 \times 10^3$ for $Bi = 2$ to $3 \times 10^2$ for $Bi = 50$. The convergence rates are similar, so the disparity grows with $n$. Clearly, the error with boundary collocation can be several orders of magnitude greater even for quite large values of $Bi$. The errors with Chebyshev points are also reduced by using a natural boundary condition treatment, but to a lesser extent, but still almost an order of magnitude at $n = 8$.

As discussed above, the natural boundary condition treatment, Eq. (3.27), sets a combination of the boundary condition and the boundary value of the interior residual to zero. Neither residual will be identically zero, but they converge to zero at an exponential rate. Fig. 3.13 shows the convergence behavior of these two residuals as a function of $n$ for $Bi = 10$. Graphs for the other values of $Bi$ are very
similar. As shown in Eq. (3.27), the ratio of the two residuals is equal to the boundary quadrature weight, \( W_0 \) and \( W_{n+1} \) which are \( O(1/(2n^2)) \) for Lobatto quadrature and half that for Clenshaw-Curtis quadrature (Chebyshev points).

We also note that in Fig. 3.12 the Galerkin and Moments methods improve in a stair step fashion. When \( n \) is even, the results with Galerkin and Lobatto points are identical and the results with Moments and Gauss points are identical. This result is an artifact caused by the symmetry of the solution about \( x = 0.5 \) for this particular problem. For example, with \( n = 4 \) the trial solution is a 5th order polynomial, but the highest order term has a zero coefficient because of the symmetry. Lobatto points normally miss exact integration of the Galerkin method by one degree, but since the problem symmetry knocks out the highest degree the two methods agree. With \( n = 3 \) the polynomial is 4th order, so the Galerkin method is just as good as when a 5th order term is included. An analogous situation applies for the Moments method and collocation at Gauss points.

When the differential equation contains values of the dependent variable (\( y \) in this case), not exclusively its derivatives, this very special set of circumstances, i.e. linear, constant coefficient and symmetric solution, are the only times when collocation at Lobatto points is identical to a Galerkin method and collocation at Gauss points is identical to the Moments method. If the source \( r(x,y) \) in Eq. (3.1) is dependent only on \( x \) and is a polynomial of degree \( n \) or less, Lobatto points is identical to the Galerkin method and Gauss points is identical to the Moments method. Other conditions can easily be evaluated by determining the degree of polynomial which must be integrated. Usually, Gauss and Lobatto points give approximations to the Moments and Galerkin methods, but often very good ones. Some texts incorrectly state that collocation at Gauss points replicates the Galerkin method for more general conditions [Boyd (2000), p. 89]. This example problem can be treated much more efficiently by using polynomials in \( x^2 \), which we discuss below.

### 3.1.4 Mass Conservation and Fluxes

One is usually interested in the flux at the boundaries. For example, if a fluid were flowing on both sides of the slab we would want to know the rate of mass or heat transfer from the slab. For the symmetric problem, the transfer is quantified by a single normalized flux, \( \eta \) given by Eqs. (3.4) and (3.5). For a nonsymmetric problem one would generally want the breakdown of left and right side fluxes, while in multiple dimensions the flux profiles along the boundaries may be important. The average energy equation or divergence theorem gives only the total flux. If the sum of the individual fluxes (or integral of the boundary profiles in multiple dimensions) equals the total flux, this gives us greater confidence in the individual values.

The total of the fluxes will give the average rate provided the method conserves mass. Eq. (3.5), the divergence theorem or average energy equation is just an overall balance. It is derived by integrating Eq. (3.1) across the domain. In general, a method will be conservative if
the integral of the residual is zero. If the MWR weights \( w_i(x) \) in Eq. (1.4)) contain unity in some combination, the method will be conservative.

Moments and collocation at Gauss points are conservative regardless of the formulation. If formulated with monomials or Legendre polynomials the first weight function would be \( x^0 \) or \( P_0 = 1 \). With our formulation from Section 3.1.2, \( \sum \ell'_i(x) = 1 \). Since these methods are conservative, Eq. (3.5) is obeyed when the boundary fluxes are calculated by differentiation of the approximate solution, Eq. (3.14). This explains why the errors in Fig. 3.3 are the same with either method of calculation for collocation at Gauss points.

For the Galerkin method and collocation at Lobatto points, the sum of all the Lagrange interpolating polynomials is also unity. However, if we consider Dirichlet boundary conditions, the boundary values are directly substituted into Eq. (3.7) so the first and last interpolating polynomials are not used as weight functions. The method appears not to be conservative due to the left over terms on the right side below:

\[
\frac{dy}{dx}\bigg|_0 + \int_0^1 r(x, y) \, dx = \int_0^1 [\ell_0(x) + \ell_{n+1}(x)] R(x, y) \, dx
\]  
(3.29)

Using quadrature, Eq. (3.29) is approximated by:

\[
\frac{dy}{dx}\bigg|_0 + \int_0^1 r(x, y) \, dx = W_0 R(0, y) + W_{n+1} R(1, y)
\]  
(3.30)

The two terms on the right side are those needed to correct the fluxes in the natural boundary condition treatment, Eq. (3.27), so the correction makes the method conservative. To be consistent with the Galerkin method, the individual fluxes should be approximated by:

\[
\frac{dy}{dx}\bigg|_{x=0} = \sum_{i=0}^{n+1} A_{0i} y_i + W_0 \left( \sum_{i=0}^{n+1} B_{0i} y_i + r(0, y_0) \right)
\]  
(3.31)

\[
\frac{dy}{dx}\bigg|_{x=1} = \sum_{i=0}^{n+1} A_{n+1,i} y_i - W_{n+1} \left( \sum_{i=0}^{n+1} B_{n+1,i} y_i + r(1, y_{n+1}) \right)
\]

The same equations written with the stiffness matrix are:

\[
\frac{dy}{dx}\bigg|_{x=0} = - \sum_{i=0}^{n+1} C_{0i} y_i + W_0 r(0, y_0)
\]  
(3.32)

\[
\frac{dy}{dx}\bigg|_{x=1} = \sum_{i=0}^{n+1} C_{n+1,i} y_i - W_{n+1} r(1, y_{n+1})
\]

When computed from these equations, the two fluxes will be consistent with Eq. (3.5), the average energy equation or divergence theorem. For the full Galerkin method, the fluxes must be calculated with the equivalent expression using the mass matrix, \( D \). For the Robin boundary conditions the expressions are consistent with Eq. (3.1b), so it would be far simpler to calculate fluxes by multiplying \( 2Bi \) by the boundary values of \( y \). However, this calculation is subject to roundoff errors when \( Bi \) is large.
Note that Eqs. (3.31) and (3.32) are also valid with Gauss points, since the boundary quadrature weights are zero. This equivalence is useful when writing one computer code which will work with either type of points, provided one is not averse to a few unnecessary calculations. Using these equations with Chebyshev points will also make that method conservative, which provides some additional justification for using a natural treatment with Chebyshev points.

The relationship shown above, between the overall balance and natural boundary condition treatment was noted by Finlayson and Scriven (1966). Ferguson (1971) proposed approximation of flux boundary conditions using the integral rather than the derivative in Eq. (3.5). Ferguson’s procedure has some limitations, but is equivalent to a natural treatment when it is applicable. The natural treatment is equivalent, more general and easier to implement.

**Linear Source, Variable Coefficients, Dirichlet B.C.** To make the problem nonsymmetric and more interesting, consider the case of a first order source, but with a coefficient which varies with position according to:

\[
\hat{r}(x, y) = (0.2 + 1.6x^2(3 - 2x))(1 - y) = q(x)(1 - y)
\]  

(3.33)

The spatial variation, \(q(x)\), goes from 0.2 on the left edge to 1.8 on the right edge, with an average value of 0.5 on the left half and 1.5 on the right half giving an overall average of 1.0. For a given value of \(\varphi\), the average rate constant is the same as with Eq. (3.2) for \(k = 1\). Fig. 3.14 shows solutions to this problem for \(\varphi = 5\) with collocation at Gauss, Chebyshev, and Lobatto points. The solution with Lobatto points is again slightly more accurate than with the other two choices. In Fig. 3.14, the “exact” solution is one computed with large \(n\) which is exact for practical purposes.

When the problem is solved with the Galerkin or Moments method a more accurate mass matrix must be calculated. For many problems, especially nonlinear ones, an exact calculation would be cumbersome and is not necessary. Approximate calculations with quadrature formulas are most often used. Orthogonal collocation is a special case of approximate quadrature, when the interior quadrature points and collocation points coincide. Another common procedure is to interpolate variable or nonlinear terms into the trial space, i.e.:

\[
r(x, y) \approx \sum_{i=0}^{n+1} \ell_i(x)r(x_i, y(x_i))
\]  

(3.34)

For the Moments method, these various approaches give:

\[
D_{ji} = \int_0^1 \ell_j^*(x)\ell_j(x)q(x) \, dx \approx \sum_{k=1}^m W_k \ell_j^*(x_k)\ell_j(x_k)q(x_k) \quad \text{or}
\]

\[
\approx q(x_i)\int_0^1 \ell_j^*(x)\ell_j(x) \, dx \quad \text{or}
\]

\[
\approx q(x_j)W_j \delta_{ij}
\]  

(3.35)
Where the asterisk denotes the reduced functions defined by Eq. (3.17) and $D_{0i} = D_{n+1,i} = 0$ for the Moments method. For the Galerkin method the trial functions, $\ell_j(x)$, are used instead of the reduced functions. The approximation with quadrature is given on the first line, while the approximation for interpolation into the trial space, Eq. (3.34), is given on the second line. Finally, the approximation using collocation is given on the last line. Once calculated, the mass matrix is substituted into Eq. (3.22). The collocation approximation reduces to Eq. (3.25). As $Bi \to \infty$ the equations reduce to specified boundary values, i.e. Dirichlet conditions.

When the mass matrix is calculated with quadrature, one must select the quadrature formula and the number of quadrature base points, $m$ in Eq. (3.35). We select Gaussian quadrature for the Moments method and Lobatto quadrature for the Galerkin method. In the discussion above for the constant coefficient problem, when comparing collocation at Gauss points with the Moments method and collocation at Lobatto points with the Galerkin method, we found collocation missed exact integration by one degree. The variable coefficient for this example, $q(x)$ in Eq. (3.33), is degree 3. For the Galerkin method, both the trial functions and weight functions are of degree $n + 1$, while the weight functions are $n - 1$ degree for the Moments method. Table 3.1 summarizes the degree of the mass matrix integrands in Eq. (3.35) and compares it to the accuracy of Gaussian and Lobatto quadrature. With the cubic variation, both miss exact integration by four degrees, so to achieve exact integration with quadrature requires in each case, two additional quadrature points.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
 & without spatial variation & interpolated source term & with spatial variation, Eq. (3.33) \\
\hline
Gaussian Quadrature & $2n - 1$ & $2n - 1$ & $2n - 1$ \\
\hline
Moments Method & $2n$ & $2n$ & $2n + 3$ \\
\hline
Lobatto Quadrature & $2n + 1$ & $2n + 1$ & $2n + 1$ \\
\hline
Galerkin Method & $2n + 2$ & $2n + 2$ & $2n + 5$ \\
\hline
\end{tabular}
\caption{Integrand Degree for Mass Matrix Calculation, Eq. (3.35)}
\end{table}

This example provides a good test of the integration accuracy requirements, because with only one additional quadrature point, the integration is still two degrees shy of exact. So, besides collocation and the Galerkin method there are two additional ways to approximate $D$: (1) interpolation of the source terms, Eq. (3.34) and (2) approximate integration with $n + 1$ interior points.
quadrature points. Similarly, there are two approximate methods of calculating \( D \) for the Moments method. All of these possibilities give a large volume of results for this problem.

Fig. 3.14 shows the calculated results with collocation using the three choices of collocation points. Fig. 3.15 compares collocation at Gauss points with the Moments method using both an exact mass matrix and interpolated rates. Fig. 3.16 compares collocation at Lobatto points with the Galerkin method. For the Galerkin method, the other approximations were not distinguishable from those in Fig. 3.16.

Table 3.2 shows results of the flux calculations for this problem when \( n = 4 \). For the results in the table, "\( n = 5 \)" indicates the number of interior quadrature points used in the calculation of the mass matrix and "interpolated r" indicates those for which Eq. (3.34) was used. In these results Eq. (3.31) or (3.32) was used to calculate the fluxes for Lobatto or Chebyshev points, or the equivalent method for the Galerkin method. The table results labeled, "derivative",

<table>
<thead>
<tr>
<th></th>
<th>Flux left</th>
<th>Flux right</th>
<th>Flux Total</th>
<th>Error left</th>
<th>Error right</th>
<th>Error total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact</td>
<td>0.05062</td>
<td>0.13368</td>
<td>0.18429</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gauss Collocation</td>
<td>0.05013</td>
<td>0.12097</td>
<td>0.17110</td>
<td>-0.96%</td>
<td>-9.51%</td>
<td>-7.16%</td>
</tr>
<tr>
<td>Chebyshev Collocation</td>
<td>0.05026</td>
<td>0.12910</td>
<td>0.17936</td>
<td>-0.71%</td>
<td>-3.42%</td>
<td>-3.60%</td>
</tr>
<tr>
<td>Lobatto Collocation</td>
<td>0.05073</td>
<td>0.13742</td>
<td>0.18814</td>
<td>0.21%</td>
<td>3.80%</td>
<td>3.09%</td>
</tr>
<tr>
<td>Moments</td>
<td>0.04902</td>
<td>0.12598</td>
<td>0.17500</td>
<td>-3.16%</td>
<td>-5.76%</td>
<td>-5.04%</td>
</tr>
<tr>
<td>Moments, interpolated r</td>
<td>0.04623</td>
<td>0.12473</td>
<td>0.17097</td>
<td>-8.66%</td>
<td>-6.69%</td>
<td>-7.23%</td>
</tr>
<tr>
<td>Moments, ( n = 5 ) Gauss</td>
<td>0.04903</td>
<td>0.12654</td>
<td>0.17557</td>
<td>-3.14%</td>
<td>-5.34%</td>
<td>-4.73%</td>
</tr>
<tr>
<td>Galerkin</td>
<td>0.05053</td>
<td>0.13561</td>
<td>0.18614</td>
<td>-0.17%</td>
<td>1.45%</td>
<td>1.00%</td>
</tr>
<tr>
<td>Galerkin, interpolated r</td>
<td>0.05070</td>
<td>0.13601</td>
<td>0.18671</td>
<td>0.16%</td>
<td>1.75%</td>
<td>1.31%</td>
</tr>
<tr>
<td>Galerkin, ( n = 5 ) Lobatto</td>
<td>0.05055</td>
<td>0.13583</td>
<td>0.18638</td>
<td>-0.13%</td>
<td>1.61%</td>
<td>1.13%</td>
</tr>
<tr>
<td>Chebyshev, derivative</td>
<td>0.04795</td>
<td>0.11116</td>
<td>0.15911</td>
<td>-5.27%</td>
<td>-16.84%</td>
<td>-13.66%</td>
</tr>
<tr>
<td>Lobatto, derivative</td>
<td>0.04666</td>
<td>0.10497</td>
<td>0.15163</td>
<td>-7.82%</td>
<td>-21.47%</td>
<td>-17.73%</td>
</tr>
<tr>
<td>Galerkin, derivative</td>
<td>0.03953</td>
<td>0.10974</td>
<td>0.14927</td>
<td>-17.21%</td>
<td>-16.93%</td>
<td>-17.00%</td>
</tr>
</tbody>
</table>
indicates Eq. (3.14) was used to calculate the fluxes. From Fig. 3.15 it does not appear that the additional complexity of the Moments method and a full mass matrix, $D$, adds any substantial accuracy to the results. However, Table 3.2 indicates there is some improvement, but the improvement is spotty. Some flux errors are larger, especially those at $x = 0$, which is the “easy” side due to the more gentle profile. Fig. 3.16 indicates that relative to Lobatto collocation, the Galerkin method gives a small improvement to the profiles. Table 3.2 shows that with the Galerkin method, the right side and total errors are reduced by about a factor of 3. Also, a Galerkin approximate mass matrix with 5 interior points is very nearly as good as an exact mass matrix.

Fig. 3.17 shows the $L_2$ error norms versus $n$. As before, the $L_1$ and $L_2$ errors are relatively insensitive to differences between the methods. The ratio of best to worst error averaged over all $n$ from 2 to 16 is only about 2. The ratio of the average error relative to the Galerkin method for Lobatto, Chebyshev and Gauss collocation is 1.02, 1.36 and 2.03, respectively.

Fig. 3.18 shows the error of the flux on the right side, $x = 1$, for all the methods. The graph is busy due to all the results, but it is worthwhile to examine the effect of approximate and exact integration for the Galerkin and Moments methods, since Gauss and Lobatto collocation are also approximations. In this graph, the results labeled “Interp” are for the interpolated source terms, Eq. (3.34) and those labeled “$n+1$” use one additional quadrature point to give a full but approximate mass matrix. “Der” is used to indicate fluxes estimated by the first derivative. The other methods calculate fluxes using Eq. (3.31) or (3.32) (or equivalent for Galerkin).

Again, the results calculated from the first derivative are much less accurate except for the Gauss-collocation and Moments methods for which they are identical. All the evidence shows that for other methods derivatives, Eq. (3.14), should never be used to estimate fluxes. If the reader takes this one result to heart, writing this monograph will have been worthwhile.
Fig. 3.18 shows that the Galerkin method is generally the best method, but frequently the approximate mass matrices give results that are just as good. Exact integration of the mass matrix gives no clear improvement for the Moments method either. Due to the large number and variability of the results in Fig. 3.18, they are further summarized in Table 3.3. Table 3.3 lists the ratio of the error for the individual method relative to the Galerkin method geometrically averaged for \( n = 2 \) through 16. There could be other ways to summarize these results, but this method yields several clear conclusions. The full Galerkin and Moments methods are about twice as accurate as the respective collocation counterparts (Lobatto and Gauss points).

Interpolation of the source terms adds only a slight improvement over collocation. One quadrature point greater than collocation is on the average slightly better than an exact Galerkin or Moments method. This type of behavior has been observed in finite element methods (Strang and Fix (1973)). If one considers the computational effort, collocation will invariably win. However, if a full Galerkin or Moments method is desired it is wasteful to get carried away with extremely accurate integration.

The Gauss/Moments methods give errors that are about 8 times larger than the Lobatto/Galerkin counterparts. This factor is approximately the amount the error decreases with each increment of \( n \), so the Gauss methods require roughly one additional point for equivalent accuracy. In compensation for this difference, the treatment of flux boundary conditions is simpler and more intuitive, Eq. (3.11) rather than (3.25) or (3.27), and the usually nonlinear source terms appear only at the \( n \) interior points. The simplicity of using boundary collocation is one reason for the popularity of Gauss points. The natural boundary condition treatment required to achieve good accuracy with the other choices has not generally been considered previously.

Relative to the other methods, Chebyshev collocation is not competitive when flux calculations are carried out as normally recommended. With the improved flux calculations, Eq. (3.31) or (3.32), the results are good for small \( n \), but suffer from a lower convergence rate for \( n > 6 \). When compared to the other choices, the case supporting the use of Chebyshev points is:

- \( x \) and \( W \) can be efficiently calculated
- theoretical support from approximation theory
- justified by point distribution relative to other methods

On the other side of the argument:

- efficiency of \( x \) and \( W \) calculation relatively unimportant
- poor approximation to integrated MWR methods (Galerkin, Moments)
- convergence rates sometimes slower than with Gauss or Lobatto points
- less efficient calculations due to nonsymmetric matrices

The argument concerning point distributions is an interesting one. Many view the collocation point selection problem strictly from the standpoint of point distributions. Since Chebyshev points are between Gauss and Lobatto points (see Fig. 1.1), the results should be intermediate between them. If we compare the profiles and error norms that argument holds up, but it falls apart when comparing fluxes. The other characteristic to consider is that Gauss and Lobatto quadrature are both accurate for $O(2^n)$, while Clenshaw-Curtis quadrature is accurate for $O(n)$, see Figs. 3.8, 2.8 and 2.9. This difference in accuracy is the reason Chebyshev collocation is not a good approximation of other integrated MWR. It appears that the differences are not so important for small $n$, but perhaps the quadrature accuracy affects the convergence rate for fluxes at large $n$. Further analysis on this subject is warranted.

Proponents for the use of Chebyshev points cite the greater efficiency for computing the fundamental approximations, $x$ and $\mathbf{W}$. This advantage is irrelevant for most problems, but could become important for very large $n$, say $n > 100$. “Back in the day”, we computed the arrays once and punched the results out on cards, which were then added to the deck used to solve applications. A modernized version of this technique could certainly be used today, requiring no calculation. Also, one can make the argument that if a problem requires $n > 10$, finite element trial functions will likely be more efficient in most cases. One can base these methods on orthogonal collocation and construct such methods with any desired order of accuracy.

Many authors bemoan the lack of symmetry for collocation or pseudospectral approximations for self adjoint operators. When conventionally formulated, all methods suffer from this deficiency. However, a simple rescaling of the equations with the quadrature weights makes the method symmetric for Gauss and Lobatto points, see Eq. (3.19) and (3.25). The number of arithmetic operations to solve a symmetric matrix problem is roughly half that for a nonsymmetric problem (Fadeeva, 1959). There are theoretical and computational advantages for other problems as well, e.g. eigenvalue problems for parabolic equations in Chapter 4. These computational advantages tend to offset any advantages of Chebyshev points.

### 3.1.5 Symmetric Problems

Consider now the diffusion/conduction problem similar to Eq. (3.1), but for which the solution is symmetric about the centerline, $x = \frac{1}{2}$. We renormalize the coordinates to place the centerline at $x = 0$, while the outer boundary remains at $x = 1$. The governing equation for planar, cylindrical and spherical geometry ($\gamma = 0,1,2$ respectively) is:

$$\frac{1}{x^\gamma} \frac{d}{dx} \left( x^\gamma \frac{dy}{dx} \right) + r(y) = 0$$

(3.36)

with
\[ \frac{dy}{dx} \bigg|_{x=1} + Bi \, y(1) = 0 \quad \text{and} \quad \frac{dy}{dx} \bigg|_{x=0} = 0 \quad (3.37) \]

By using symmetric trial functions, the condition at \( x = 0 \) is automatically satisfied. For planar geometry, \( \gamma = 0 \), Eq. (3.36) is equivalent to Eq. (3.1) and (3.1b) when the source is not dependent on \( x \). The third kind or Robin condition is used at \( x = 1 \), since it reduces to a Dirichlet condition when \( Bi \) goes to infinity. Here we consider nonlinear source functions in addition to different geometries. Since the solution will be symmetric, we use trial functions which are Lagrange interpolating polynomials in \( x^2 \), Eq. (1.7):

\[ y(x^2) = \sum_{i=1}^{n+1} y(x_i) \ell_i(x^2) \quad (3.38) \]

Although the application of a Method of Weighted Residuals (MWR) is fundamentally the same, the geometry and symmetric trial functions cause some subtle differences to occur. Substituting the trial functions, the residual for the problem is:

\[ R(x, y) = \sum_{i=1}^{n+1} y_i \frac{1}{x^\gamma} \frac{d}{dx} \left( x^\gamma \frac{d \ell_i(x^2)}{dx} \right) + r(y(x^2)) = 0 \quad (3.39) \]

With the collocation method, the residual is set to zero at the interior collocation points, \( j = 1, \ldots, n \) and the boundary condition provides the final equation. The usual recommendation is to apply boundary collocation, but from the previous examples (see Fig. 3.12) we found this to be the best approach only for Gauss points. A natural boundary condition treatment is better in general. Using conventional collocation at interior points and a natural boundary condition treatment the equations are:

\[ \sum_{i=1}^{n+1} B_{ji} y_i + r(y_i) = 0 \quad (3.40) \]

at the interior points, i.e. \( j = 1, \ldots, n \) and from the boundary condition:

\[ Bi \, y_{n+1} + \sum_{i=1}^{n+1} A_{n+1,i} y_i - W_{n+1} \left( \sum_{i=1}^{n+1} B_{n+1,i} y_i + r(y_{n+1}) \right) = 0 \quad (3.41) \]

Eq. (3.41) sets the combination of the residual of the boundary condition and of the differential equation to zero at the boundary. It also insures the method is conservative as discussed in Sec. 3.1.4. The following matrix operators are defined as before:

\[ A_{ji} = \left. \frac{d \ell_i(x^2)}{dx} \right|_{x_j} \quad \text{and} \quad B_{ji} = \frac{1}{x^\gamma} \frac{d}{dx} \left[ x^\gamma \frac{d \ell_i(x^2)}{dx} \right]_{x_j} \]

Eq. (3.42) is not symmetric even for the self adjoint operator in Eq. (3.36). However, as shown in Sections 3.1.2 and 3.1.3 a minor restructuring of the equations into its weak form produces a symmetric matrix problem:

\[ \delta_{j,n+1} Bi y_{n+1} + \sum_{i=1}^{n+1} C_{ji} y_i - W_j r(y_j) = 0 \quad (3.42) \]

where:
When the stiffness matrix, $C$, is calculated with the left equality in Eq. (3.43) it is clear that Eq. (3.42) is equal to Eq. (3.40) after each row is multiplied by the quadrature weight, $W_j$. The equality on the far right of Eq. (3.43) follows from integrating the Laplacian, $B$, by parts as is done to convert a Galerkin method to weak form, e.g. Eq. (3.24). Since the integrand is a polynomial of degree $2n-1$ in $x^2$, the equality is valid for both Gauss and Lobatto quadrature, but not for the Clenshaw-Curtis quadrature used for Chebyshev points. For Chebyshev points, $C$ must be calculated using the left equality. The resulting stiffness matrix is not symmetric for $n > 2$.

Since orthogonal collocation is known to produce good results when it closely approximates the Moments or Galerkin methods, we compare it to these methods for this symmetric problem. For the Moments method, weighting the residual by the Lagrange interpolating polynomials through only the interior points (like Eq. (3.17)) is equivalent to weighting by monomials in $x^2$. For the Galerkin method, the weight functions are the same as the trial functions in Eq. (3.38). The trial functions and the weight functions are both degree $n$ for the Galerkin method, while the weight functions are degree $n - 1$ for the Moments method. The integrands for the second order term are degree $2n - 1$ and $2n - 2$ for Galerkin and Moments methods respectively. Lobatto and Gaussian quadrature are exact for degree $2n$ and $2n - 1$, respectively, so the second order term is always integrated exactly with those methods. For a simple first order source term, $k = 1$ in Eq. (3.2) or if the source term is interpolated like Eq. (3.34), the integrands for the source terms are degree $2n$ and $2n - 1$ for Galerkin and Moments methods, respectively. Lobatto quadrature is exact for the Galerkin method and Gaussian quadrature is exact for the Moments method.

A nonlinear reaction term could be approximated with a greater number of quadrature base points, and this may lead to some improvement in the solution. The results from the variable coefficient problem, see Fig. 3.18, suggest this would not be worth the extra effort.

The results for the symmetric problem are similar to those found for nonsymmetric problems, Secs. 3.1.2 and 3.1.3, i.e. Gauss points yield a good approximation to the Moments method, while Lobatto points accurately approximate the Galerkin method. However, for nonsymmetric problems we found that in each case the corresponding quadrature formula to be one degree shy of the accuracy needed for exact integration of the source terms, even for a linear problem, so this result for symmetric problems is slightly better. This analysis provides further explanation for the exact correspondence between Moments, Galerkin and collocation methods for the problem whose results are shown in Fig. 3.12 with an even number of points.

In order to achieve the accuracy stated above for Gaussian and Lobatto quadrature, the geometric parameter, $\gamma$, must be taken into account. Chapter 2 explains how the Jacobi
Polynomial roots and quadrature weights are determined and some examples are shown in Chapters 1 and 2. The collocation points are shifted closer to the boundary for cylindrical and spherical geometry (see Figs. 1.2, 2.6 and 2.7).

The justification for the use of Chebyshev points relies more on approximation theory than on accurate approximations of integrated MWR. It appears that the Chebyshev points are not normally altered to better suit the geometry (Trefethen (2000), p. 115; Boyd (2000), p. 380). For our purposes, we simply use the right half of the points used for a nonsymmetric problem. Chapter 2 describes the calculation for the interpolatory quadrature weights for cylindrical and spherical coordinates. The second derivative matrix operator, $B$, is calculated the same way as for other points (see Chapter 2) and the stiffness matrix, $C$, is calculated using the left expression in Eq. (3.43).

The source term used for the examples which follow is of the form:

$$\hat{r}(y) = \frac{(1 - y)^k}{(1 - K_a y)^2}$$  \hspace{1cm} (3.44)

where we note $\hat{r}(0) = 1$. When the denominator term, $K_a = 0$, the equation is identical to Eq. (3.2). When $K_a > 0$ and $k = 1$, the source function exhibits some interesting nonlinear behavior. It is said to be *autocatalytic*, i.e. the rate can increase with an increase in $y$.

**kth Order Source, Dirichlet B.C., Various Geometry:** Here we consider Eq. (3.36) with source term Eq. (3.44) with $K_a = 0$. The normalized flux or effectiveness factor, $\eta$, defined in Eqs. (3.4) and (3.5) is again the quantity of primary interest from the solution. As described at the beginning of Section 3.1, $\eta$ is unity for small $\phi$ and asymptotic for large $\phi$. This behavior is qualitatively the same for all source functions and geometries. The asymptotic behavior corresponds to the condition where the reaction occurs in a layer near the external boundary and all reactants are consumed before reaching the center. Under these conditions there is an asymptotic solution for Dirichlet boundary conditions, $Bi \to \infty$, which gives $\eta$ in terms of a simple integral of the source function (Rawlings and Ekerdt, 2015).

It can also be shown that for different geometries, the scaling of $\phi$ and $Bi$ by the geometric factor, $(\gamma + 1)$, produces results that correspond at the extremes of $\phi$. This scaling factor is equivalent to using a characteristic length equal to the ratio of the particle volume to its surface area.

For the Dirichlet boundary conditions all source functions and geometries have the same asymptotic behavior when correlated using the generalized parameter, $\phi^*$, defined by:

$$\phi^* = \frac{\phi}{\gamma + 1} \left[ 2 \int_0^1 r(\hat{y}) d\hat{y} \right]^{-\frac{1}{2}} = \phi \frac{\sqrt{2(k + 1)}}{2(\gamma + 1)}$$  \hspace{1cm} (3.45)
where \( k \) is the order or exponent of the source function in Eq. (3.44) \((K_x = 0)\). Fig. 3.19 shows the normalized flux for a first order source, \( k = 1 \), planar and spherical geometry, together with numerical solutions for \( n = 2 \).

The results for planar geometry with \( \varphi^* = 5 \) in Fig. 3.19 correspond to those displayed in Figs. 3.1 through 3.3. This symmetric treatment with \( n = 2 \) corresponds to the nonsymmetric treatment with \( n = 4 \) in Fig. 3.1. This value of \( \varphi^* \) corresponds to the point where the numerical results with \( n = 2 \) begin to depart from the exact solution. For smaller values of \( \varphi^* \) the results agree reasonably well with the analytical solutions. For spherical geometry, \( \varphi = 3\varphi^* \), so the problem is correspondingly more difficult and the departures of the numerical results from the analytical solutions occur at lower values of \( \varphi^* \).

Based on the results in Fig. 3.19 one would conclude that Chebyshev points are better, which seems to conflict with the results in Fig. 3.3. Closer examination reveals the apparent difference is due to the scale of the graphs. For reference, the maximum difference between the analytical results for planar and spherical geometry is about 15%. The numerical errors for planar geometry at \( \varphi^* = 5 \) are not distinguishable from the exact curve in Fig. 3.19. As discussed in Section 3.1.1, for \( \varphi^* = 5 \) the errors are 0.8, -3.1, and -4.3 percent for Lobatto, Chebyshev and Gauss points, respectively. For smaller \( n \), the Chebyshev results in Fig. 3.3 are the most accurate, but the flux errors are greater than 1 percent and there are significant errors in the profiles. However, for a relatively loose error tolerance Chebyshev points are characteristically more accurate for these problems, in addition to Fig. 3.3 see Figs. 3.12 and 3.18.

Now consider a second order source, \( k = 2 \), still with \( K_x = 0 \). Fig. 3.20 shows calculated profiles for \( n = 4 \) for spherical geometry. The errors near the center of the sphere are significant for Gauss and Lobatto points. This result is likely because the Galerkin and Moments methods shift the points away from the center where the volume is small. Chebyshev points are not based on integrated MWR, so the point locations remain the same, regardless of the geometry. Although there are some excursions between the nodes, the errors at the collocation points are characteristically small. As stated above, some finite element methods exhibit a phenomenon called superconvergence, where the accuracy at some nodes is greater than that of the solution in general. I am not aware of any studies of this phenomenon for
Despite the errors in the profiles in Fig. 3.20, the Chebyshev points give a flux error of 1.7 percent which is virtually the same as the error with Gauss points and twice the error given by Lobatto points. Fig. 3.21 shows the flux error as a function of \( n \) for this problem and the corresponding one with planar geometry. The relative performance of the methods is similar to that shown in Figs. 3.3, 3.12 and 3.18. The results suggest that for comparable accuracy spherical problems require about 1.6 to 1.7 times as many points as planar geometry when compared for a given accuracy and value of \( \phi^* \). For this nonlinear problem, the errors for a given number of points is not substantially worse than for the linear first order source function. For example with planar geometry, \( \phi^* = 5 \) and \( n = 2 \), Lobatto, Chebyshev and Gauss points give respectively, flux errors of 0.8, -3.1 and -4.3 percent for \( k = 1 \) and 3.0, -3.1 and -6.0 percent for \( k = 2 \). However, the convergence rate is somewhat slower, so the disparity grows with \( n \).

Fig. 3.22 shows the normalized flux as a function of \( \phi^* \) for the second order source for planar and spherical geometry, along with numerical results for \( n = 3 \). In this case, the “exact” solution was computed with \( n = 20 \).

Comparing with Fig. 3.19, the maximum difference between the curves for first and second order source functions is only about five percent. The
numerical errors also appear to be qualitatively similar. The numerical methods track the exact solution up to the start of the asymptotic solution. In a full scale chemical reactor simulation, one must solve this problem repeatedly at various locations and times in the larger system. Using OC with small \( n \) in combination with the asymptotic solution is an economical method for treating the full system [e.g. Finlayson and Young (1979)].

**Autocatalytic Source, Third Kind B.C.:** For this example values of \( Bi = 3, 10 \) and 100 are used for comparison. With the largest value of \( Bi \) the conditions approach a Dirichlet boundary condition. For a linear source, boundary collocation gave poor results compared to use of a *natural* treatment, Eq. (3.41), see Fig. 3.12. Since the geometry causes no fundamental differences in solution procedure, planar geometry is used. The source function is Eq. (3.44) with \( k = 1 \) and \( K_a = 0.95 \). This creates a rate which is essentially negative one order for small \( y \) and is called *autocatalytic*. This type of expression is not uncommon and occurs for the oxidation of carbon monoxide to carbon dioxide in millions of automotive catalytic converters.

The nonlinear equations are solved with a Newton-Raphson method, Eq. (3.12). Using a quadratic profile for an initial guess works reasonably well. However, when two solutions exist convergence problems can occur if the initial estimate is poor. During the calculations, the value of \( y \) can exceed the physical limit of 1. In this case it is generally better to use a linear extrapolation of the rate for \( y > 1 \) rather than setting \( r = 0 \).

An autocatalytic rate expression can lead to multiple steady state solutions. This is easily illustrated by considering the approximation with a single term, \( n = 1 \). The governing equations are Eqs. (3.40) and (3.41) or equivalently Eq. (3.42), where for all methods the stiffness matrix is of the form:

\[
C = \begin{bmatrix}
c_0 & -c_0 \\
-c_0 & c_0
\end{bmatrix}
\]  

(3.46)

The values of \( c_0 \) are listed in Table 3.4 along with the quadrature weights. Eq. (3.42) for \( n = 1 \) is:

\[
c_0(y_1 - y_2) - W_1 \varphi^2 \hat{r}(y_1) = 0
\]

\[
Bi y_2 + c_0(y_2 - y_1) - W_2 \varphi^2 \hat{r}(y_2)
\]

(3.47)

\( \hat{r}(y_2) \) could be approximated by a Maclaurin series, but given the magnitude of \( Bi = 100 \), there is little error by using \( \hat{r}(y_2) = 1 \). After using the second equation to eliminate \( y_2 \), the remaining equation is of the form:

\[
a \frac{y_1}{\varphi^2} - b = \hat{r}(y_1)
\]

(3.48)

**Table 3.4 Autocatalytic Reaction Parameters for \( n = 1 \)**

<table>
<thead>
<tr>
<th></th>
<th>( c_0 )</th>
<th>( W_1 )</th>
<th>( W_2 )</th>
<th>( a )</th>
<th>( b )</th>
<th>( \varphi_{\text{min}} )</th>
<th>( \varphi_{\text{max}} )</th>
<th>Error ( \varphi_{\text{min}} )</th>
<th>Error ( \varphi_{\text{max}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gauss</td>
<td>3.0000</td>
<td>1.0000</td>
<td>0.0000</td>
<td>3.9126</td>
<td>0.0000</td>
<td>0.723</td>
<td>0.901</td>
<td>1.8%</td>
<td>13.6%</td>
</tr>
<tr>
<td>Chebyshev</td>
<td>3.3704</td>
<td>0.8889</td>
<td>0.1111</td>
<td>3.6049</td>
<td>0.0029</td>
<td>0.683</td>
<td>0.852</td>
<td>-3.8%</td>
<td>6.4%</td>
</tr>
<tr>
<td>Lobatto</td>
<td>3.0833</td>
<td>0.8333</td>
<td>0.1667</td>
<td>3.4490</td>
<td>0.0041</td>
<td>0.663</td>
<td>0.825</td>
<td>-6.7%</td>
<td>3.2%</td>
</tr>
</tbody>
</table>

[28]
Table 3.4 gives the parameters for the different approximations, while Fig. 3.23 is a graph of Eq. (3.48). The graph illustrates that three solutions occur for a range of \( \varphi \). The intermediate solution is unstable, while the other two are stable. The predicted range of occurrence is calculated from the limiting slopes in Fig. 3.23. We will find that with an accurate approximation, multiple solutions occur for \( 0.71 < \varphi < 0.80 \), so the equation with \( n = 1 \) are only approximate. However, considering the simplicity of the approximation for this highly nonlinear problem, the results are good. Also, the graphical solution shown in Fig. 3.23 is helpful for understanding the multiple solution phenomenon.

For \( \varphi = 0.75 \), Figs. 3.24 and 3.25 show calculated conversion profiles for several approximations together with an accurate profile (\( n = 12 \)). As predicted by the simple \( n = 1 \) approximation, there are two stable profiles at this condition. The lower one is accurately approximated with \( n = 1 \) or 2, whereas the higher profile requires more terms to achieve an accurate solution. Even though the profile appears to be relatively smooth, keep in mind that the second derivative
must follow the rate expression plotted in Fig. 3.23. The nonlinearity of the rate function makes the problem more difficult for larger conversion, \( y > 0.8 \). The low order approximations predict nonphysical conversion values greater than unity.

Fig. 3.26 shows the residual, Eq. (3.39), of the upper solution for several approximations. Since for these conditions the greatest nonlinearity occurs for \( y > 0.8 \) or \( x < 0.4 \), the residual becomes very large and negative near the center. These errors contribute to the errors observed in Figs. 3.24 and 3.25.

The normalized boundary flux, \( \eta \), is shown for a range of \( \varphi \) in Figs. 3.27 and 3.28. The “exact” solution, calculated with \( n = 12 \), is accurate to the scale of the graph. It shows the two solutions for \( 0.71 < \varphi < 0.80 \). The lower solution shows convergence to within 0.05% for \( n = 2 \), in agreement with Fig. 3.24. However, considerably more points are required to approximate the upper solution and large \( \varphi \). For \( n = 3 \), Chebyshev and Gauss points predict false multiple solutions at large \( \varphi \). The maximum percentage error for the upper solution, \( \varphi < 1.4 \), with Gauss, Chebyshev and Lobatto points, respectively, is 7.6, 6.3 and 4.8 for \( n = 4 \), and 3.3, 2.6 and 2.2 for \( n = 6 \).

The flux for this problem can also be correlated with a generalized Thiele parameter like Eq. (3.45), where for this source function and \( k = 1 \):

\[
\varphi^* = \frac{\varphi}{\gamma + 1} \left[ 2 \int_0^1 r(\bar{y})d\bar{y} \right]^{-\frac{1}{2}} = \frac{\varphi}{\gamma + 1} \sqrt{-2[K_a + \ln(1 - K_a)]}
\]

For planar geometry and \( K_a = 0.95 \), \( \varphi^* = 0.4697\varphi \), so that asymptotically \( \eta = 1/\varphi^* \) for infinite \( Bi \). The asymptotic values plotted in Fig. 3.27 for comparison track the upper solution for \( Bi = 100 \).
Figs. 3.29 and 3.30 show errors in the normalized flux or effectiveness factor as a function of \( n \) for \( \varphi = 0.75 \) and values of \( Bi \) of 3 and 100. There are no major differences in accuracy with the different points, at least for the upper solution when a natural boundary condition treatment is used. Lobatto points produce slightly better results, but the geometric average of the error ratios for \( n < 40 \) is no more than about 2 for the three values of \( Bi \) considered. This result is different from the other problems with a linear or second order source where we found a slower convergence rate with Chebyshev points. Compared to the other problems, the number of points required to achieve high accuracy is much greater, especially when you consider that this symmetric treatment is equivalent to using twice as many points in a nonsymmetric treatment. However, even for this highly nonlinear problem, engineering accuracy is achieved with only 5 to 10 points.

For the easier lower solution, Chebyshev points give a slower convergence rate than the other two choices, which is the same behavior as for the other problems tested. However, the results for all three are similar for errors greater than \( 10^{-6} \), so for engineering accuracy, all the methods give similar results.

For comparison, both boundary collocation (labeled \textit{coll}) and a natural treatment, Eq. (3.41), are included. The results in Fig. 3.12 show that a natural boundary condition treatment is significantly better for that problem even for small \( n \), relatively loose error tolerance, and \( Bi = 2 \) to 50. However, for this problem Figs. 3.29 and 3.30 show that the differences are not as dramatic. The accuracy with both approaches is similar for errors larger than \( 10^{-4}, 10^{-6} \) and \( 10^{-8} \) for \( Bi \) of 3, 10 and 100, respectively. For the linear problem and \( Bi = 5 \), the value \( y_{n+1} = 0.5000 \) indicates that about half the transfer resistance occurs at the boundary. For this problem and \( Bi = 3 \) the value \( y_{n+1} = 0.4578 \) is similar. The difference in convergence behavior is due to the extreme nonlinearity of the source function.
To examine the boundary condition treatment, observe that the first two terms in Eq. (3.41) give the residual of the boundary condition, while the third term is the product of the quadrature weight, \( W_{n+1} \), and the interior residual, Eq. (3.39), evaluated at the boundary, i.e. \( R(1, \gamma) \). For Gauss points, the equation reduces to boundary collocation because \( W_{n+1} = 0 \), while the boundary weight is \( O(1/(2n^2)) \) for Lobatto points and \( O(1/(4n^2)) \) Chebyshev points. The natural boundary condition treatment drives both residuals to zero as the approximation is refined. This result is demonstrated for the linear problem in Fig. 3.13. For this highly nonlinear problem, Figs. 3.31 and 3.32 show the behavior of these residuals for increasing \( n \). Fig. 3.31 plots the boundary condition residual for two values of \( Bi \), while Fig. 3.32 shows both the boundary condition residual and \( R(1, \gamma) \) for \( Bi = 3 \). Both residuals go to zero at an exponential rate, but the convergence is somewhat erratic with periodic dips and frequent sign changes. This erratic behavior is evident in Fig. 3.26, but is difficult to see due to the number of curves. The ratio of the two residuals in Fig. 3.32 is the quadrature weight, \( W_{n+1} \). The exponential convergence rate of the residuals overwhelms the \( O(1/n^2) \) change in the quadrature weight.

Apparently, for this problem with large \( Bi \) and smaller values of \( n \), the errors due to the boundary condition treatment are less important than other errors in the approximation, e.g. the residuals in Fig. 3.26 which are largest near the center. For engineering accuracy, the boundary condition treatment makes little difference. This is quite different from the behavior of other problems tested. Even for small \( n \) and loose error tolerance, other problems exhibit orders of magnitude improvement using the natural boundary condition approximation. When you consider the natural treatment is easy to apply, often better and never worse, it must be the preferred method.

It may seem that we are belaboring the point regarding the treatment of flux boundary conditions. Texts have continued to recommend boundary collocation [Trefethen (2000), p. [32]
137; Boyd (2000), p. 111, Peyret (2002), p. 59], when it was shown to give poor results many years ago [see Table 5.7 and Fig. 5.7 in Finlayson (1972)]. Again, this issue does not apply to Gauss points because \( W_{n+1} = 0 \). Although many claim benefits for a nonzero boundary quadrature weight, it is not an asset when boundary collocation is used. Apparently, the benefits of a natural boundary condition treatment in conjunction with the collocation method have not generally been known. In the OC literature Gauss points have become the preferred choice since it avoids this issue altogether.

Although methods similar to the natural treatment of flux boundary conditions have been proposed in the PS literature [Canuto, et al. (1988), Funaro (1992), Shen and Tang (2006) ], it appears these alternatives have not caught on. Most PS applications continue to make the mistake of using boundary collocation with Chebyshev and Lobatto points. The problem is easily avoided by using a natural treatment of flux boundary conditions instead. Chapter 4 on parabolic problems further emphasizes the pitfalls of using boundary collocation.

### 3.1.6 Orthogonal Polynomial Trial Functions

The choice of trial functions is discussed in Section 1.2 and the justification for using Lagrange interpolating polynomials or nodal methods is given there. For problems of interest, interpolating polynomials rather than orthogonal polynomials are usually the best choice. For completeness, a couple of examples are considered here using orthogonal polynomial trial functions (modal methods) to give the reader a flavor of the differences between the two approaches.

The problem of diffusion with a linear source and Dirichlet boundary conditions, Eqs. (3.1) with (3.1a) and (3.2), is considered with constant and variable coefficients. We do not consider Chebyshev polynomials here, but their use is well documented in the references (Shen and Tang (2006), Ch. 3 is highly recommended). For convenience, the domain \([-1,1]\) is used. The factor of 4 used in the rate function is dropped to solve the same problem with this larger domain.

We wish to use Legendre polynomials as trial functions, so the solution is approximated by Eq. (2.1) duplicated here:

\[
\tilde{y} = \sum_{k=0}^{n+1} P_k(x) \, \tilde{a}_k
\]  

(3.49)

where \( n \) corresponds to the number of interior points in a nodal method. The method is called a modal method, because the unknown coefficients, \( \tilde{a} \), are equivalent to the modes in a Fourier series.

As discussed in Section 2.1, the even numbered polynomials are even functions of \( x \), i.e. symmetric about \( x = 0 \), while the odd numbered ones are odd. If the problem is symmetric like those considered in Section 3.1.5, only even number polynomials would be used in the
approximation, Eq. (3.49). The first few Legendre polynomials are given in Eq. (2.20) while the others can be built up using the three term recurrence relations discussed in Section 2.1.

These polynomials are orthogonal on the interval [-1,1] with a weight function of unity. They correspond to a Jacobi polynomial with $\alpha = \beta = 0$. There are several ways one could treat the boundary conditions.

$$y(\pm 1) = 0$$

One method is to add constraint equations to explicitly enforce them, while another method is to select combinations of the orthogonal polynomials which will meet the boundary conditions [Boyd (2000)]. Shen and Tang describe this later approach for a general boundary condition, including 1st, 2nd and 3rd type as special cases. That approach is used here for this simple problem. Legendre polynomials are normalized so the boundary values are:

$$P_k(1) = 1 \quad \text{and} \quad P_k(-1) = (-1)^k$$

Since the polynomials are alternately odd and even about $x = 0$, the simplest method to meet the boundary conditions is to subtract $P_0 = 1$ from the even numbered polynomials in Eq. (3.49) and subtract $P_1 = x$ from the odd numbered ones. A better matrix structure is achieved if Eq. (3.49) is modified to:

$$\tilde{y} = \sum_{k=0}^{n-1} \left( P_{k+2}(x) - P_k(x) \right) a_k = \sum_{k=0}^{n-1} \psi_k(x) a_k$$

(3.50)

Villadsen and Michelsen (1978) considered this problem for a symmetric solution in cylindrical geometry. They expanded the solution in terms of Jacobi polynomials. A similar approach for this geometry would use Jacobi polynomials with $\alpha = \beta = 1$, and a multiplier of $(1 - x^2)$ enforces the boundary conditions. These trial functions are equivalent to those in Eq. (3.50), since Eq. (2.39) is the identity:

$$\left(1 - x^2\right) P_k^{(1,1)}(x) = -\frac{2k + 2}{2k + 3} \left( P_{k+2}(x) - P_k(x) \right)$$

(3.51)

Here we follow the same convention used in Chapter 2. A superscript is used to designate the $\alpha$ and $\beta$ of a Jacobi polynomial, e.g. $(1,1)$ designates $\alpha = \beta = 1$, while no superscript indicates a Legendre polynomial, $\alpha = \beta = 0$. The roots of the Jacobi $(1,1)$ polynomials are the Lobatto quadrature base points.

Given the trial functions, $\psi$, the residual of Eq. (3.1) with first order source and constant coefficients is:

$$R(x, a) = \sum_{k=0}^{n-1} \left[ \psi_k(x) - \varphi^2 \psi_k \right] a_k + \varphi^2$$

(3.52)

Substituting the trial functions, Eq. (3.50), and then Eq. (2.42) for the second derivative, the residual is:
\[ R(x, a) = \sum_{k=0}^{n-1} \left[ P_{k+2}'' - P_k'' - \varphi^2(P_{k+2} - P_k) \right] a_k + \varphi^2 \]

(3.53)

where \( S \) is constructed from Eq. (2.42).

With collocation the residual, Eq. (3.53), is set to zero at \( n \) interior collocation points. The resulting problem is a full \( n \times n \) matrix problem. Solution of the matrix problem produces coefficients, \( a_k \), which give exactly the same nodal values as the nodal solutions in Section 3.1.1. The effort required to solve the problem is similar to that of a nodal formulation. It turns out though, that if an integrated MWR is applied simplifications sometimes occur.

**Moments Method**

Section 1.1 briefly described the Moments method and Section 3.1.1 developed a nodal formulation for the problem considered here. Now we wish to use the trial functions in Eq. (3.50) rather than interpolating polynomials. The weight functions with the Moments method are nominally the monomial powers of \( x \). However, as explained in Section 1.1, any linearly independent set of polynomials which contains the monomials is equivalent. The natural choice of weight functions are the Legendre polynomials:

\[ \int_{-1}^{1} \sum_{n=0}^{n-1} \left[ \psi_k''(x) - \varphi^2 \psi_k \right] a_k + \varphi^2 \right] P_{\ell}(x) dx = 0 \]

(3.54)

\( \ell = 0, \ldots, n - 1 \). The Moments method is equivalent to the Tau method [Lanczos (1956)]. The only difference is that here the trial functions have been constructed to meet the boundary conditions, while with the Tau method side conditions are used to enforce the boundary conditions. The end result is the same, but the Moments method produces a better matrix structure.

Eq. (3.54) may be written in matrix notation as:

\[ [C - \varphi^2 D] a = -\varphi^2 h \]

(3.55)

We identify \( C \) and \( D \) as the stiffness and mass matrices, respectively, and \( h \) as the load vector. These quantities can be integrated easily using Eqs. (2.6) and (2.42).

\[ C_{ij} = \int_{-1}^{1} P_i(x) \psi_j''(x) dx = 2(2j + 3) \text{ for } j \geq i, i + j \text{ even} \]

\[ D_{ij} = \int_{-1}^{1} P_i(x) \psi_j(x) dx = \frac{2}{2l + 1} \left( \delta_{i,j+2} - \delta_{ij} \right) \]

\[ h_i = \int_{-1}^{1} P_i(x) dx = \int_{-1}^{1} P_i(x) P_0(x) dx = 2 \delta_{0i} \]

The stiffness matrix, \( C \), is upper triangular with alternate diagonals zero due to the alternating odd/even nature of the polynomials. The mass matrix, \( D \), consists of a diagonal and one
subdiagonal. The load vector, \( \mathbf{h} \), is integrated by substituting \( P_0 = 1 \). Only the first row has a nonzero value. The matrices for \( n = 6 \) are:

\[
\mathbf{D} = \begin{pmatrix}
-2 & 0 & 0 & 0 & 0 & 0 \\
0 & -2/3 & 0 & 0 & 0 & 0 \\
2/5 & 0 & -2/5 & 0 & 0 & 0 \\
0 & 2/7 & 0 & -2/7 & 0 & 0 \\
0 & 0 & 2/9 & 0 & -2/9 & 0 \\
0 & 0 & 0 & 2/11 & 0 & -2/11
\end{pmatrix}
\]

\[
\mathbf{C} = \begin{pmatrix}
6 & 0 & 14 & 0 & 22 & 0 \\
0 & 10 & 0 & 18 & 0 & 26 \\
0 & 0 & 14 & 0 & 22 & 0 \\
0 & 0 & 0 & 18 & 0 & 26 \\
0 & 0 & 0 & 0 & 22 & 0 \\
0 & 0 & 0 & 0 & 0 & 26
\end{pmatrix}
\]

When combined the matrix is upper triangular except for the addition of the subdiagonal from the mass matrix. The matrix problem can be solved more efficiently than a full matrix, since only the subdiagonals need be eliminated. Some sample solutions with \( \varphi = 5 \), are:

\[
\mathbf{a} = \{-0.808625, 0, -0.336927, 0\} \quad \text{for } n = 3 \text{ or } 4
\]

\[
\mathbf{a} = \{-0.800214, 0, -0.281414, 0, -0.056737\} \quad \text{for } n = 5 \text{ or } 6
\]

\[
\mathbf{a} = \{-0.800020, 0, -0.280131, 0, -0.050260, 0, -0.005711, 0\} \quad \text{for } n = 7 \text{ or } 8
\]

When Eq. (3.50) is evaluated at the Gauss points with these coefficients, the values are identical to those found in Fig. 3.1 of Section 3.1.1 with an even number of Gauss points (which is equivalent to Moments). Since the solution is symmetric in \( x \), the alternate coefficients, i.e. those of the odd polynomials, are zero. The solutions for the two sets of points indicated are identical except for an extra zero coefficient for the last odd function. The rapid decay of the coefficient values and the rapid convergence of the lower order coefficients, indicates the fast convergence of the solution with increasing \( n \). This convergence behavior has important consequences for the accuracy of the boundary flux, Eq. (3.4).

Integration of the approximate solution, Eq. (3.50), gives the normalized flux:

\[
\eta = \frac{1}{2} \int_{-1}^{1} (1 - y) \, dx = 1 - \frac{1}{2} \sum_{k=0}^{n-1} a_k \int_{-1}^{1} (P_{k+2}(x) - P_k(x)) \, dx = 1 + a_0
\]

The analytical solution, Eq. (3.6), gives the exact value \( a_0 = -0.800018 \). Normally, the derivatives of an approximate solution converges slower than the solution itself. However, the boundary flux is a special case, since integration can be used as explained in the discussion of Eq. (3.5). When solved with the Moments method the boundary flux converges more quickly than the overall solution. This behavior is clearly demonstrated by comparison of the L2 error norms and flux errors in the examples above.

**Galerkin Method**

Now consider the solution of this problem with the trial functions, Eq. (3.50), giving the same residual, Eq. (3.52). However, with weighting by the trial functions, the weighted residual is modified to:
\[
\int_{-1}^{1} \left\{ \sum_{k=0}^{n-1} \left[ \psi_k''(x) - \varphi^2 \psi_k \right] a_k + \varphi^2 \right\} \psi_\ell(x) dx = 0
\] (3.57)

By comparing Eqs. (3.54) and (3.57), a simple relationship between the Galerkin and Moments approximations is apparent. Both the mass matrix and stiffness matrix are related as follows:

\[
C_{ij}^G = C_{i+2,j}^M - C_{ij}^M
\] (3.58)

where the superscripts, \(G\) and \(M\), indicate the matrices for the Galerkin and Moments methods, respectively. The resulting matrices are:

\[
D^G = \begin{pmatrix}
2.4 & 0 & -2/5 & 0 & 0 & 0 \\
0 & 0.9524 & 0 & -2/7 & 0 & 0 \\
-2/5 & 0 & 0.6222 & 0 & -2/9 & 0 \\
0 & -2/7 & 0 & 0.4675 & 0 & -2/11 \\
0 & 0 & -2/9 & 0 & 0.3761 & 0 \\
0 & 0 & 0 & -2/11 & 0 & 0.3152
\end{pmatrix}
\]

\[
C^G = \begin{pmatrix}
6 & 0 & 0 & 0 & 0 & 0 \\
0 & 10 & 0 & 0 & 0 & 0 \\
0 & 0 & 14 & 0 & 0 & 0 \\
0 & 0 & 18 & 0 & 0 & 0 \\
0 & 0 & 0 & 22 & 0 & 0 \\
0 & 0 & 0 & 0 & 26 & 0
\end{pmatrix}
\]

Compared to the Moments method, these modifications produce a matrix problem which can be solved more efficiently, especially for large \(n\). The stiffness matrix is diagonal, while the mass matrix is pentadiagonal. If the symmetric nature of the problem were exploited by eliminating the odd polynomials from the expansion the mass matrix would be tridiagonal.

One nice feature of the modal formulation is that it is much easier to calculate the solution for different \(n\) in order to monitor the convergence. The matrices can be calculated for the maximum \(n\) of interest, which gives the matrices for smaller \(n\) as just the upper left portion of the larger matrix. For example, the solution for \(n = 3\) can be obtained by using the upper left 3x3 submatrix of the larger matrix. For a nodal formulation, the matrix problem is different for each \(n\), so monitoring convergence requires the calculation of a new matrix for each \(n\) of interest.

Some example solutions for \(\varphi = 5\), are:

\[
a = \{-0.798511, 0, -0.270173, 0\} \quad \text{for } n = 3 \text{ or } 4
\]

\[
a = \{-0.799998, 0, -0.279987, 0, -0.049535, 0\} \quad \text{for } n = 5 \text{ or } 6
\]

\[
a = \{-0.800018, 0, -0.280119, 0, -0.050201, 0, -0.005249, 0\} \quad \text{for } n = 7 \text{ or } 8
\]

When Eq. (3.50) is evaluated at the Lobatto points with these coefficients, the values are identical to those found in Fig. 3.1 of Section 3.1.1 with an even number of Lobatto points (which is equivalent to a Galerkin method).

The convergence rate of these coefficients is somewhat better than that for the Moments method, so the comments above regarding the accuracy of flux calculations apply equally to the Galerkin method.
Variable Coefficients

The banded matrix structure given by the Galerkin method above and even the near upper triangular matrix produced by the Moments method can be solved more efficiently than the full matrix problems produced by the nodal approximations resulting from Lagrange interpolating polynomials. Unfortunately, these efficient matrix structures are lost if the problem is nonlinear or even if it has variable coefficients. To demonstrate this property, consider the problem with source function in Eq. (3.33), which is linear in $y$, but cubic in $x$. The cubic function $q(x)$ varies from 0.2 to 1.8, but with an average value of unity (note the factor of 4 is dropped from the equation because the domain here is $[-1,1]$). To facilitate solution of the problem, the variable coefficient is expressed as a function of Legendre polynomials. One way to determine these coefficients is through a discrete Legendre transform as described in section 2.5. The result is:

$$q(x) = P_0 + 0.96P_1 - 0.16P_3$$

In this form it is clear from the properties of Legendre polynomials that the average value is unity and the values at the two boundaries are as stated above.

Eq. (3.55) still applies for this problem and the stiffness matrices are the same as before, but the mass matrix and load vector change. For the Galerkin method, these quantities are:

$$D_{ij} = \int_{-1}^{1} \psi_i(x) \psi_j(x) q(x) \, dx$$

$$h_i = \int_{-1}^{1} \psi_i(x) q(x) \, dx$$

The load vector can easily be integrated analytically when $q$ is expressed in terms of the Legendre polynomials. Integration of the mass matrix analytically is more complicated. One possibility is to interpolate the product of $q$ and $\psi$ as a discrete Legendre series and then integrate. It is probably simpler just to use numerical quadrature to integrate the expressions as was done when this problem was solved by nodal methods. Table 3.1 details the degree of the mass matrix integrand. Exact integration for the Galerkin method is obtained using Gaussian quadrature with $n + 3$ points or Lobatto quadrature with $n + 2$ interior points. One less quadrature point is needed for Moments.

Exact integration of the Galerkin mass matrix and load vector gives:

$$D^c = \begin{bmatrix}
2.4000 & 0.6095 & -0.4000 & -0.2119 & 0 & 0.0139 \\
0.6095 & 0.9524 & 0.2101 & -0.2857 & -0.1446 & 0 \\
-0.4000 & 0.2101 & 0.6222 & 0.1432 & -0.2222 & -0.1131 \\
-0.2119 & -0.2857 & 0.1432 & 0.4675 & 0.1090 & -0.1818 \\
0 & 0.1446 & -0.2222 & 0.1090 & 0.3761 & 0.0884 \\
0.0139 & 0 & -0.1131 & -0.1818 & 0.0884 & 0.3152
\end{bmatrix} \quad \text{and} \quad h^c = \begin{bmatrix}
-2.0000 \\
-0.6857 \\
0 \\
0.0457 \\
0 \\
0
\end{bmatrix}$$

Except for a couple of zeros, this matrix is full, so the beneficial matrix structure found for the constant coefficient problem is lost. A full matrix also results when the source is nonlinear. Solving a nonlinear problem is further complicated by the indirect dependence of the dependent variable, $y$, on the unknown coefficients, $a$. Nonlinear problems are much easier to solve with a nodal method.
Some examples of solutions using the Moments method with $\varphi = 5$ are:

- \( a = \{-0.780464, -0.139001, -0.290515, -0.136558\} \) for \( n = 4 \)
- \( a = \{-0.765444, -0.154772, -0.232674, -0.105690, -0.061149, -0.026173\} \) for \( n = 6 \)

Some solutions with the Galerkin method are:

- \( a = \{-0.765254, -0.154036, -0.229734, -0.106528, -0.047540, -0.022546\} \) for \( n = 6 \)

These solutions are identical to those found for the nodal Moments and Galerkin methods shown in Figs. 3.15 and 3.16 and Table 3.2 when converted to a common basis.

Methods to convert between modal and nodal representations are discussed in Section 2.5. To demonstrate the conversion, consider the Galerkin solution of the variable coefficient problem with \( n = 4 \). The values at the Lobatto points are illustrated in Fig. 3.16. These values can be compared by using the coefficients above to evaluate Eq. (3.50) at the Lobatto points. The result is:

\[ y = \{0, 0.46603, 0.92731, 0.97087, 0.79069, 0\} \]

which is identical to the results found using the nodal formulation.

Suppose we had not solved the problem with the modal formulation but want to know the modal coefficients. In that case, a Legendre transformation matrix, discussed in Section 2.5, can be used to calculate the coefficients from the nodal values by \( \hat{a} = Qy \). The result of the calculation is the coefficients of Eq. (3.49), which are:

\[ \hat{a} = \{0.76443, 0.15135, -0.54577, -0.04337, -0.21866, -0.10798\} \]

These can be determined from those above, by collecting like terms. The point of this discussion is that the solution is exactly the same, but there are different ways to represent it. Using one form of trial function or another does not change the solution, though some texts seem to suggest that it does.

### 3.2 Chemical Reactor with Axial Dispersion

For the next example of a boundary value problem, consider nonisothermal flow through a chemical reactor with axial conduction and dispersion. This is a coupled heat and mass transfer problem. The governing equations for the problem are:

\[
\frac{1}{Pe_m} \frac{d^2y}{dz^2} - \frac{dy}{dz} + r_m(y, T) = 0 \quad \text{and} \\
\frac{1}{Pe_t} \frac{d^2T}{dz^2} - \frac{dT}{dz} - UT + r_t(y, T) = 0
\]

with:

\[
\frac{dy}{dz} = Pe_m y \quad \text{and} \quad \frac{dT}{dz} = (Pe_t + U)T \quad \text{at} \quad z = 0 \quad \text{and} \\
\frac{dy}{dz} = 0 \quad \text{and} \quad \frac{dT}{dz} + UT = 0 \quad \text{at} \quad z = 1
\]

The model allows for cooling or heating at the wall using a lumped parameter approximation with an overall heat transfer coefficient, parameter \( U \). These problems are convection...
dominated, since the Peclet numbers, $P_{Em}$ and $P_{Et}$, are normally large. The solution is nonsymmetric and all the boundary conditions are of the second or third kind. Models with heating and cooling often fail to account for this effect in the boundary conditions. The boundary conditions were correctly treated by Young and Finlayson (1973).

In an industrial reactor model several component balances could be required to represent a system of reactions, so Eq. (3.59) is formulated as the following system of coupled equations:

$$\frac{1}{Pe_k} \frac{d^2 y_k}{dz^2} - \frac{dy_k}{dz} - U_k y_k + r_k(y) = 0 \quad (3.60)$$

with

$$\frac{dy_k}{dz} = (Pe_k + U_k) y_k \text{ at } z = 0 \quad \text{and} \quad \frac{dy_k}{dz} + U_k y_k = 0 \text{ at } z = 1$$

for $k = 0, \ldots, n_m$. We assign $k = 0$ to the energy balance which is combined with as many mass balances as required to represent the reaction system of interest.

After the application of conventional orthogonal collocation, Eq. (3.60) is approximated by:

$$\sum_{i=0}^{n+1} \left( \frac{1}{Pe_k} B_{ji} - A_{ji} - U_k \delta_{ji} \right) y_{ik} + r_k(y_j) = 0 \quad (3.61)$$

for $j = 1, \ldots, n$, and using boundary collocation as recommended by most authors:

$$\sum_{i=0}^{n+1} \left[ A_{0,i} - (Pe_k + U_k) \delta_{0,i} \right] y_{ik} = 0 \quad \text{and} \quad (3.61a)$$

$$\sum_{i=0}^{n+1} \left[ A_{n+1,i} + U_k \delta_{n+1,i} \right] y_{ik} = 0$$

where $y_{ik} = [T(z_i), y_1(z_i), \ldots, y_{n_m}(z_i)]$ and $y_j$ is the vector of all values evaluated at point $z_j$. Based on the results for the previous example, we anticipate that boundary collocation, Eq. (3.61a), will work well for Gauss points but not so well for Lobatto or Chebyshev points.

A natural treatment of the boundary conditions works better for Lobatto and Chebyshev points. It is the standard method for treating flux boundary conditions with Galerkin methods and it falls out when the equations are cast in weak form. The weak form of the Galerkin method is derived by integrating the Laplacian by parts and substituting the boundary conditions for the boundary derivatives. This exercise was carried out for the previous problem in Eqs. (3.23) and (3.24). Then quadrature is used to approximate the integrals as in Eq. (3.25). With a slight generalization in the definition of the stiffness matrix, $C$ in Eq. (3.26), the same formulation can be applied for any type of points. For this problem, the resulting weak form is:

$$\delta_{0,j} \left( 1 + \frac{U_k}{Pe_k} \right) y_{0,k} + \delta_{n+1,j} \frac{U_k}{Pe_k} y_{n+1,k} + \sum_{i=0}^{n+1} \left( \frac{1}{Pe_k} C_{ji} + W_j A_{ji} + U_k W_j \delta_{ji} \right) y_{ik} - W_j r_k(y_j) = 0 \quad (3.62)$$

From the previous discussions, we know the weak form, Eq. (3.62), is equivalent to the conventional form, Eq. (3.61), at the interior points, but the boundary conditions are treated
In order to compare the two methods for treating the boundary conditions, Eq. (2.60) or (3.26) is substituted for \( C \), which gives the boundary equations:

\[
\sum_{i=0}^{n+1} \left[ (1 + \frac{U_k}{P e_k}) \delta_{0,i} - \frac{A_{0,i}}{P e_k} \right] y_{ik} - W_0 \left[ \sum_{i=0}^{n+1} \left( \frac{B_{0,i}}{P e_k} - A_{0,i} - U_k \delta_{0,i} \right) y_{ik} + r_k(y_0) \right] = 0
\]

\[
\sum_{i=0}^{n+1} \left[ \frac{U_k}{P e_k} \delta_{n+1,i} + \frac{A_{n+1,i}}{P e_k} \right] y_{ik} - W_{n+1} \left[ \sum_{i=0}^{n+1} \left( \frac{B_{n+1,i}}{P e_k} - A_{n+1,i} - U_k \delta_{n+1,i} \right) y_{ik} + r_k(y_{n+1}) \right] = 0
\]

The first term is the boundary condition residual and the second is the interior residual evaluated at the boundary and multiplied by the quadrature weight. This relationship is like that found for the previous example, see Eq. (3.27). Rather than setting one or the other residual to zero, this procedure sets a combination of the two to zero. Although neither residual will be identically zero, they will converge to zero. For Gauss points, the boundary quadrature weights are zero, so Eq. (3.61a) and (3.63) are equivalent.

When Gauss points are used, the method is an accurate approximation to the Moments method and when Lobatto points are used, it is an accurate approximation to the Galerkin method. In each case the dispersion and convection terms are integrated exactly. If the rate terms are interpolated, like Eq. (3.34), integration of both the rate term and heat transfer term miss exact integration by one degree. The reaction term is nonlinear, so the accuracy of the approximation depends on the severity of the nonlinearity.

Eq. (3.62) can be expressed in the form:

\[
\sum_{i=0}^{n+1} \tilde{A}_{ji} y_{ik} - W_j r_k(y_j) = 0
\]

where:

\[
\tilde{A}_{ji} = \frac{1}{P e_k} C_{ji} + W_j A_{ji} + \delta_{ji} \left[ U_k W_j + \delta_{0,j} \left( 1 + \frac{U_k}{P e_k} \right) + \delta_{n+1,j} \frac{U_k}{P e_k} \right]
\]

Eq. (3.61) can be expressed in the same form. The form of Eq. (3.64) is similar to that of Eq. (3.10), so the same solution procedures can be used. The only difference is that here we have a coupled set of equations rather than a single equation.

Although the problem is formulated for any number of mass balance equations, we consider only one mass balance together with the energy balance. A first order exothermic reaction is used:

\[
r_k = Da \hat{r}(y, T) = Da_k (1 - y) \exp \left( \frac{20T}{T + 1} \right)
\]

\( Da \) are the Damkohler numbers for energy and mass. The parameter values considered in the example are: \( Pe_t = 100, Pe_m = 200, U = 3, Da_t = 0.2, Da_m = 0.5 \). This problem is highly nonlinear due to the temperature dependence in Eq. (3.65).

Since Eqs. (3.64) are nonlinear, an iterative procedure is required. The rate equations also couple the energy and mass balance equations, so a Newton-Raphson method will require the
simultaneous solution of all equations. An alternative procedure which solves the equations sequentially is a Picard iteration:

$$\sum_{i=0}^{n+1} (\tilde{A}_{ji}^k + \delta_{ji} \mu_{ki}) \Delta y_{ik} = W_j r_k^s(y_j^s) - \sum_{i=0}^{n+1} \tilde{A}_{ji}^k y_{ik}^s$$  \hspace{1cm} (3.66)

where the $\Delta$ indicates the change in the variable over the iteration and superscript $s$ indicates the values at the start of the iteration. The $\mu$ are iteration parameters. One can think of these parameters as an approximation to the rate derivatives. For proper scaling, we take $\mu_{ki} = a_k W_i D a_k$, where an $a$ is supplied for each balance. Larger values of the iteration parameter make the matrix more diagonally dominant and will tend to reduce the size of changes. With this procedure, the matrices are modified by the iteration parameters and then factored requiring $O(\frac{3}{2}(n_m+1)(n+2)^3)$ operations. Since the matrix does not change from one iteration to the next, the factors need to be calculated only once. Once the matrices are factored, the iterations consist of first calculating the rates and then sequentially for each balance, calculate the right side of Eq. (3.66) and updating the values by a forward and back substitution with the factors of the matrices. The calculations per iteration are $O(4(n_m+1)(n+2)^2)$. This method normally converges slowly, because it is basically successive substitution and does not account for how the rates change with changes in $y$. However, the iterations require little calculation.

The rate terms can be linearized about some initial or intermediate estimates $y^s$ to give the approximation

$$r_k(y_j) \approx r_k(y_j^s) + \sum_{\ell=0}^{n_m} \frac{\partial r_k}{\partial y_{j\ell}} y_{j\ell}^s \Delta y_{j\ell} = r_{jk} + \sum_{\ell=0}^{n_m} d_{jk}^k \Delta y_{j\ell}$$  \hspace{1cm} (3.67)

where $y$ is the vector which includes $T$ as its first member. This linearization is substituted into Eq. (3.64) and a Newton-Raphson iteration, like Eq. (3.12), can be used to solve the system of equations.

The iterations are then:

$$\sum_{i=0}^{n+1} (\tilde{A}_{ji}^k - \delta_{ji} W_j d_{jk}^k) \Delta y_{jk}^s - W_j \sum_{\ell=0}^{n_m} d_{j\ell}^k \Delta y_{j\ell} = W_j r_{jk} - \sum_{i=0}^{n+1} \tilde{A}_{ji}^k y_{ik}^s$$  \hspace{1cm} (3.68)

It is usually more accurate to formulate the equations to solve for changes, i.e. $\Delta y$, rather than directly for the updated values.

The resulting matrix for the linearized problem, Eq. (3.68), with 3 interior points is of the form:
This equation is a block 2x2 matrix (or in general \((n_{m+1})x(n_{m+1})\), where each block is an \((n+2)x(n+2)\) matrix. The diagonal blocks are full and the off-diagonal blocks are diagonal.

The coupling between \(y\) and \(T\) is through the derivatives of the rate terms in Eq. (3.67), which appear on the diagonals of each block. This coupling is not present with the Picard iteration. This procedure, Eq. (3.68), requires solving one large matrix, rather than \(n_{m+1}\) smaller ones. Also, with the Newton-Raphson iteration, the matrix changes each iteration. Overall, each iteration requires \(O\left(\frac{2}{3}(n_{m+1})^3(n+2)^3\right)\) operations, substantially more than for the Picard iteration. We refer to this procedure as a full Newton-Raphson, since all the nonlinear equations are solved together.

The rate dependence in the energy equation can always be expressed as a linear combination of the mass balance equations. This relationship can be used to develop an alternate formulation. We will demonstrate the procedure for \(n_{m} = 1\), but keep in mind the energy equation can always be eliminated with a similar procedure. The energy and mass balance equations can be combined to give in matrix notation:

\[
\tilde{A}^0 \tilde{y}_0 = \frac{D a_0}{D a_1} \tilde{A}^1 y_1 \text{ or } y_0 = \frac{D a_0}{D a_1} (\tilde{A}^0)^{-1} \tilde{A}^1 y_1 = \frac{D a_0}{D a_1} G y_1
\]  

Then the Newton-Raphson iterations simplify to:

\[
\sum_{i=0}^{n+1} (\tilde{A}_{ji}^j - \delta_{ji}W_j a_{j1}^0 - W_j a_{j0}^0 G_{ji}) \Delta y_{i1} = W_j r_{j1} - \sum_{i=0}^{n+1} \tilde{A}_{ji}^j y_{i1}^s
\]  

This formulation requires more calculations to set up the iterations, since the calculation of \(G\) requires roughly four times the calculations required to factor \(\tilde{A}^0\) or \(O(2^{3/2}(n+2)^3)\), but it requires fewer calculations per iteration since the dimension of the matrix is half that for full Newton-Raphson, Eq. (3.68). With this procedure, Eq. (3.71) is calculated and solved for \(\Delta y_1\), then Eq. (3.70) is used to calculate \(\Delta y_0\). We will call this the reduced procedure, since it reduces the size of the problem by eliminating the temperature dependence.

From the discussion above, it is apparent there are tradeoffs between the iterative procedures discussed. The Picard iterations is favored for larger systems, since the computational work grows in direct proportion to \(n_m\) rather than to some power. It is also favored for large \(n\). From testing we have found that solution of the algebraic problem to 6 digits of accuracy requires about 8 Newton-Raphson iterations or 20 Picard iterations. The Picard method typically requires 2½ or more times as many

\[43\]
iterations as a Newton method depending on the convergence tolerance. The number of iterations is not sensitive to the values of the iteration parameters. For the Picard method, it is surprising that the number of iterations tends to decrease with $n$. If $n$ is too small, none of the methods converge.

Table 3.5 summarizes operation counts (in thousands of floating point operations, Kflops) for the three iterative procedures described above. The full Newton method is not nearly as efficient as the other two. This problem is small enough that for a single solution, efficiency is not a major issue, so an in-depth analysis is overkill. However, knowing how to analyze algorithms, without solving all the alternatives, is an important skill to have. Also, there are many cases where the problem could be much larger or could require many solutions. For example, the number of reactions and $n_m$ could be quite large. Radial as well as axial dispersion could be important making the problem two dimensional with many more points. Suppose a small reactor was used to measure reaction rates, and the data needs to be analyzed as an integral reactor using nonlinear regression. In that case the problem would be solved hundreds of times. This type of analysis has been done and efficient methods are essential [Young and Finlayson (1973)].

There are some other techniques that could potentially improve the efficiency of the calculations. Eq. (3.69) is a band matrix, so a band solver would be slightly more efficient. When Gauss points are used, the boundary equations for the first and last points do not depend on the rate, see Eqs. (3.61a) and (3.63). Those two equations can be eliminated at the outset to reduce the number of nonlinear equations which must be solved.

In Newton-Raphson methods, the matrix in Eq. (3.68) or Eq. (3.71) is called the **Jacobian** matrix. Frequently, some efficiency can be gained by updating the Jacobian only periodically, e.g. every other iteration. For our problem calculating and factoring the Jacobian consumes about 70% of the calculations for an iteration, so there is a good potential for improvement. Other solution strategies may be discovered by working with the codes for this problem, e.g. start with a few Picard iterations, then switch to a Newton-Raphson.

Figs. 3.33 – 3.35 show some example profiles with Gauss, Chebyshev and Lobatto points. In all cases, the solid black line is the solution with $n = 51$, which is exact for practical purposes. The graphs for Chebyshev and Lobatto points show the solutions for both boundary collocation, Eq. (3.61a), and a natural treatment, Eq. (3.62) or (3.63). In the previous problem, the errors caused by boundary collocation were not obvious when comparing the profiles, see Figs. 3.9 and 3.10, and were only apparent when comparing fluxes. For this problem, the differences are clearly visible in the profiles of Figs. 3.34 and 3.35.

The differences in the profiles are also visible in the $L_1$ error norms for the conversion, $y$, shown in Fig. 3.36. There is again little difference in the comparison with $L_1$ or $L_2$ error norms.

Table 3.5 Estimated Kflops for $n = 18$

<table>
<thead>
<tr>
<th>Method</th>
<th>Picard</th>
<th>Reduced Newton</th>
<th>Full Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup</td>
<td>10.3</td>
<td>20.8</td>
<td>0.0</td>
</tr>
<tr>
<td>Iteration</td>
<td>3.2</td>
<td>8.4</td>
<td>43.5</td>
</tr>
<tr>
<td>total*</td>
<td>74.3</td>
<td>87.6</td>
<td>348.0</td>
</tr>
</tbody>
</table>

*8 iterations for Newton-Raphson, 20 for Picard
All of the errors for temperature, $T$, are smaller by roughly a factor of 5, which corresponds to the maximum variation of $y$ relative to $T$. For this reason, all of the reported error results are for conversion, $y$, and the mass balance equation. Fig. 3.36 shows virtually no difference in the results with the different points, Gauss, Chebyshev or Lobatto, but there are significantly greater errors when boundary collocation, Eq. (3.61a), is used with Chebyshev or Lobatto points instead of a natural treatment, Eq. (3.63). These results were calculated for every $n$ for $n = 9$ to 31 and only odd points for larger $n$. The error tends to oscillate based on where the points lie with respect to the more difficult parts of the profile such as where the rate is large.
Fig. 3.37 shows the error in conversion at the center. The error is calculated only for an odd number of points, since an even number requires interpolation to determine the value at the center and consequently the error is much larger. In agreement with Fig. 3.36, Fig. 3.37 shows significantly larger error with boundary collocation and erratic behavior for \( n < 20 \). Although the overall average convergence rate is similar with boundary collocation, the actual error at some \( n \) is greater by almost one order of magnitude for Chebyshev points and two orders of magnitude with Lobatto points. The error is larger with Lobatto points, because the difference between Eqs. (3.61a) and (3.63) is proportional to the boundary quadrature weights, \( W_0 = W_{n+1} \), and the weights are approximately twice as large for Lobatto points, so it is more important to treat the boundary condition correctly with Lobatto points.

Fig. 3.38 shows the residual of Eq. (3.60) for two cases. The residual errors tend to be largest near the inlet and where the rate declines sharply as complete conversion is approached. All point types display this behavior, but the errors are slightly smaller near the inlet and larger near the center for Chebyshev and Gauss points due to the greater concentration of points near the ends at the expense of their spacing near the center.

Fig. 3.39 shows the residual of the inlet and outlet boundary conditions for the natural treatment of the mass balance equation, i.e. the left term in Eq. (3.63). These are identically zero for Gauss points. The errors are larger at the inlet as expected from Fig. 3.38, but they converge exponentially. Eq. (3.63) relates the residual of the boundary condition to the interior residual of Fig. 3.38 at \( x = 0 \) and 1. The boundary values of the interior residual are larger by a factor of \( 1/W_0 = 1/W_{n+1} \). Although the boundary weights decrease in proportion to \( n^2 \), the residuals will decline at the faster exponential rate as shown for the previous examples in Figs. 3.13 and 3.32.

The results for Lobatto points in Fig. 3.39 were calculated for every \( n \) from \( n = 9 \) to 51 and at only odd points for larger \( n \). For Lobatto points the odd points give lower errors for \( n < 38 \) and

[46]
even numbered points give lower errors at larger \( n \). The oscillations between odd and even points are a function of the distribution of points with respect to the more difficult parts of the profile. Only odd numbered points were used for the Chebyshev calculations, so the results do not oscillate like those for the Lobatto results.

These calculations provide another example of an application to a nonlinear boundary value problem. This example is different from the previous one since it contains the first derivative convection term which dominates the smaller dispersion term. It shows, once again, that a natural treatment of flux boundary conditions is superior to boundary collocation. The sharp profiles in this example require a large number of points to achieve acceptable engineering accuracy. For example, 11 points are required to achieve a one percent average L₁ error and 15 for a one percent error in \( y \) at the center. This problem is a good candidate for a finite element approach.