

**OPTIMAL COLLOCATION SOLUTION OF THE  
ONE-DIMENSIONAL STEADY-STATE  
CONVECTION-DIFFUSION EQUATION WITH VARIABLE  
COEFFICIENTS**

STEPHEN H. BRILL  
DEPARTMENT OF MATHEMATICS  
BOISE STATE UNIVERSITY  
BOISE, IDAHO, 83725-1555 USA  
E-MAIL: BRILL@MATH.BOISESTATE.EDU

ABSTRACT. We study the Hermite collocation solution of the one-dimensional steady-state convection-diffusion equation with Dirichlet boundary conditions. The diffusion coefficient is constant while the convection coefficient is piecewise constant. A uniform mesh is imposed on each portion of the domain on which the convection coefficient is constant, but each portion may have a different uniform mesh. Formulas are derived for the exact solution of the matrix equation that arises from the collocation discretization. These formulas possess free “upstreaming/downstreaming” parameters, the values of which may be chosen to yield numerical solutions of great accuracy.

1. INTRODUCTION

It is well known that the numerical solution of convection-diffusion differential equations (DEs) is a difficult task when convection is the dominant process. Numerical techniques (such as finite differences) often give rise to spurious oscillations that are not present in the exact (i.e., not numerical/discrete) solution of the DE. To ameliorate these physically unmeaningful (and therefore undesirable) oscillations, the technique of upstream weighting is often used (Allen [1], Morton [6], Pinder and Shapiro[7], Shapiro and Pinder [10]). While upstreaming can eliminate the oscillations, it is often at the expense of “smearing” the sharp solution profile of the continuous solution of the DE.

In this paper, we study the Hermite collocation solution of the convection-diffusion equation

$$(1) \quad -D \frac{d^2 u}{dx^2} + v \frac{du}{dx} = 0$$

---

1991 *Mathematics Subject Classification.* 65L10, 65L60.

*Key words and phrases.* collocation, variable coefficients, exact solution, convection-diffusion, upstream.

defined on the interval  $(0, 1)$  with given Dirichlet boundary conditions

$$(2) \quad \begin{aligned} u(0) &= u_L \\ u(1) &= u_R. \end{aligned}$$

The coefficients  $v$  and  $D$  in (1) are both positive;  $D$  is constant and  $v$  is piecewise constant. Upon the interval  $[0, 1]$  an *initial* uniform mesh is imposed:

$$(3) \quad 0 = X_0, X_1, X_2, \dots, X_p = 1,$$

where, for  $j = 1, 2, \dots, p$ , we have  $X_j - X_{j-1} = \Delta x$ . We require that  $v$  may change value only at the points  $X_1, X_2, \dots, X_{p-1}$ , and thus write

$$(4) \quad v = \begin{cases} v_1, & \text{if } X_0 < x < X_1 \\ v_2, & \text{if } X_1 < x < X_2 \\ \vdots & \\ v_p, & \text{if } X_{p-1} < x < X_p. \end{cases}$$

To each subinterval  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ , an “upstreaming / downstreaming” parameter  $\zeta_j$  is assigned. Each of the  $p$  subintervals  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ , is refined uniformly via  $X_{j-1} = x_{j,0}, x_{j,1}, x_{j,2}, \dots, x_{j,m_j} = X_j$ , where, for  $k = 1, 2, \dots, m_j$ , we have  $x_{j,k} - x_{j,k-1} = h_j$ . For each subinterval  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ , of the original (i.e., unrefined) mesh, we define a Péclet number  $\beta_j$

$$\beta_j = \frac{v_j h_j}{D},$$

which clearly must be positive. The result of the refinement procedure is the establishment of a mesh of  $M + 1$  nodes upon the interval  $[0, 1]$ , where  $M = \sum_{j=1}^p m_j$ .

In previous papers (Brill [2], Brill [3]), we have studied this same problem in a less complicated context: namely  $v$  being constant and without refined subintervals  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ . Results from these papers will be cited below, as appropriate.

This paper is organized as follows. We first describe the Hermite collocation discretization of the DE (1), both with and without upstream/downstream weighting. We then derive the analytical solution of the matrix equation that arises from the discretization with upstreaming/downstreaming employed. Subsequently, we compare the discrete collocation solution to the continuous solution. In particular, we will discuss a strategy and algorithm to select the upstream/downstream parameters  $\zeta_j$  and refinement parameters  $m_j$ ,  $j = 1, 2, \dots, p$ , in such a way as to avoid spurious oscillations and obtain excellent agreement between the continuous and discrete solutions. We then provide several computational examples which illustrate the theory and efficacy of our algorithm. A short section summarizing our results concludes the paper.

## 2. HERMITE COLLOCATION

The differential equation (1) is defined on the interval  $(0, 1)$  with boundary conditions (2) included. We partition the interval  $[0, 1]$  into  $M$  subintervals as described above. Let us use the notation  $0 = x_0, x_1, x_2, \dots, x_M = 1$ , to indicate the  $M + 1$  nodes of our mesh. (Note that the set of nodes  $x_{i,j}$  and the set of nodes  $x_k$  are identical, but indexed differently.)

The discretization proceeds by introducing a piecewise cubic Hermite interpolating polynomial

$$(5) \quad \widehat{u}(x) = \sum_{k=0}^M [u_k f_k(x) + u'_k g_k(x)]$$

into the DE (1), obtaining

$$(6) \quad -D \frac{d^2 \widehat{u}}{dx^2} + v \frac{d\widehat{u}}{dx} = E(x),$$

where  $E(x)$  is an error function.

The Hermite basis functions, defined for  $\eta \in [-\frac{1}{2}, \frac{1}{2}]$ , are

$$(7) \quad f_k(x) = \begin{cases} \frac{1}{2}(1+2\eta)^2(1-\eta), & x_{k-1} \leq x = x_k + (\eta - \frac{1}{2})(x_k - x_{k-1}) \leq x_k \\ \frac{1}{2}(1-2\eta)^2(1+\eta), & x_k \leq x = x_k + (\eta + \frac{1}{2})(x_{k+1} - x_k) \leq x_{k+1} \\ 0, & \text{otherwise} \end{cases}$$

and

$$(8) \quad g_k(x) = \begin{cases} \frac{x_k - x_{k-1}}{8}(2\eta + 1)^2(2\eta - 1), & x_{k-1} \leq x = x_k + (\eta - \frac{1}{2})(x_k - x_{k-1}) \leq x_k \\ \frac{x_{k+1} - x_k}{8}(2\eta - 1)^2(2\eta + 1), & x_k \leq x = x_k + (\eta + \frac{1}{2})(x_{k+1} - x_k) \leq x_{k+1} \\ 0, & \text{otherwise.} \end{cases}$$

Note that  $\widehat{u}$  in (5) interpolates the values  $u_j = u(x_j)$  and  $u'_j = \frac{du}{dx}(x_j)$ ,  $j = 0, 1, \dots, M$ , because  $f_k(x_j) = \delta_{jk}$ ,  $\frac{df_j}{dx}(x_j) = 0$ ,  $g_k(x_j) = 0$ , and  $\frac{dg_k}{dx}(x_j) = \delta_{jk}$ . Here  $\delta_{jk}$  is the Kronecker symbol. Note also that  $\widehat{u}$  is in  $C^1[0, 1]$ , i.e.,  $\frac{d\widehat{u}}{dx}$  is continuous on  $[0, 1]$ .

It is clear that (6) has  $2(M + 1)$  coefficients, namely  $u_k$  and  $u'_k$ , for  $k = 0, 1, 2, \dots, M$ . However, the imposition of boundary conditions reduces this number to  $2M$ . To generate the  $2M$  equations necessary to find these undetermined coefficients, the traditional choice (known as ‘‘orthogonal collocation’’) is to enforce that the error function  $E(x)$  in (6) be identically zero at two distinct ‘‘collocation points’’ in the interior of each of the  $M$  subintervals.

Given certain smoothness conditions, the optimal (in terms of minimizing local discretization error) location of the collocation points within each subinterval corresponds to the points of Gaussian quadrature (DeBoor [5], Prenter [8]), which in turn corresponds to choosing the collocation points as

$$(9) \quad \eta = \pm \frac{1}{\sqrt{12}}$$

(see (7) and (8)) in each subinterval  $[-\frac{1}{2}, \frac{1}{2}]$  (given in local  $\eta$  coordinates). In our work, this choice will correspond to an absence of upstream/downstream

weighting. However, large Péclet numbers violate the smoothness conditions stipulated in DeBoor [5] and Prenter [8]; thus the Gaussian points are not, in general, optimal for our DE. In fact, use of the Gaussian points (i.e. no upstream/downstream weighting) is optimal (Brill [3]) only for a very small range of values of  $\beta$ .

Upstream/downstream weighting is implemented in the following manner, which was introduced by Allen [1]. As we mentioned above, we have  $2M$  equations in  $2M$  unknowns, where the  $2M$  equations are traditionally generated by forcing  $E(x) = 0$  in (6) at two collocation points in each of the  $M$  subintervals  $[x_{k-1}, x_k]$ ,  $k = 1, 2, \dots, M$ . When implementing upstreaming/downstreaming, we still enforce  $E(x) = 0$  for each of our  $2M$  equations and we still evaluate  $\frac{d^2 \hat{u}}{dx^2}$  in (6) at the Gaussian points  $\eta = \pm \frac{1}{\sqrt{12}}$ . However, we evaluate  $\frac{d\hat{u}}{dx}$  at the points

$$(10) \quad \eta_j = \pm \frac{1}{\sqrt{12}} - \zeta_j,$$

where  $\zeta_j$  controls the amount of upstreaming/downstreaming that occurs in the subinterval  $[X_{j-1}, X_j]$ . Note that regardless of whether (9) or (10) is considered, the the collocation points defined by (9) and (10) lie in this subinterval  $[X_{j-1}, X_j]$ . Note also that  $\zeta_j > 0$  corresponds to upstreaming in  $[X_{j-1}, X_j]$  while  $\zeta_j < 0$  corresponds to downstreaming in  $[X_{j-1}, X_j]$ . (In the rest of this work, we will permit the phrases “upstream” and “upstreaming” to not exclude the possibility that it is actually downstreaming that is occurring.) Because the support of each basis function  $f_k$  or  $g_k$  (see (7) and (8)) is the interval  $[-\frac{1}{2}, \frac{1}{2}]$ , it is clear that each  $\zeta_j$  must satisfy

$$(11) \quad -\zeta_{\max} \leq \zeta_j \leq \zeta_{\max},$$

where  $\zeta_{\max} = \frac{1}{2} - \frac{1}{\sqrt{12}}$ .

It is straightforward to see that choosing the collocation points in this manner leads to a matrix equation with the repeated computational molecule

$$(12) \quad \begin{bmatrix} M_{11}^{(j)} & M_{12}^{(j)} & -M_{11}^{(j)} & M_{14}^{(j)} \\ M_{21}^{(j)} & M_{22}^{(j)} & -M_{21}^{(j)} & M_{24}^{(j)} \end{bmatrix} \begin{bmatrix} q_k \\ r_k \\ q_{k+1} \\ r_{k+1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$k = 0, 1, 2, \dots, M - 1$ . Here  $q_k = u_k$  and  $r_k = u'_k$ ,  $k = 0, 1, 2, \dots, M$ . Note that the matrix equation represented by (12) is a system of  $2M$  equations in

$2(M + 1)$  unknowns. The entries of the matrix are

$$\begin{aligned}
M_{11}^{(j)} &= \frac{2\sqrt{3}D}{h_j^2} + \frac{v_j}{h_j} \left( 6\zeta_j^2 + 2\sqrt{3}\zeta_j - 1 \right) \\
M_{21}^{(j)} &= -\frac{2\sqrt{3}D}{h_j^2} + \frac{v_j}{h_j} \left( 6\zeta_j^2 - 2\sqrt{3}\zeta_j - 1 \right) \\
M_{12}^{(j)} &= \frac{D}{h_j}(1 + \sqrt{3}) + v_j \left( \frac{\sqrt{3}}{6} + \zeta_j + \sqrt{3}\zeta_j + 3\zeta_j^2 \right) \\
M_{22}^{(j)} &= \frac{D}{h_j}(1 - \sqrt{3}) + v_j \left( -\frac{\sqrt{3}}{6} + \zeta_j - \sqrt{3}\zeta_j + 3\zeta_j^2 \right) \\
M_{14}^{(j)} &= \frac{D}{h_j}(-1 + \sqrt{3}) + v_j \left( -\frac{\sqrt{3}}{6} - \zeta_j + \sqrt{3}\zeta_j + 3\zeta_j^2 \right) \\
M_{24}^{(j)} &= -\frac{D}{h_j}(1 + \sqrt{3}) + v_j \left( \frac{\sqrt{3}}{6} - \zeta_j - \sqrt{3}\zeta_j + 3\zeta_j^2 \right),
\end{aligned}$$

which reduce to those given in Brill [2] for the case of  $\zeta = 0$ . The relationship between  $j$  and  $k$  in (12) is that the nodes  $x_k$  and  $x_{k+1}$  both lie in the subinterval  $[X_{j-1}, X_j]$ .

### 3. ANALYTICAL SOLUTION OF UPSTREAM HERMITE COLLOCATION

We start with the matrix equation (12), wherein which we assume (as stated in the previous sentence) that the nodes  $x_k$  and  $x_{k+1}$  both lie in the subinterval  $[X_{j-1}, X_j]$ . We thus make the identification  $x_k = x_{j,i}$  and  $x_{k+1} = x_{j,i+1}$  where  $i \in \{0, 1, \dots, m_j - 1\}$ .

We now manipulate (12), replacing its two equations with linear combinations of them (details are in Brill [3]) to arrive at

$$(13) \quad \begin{bmatrix} 0 & \lambda_j^{(\text{num})} & 0 & -\lambda_j^{(\text{den})} \\ -A_j & C_j + B_j & A_j & C_j - B_j \end{bmatrix} \begin{bmatrix} q_{j,i} \\ r_{j,i} \\ q_{j,i+1} \\ r_{j,i+1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

where

$$\begin{aligned}
A_j &= \frac{\beta_j}{h_j} (1 - 6\zeta_j^2), \\
B_j &= 1 + \beta_j\zeta_j, \\
C_j &= 3\beta_j\zeta_j^2,
\end{aligned}$$

and  $\lambda_j^{(\text{num})}$  and  $\lambda_j^{(\text{den})}$  are, respectively, the numerator and denominator of

$$(14) \quad \lambda_j = \frac{\beta_j^2 + 6\beta_j + 12 + 6\beta_j\zeta_j(4 + \beta_j + \beta_j\zeta_j)}{\beta_j^2 - 6\beta_j + 12 + 6\beta_j\zeta_j(4 - \beta_j + \beta_j\zeta_j)}.$$

From the first equation in (13) it is easy to see that

$$(15) \quad \lambda_j = \frac{r_{j,i+1}}{r_{j,i}},$$

which leads to

$$(16) \quad r_{j,i} = \lambda_j^i r_{j,0}.$$

Now we manipulate the second equation in (13), utilize (15) and (16), and arrive at

$$(17) \quad q_{j,i+1} = q_{j,i} - \alpha_j \lambda_j^i,$$

where

$$(18) \quad \alpha_j = \frac{(C_j + B_j) + (C_j - B_j)\lambda_j}{A_j} r_{j,0}.$$

From (17) and the formula for the sum of a finite geometric series we obtain

$$(19) \quad q_{j,i} = q_{j,0} - \alpha_j \frac{1 - \lambda_j^i}{1 - \lambda_j},$$

which, combined with (18) and letting  $i = m_j$ , leads to

$$(20) \quad r_{j,0} = \frac{1 - \lambda_j}{1 - \lambda_j^{m_j}} (q_{j,m_j} - q_{j,0}) \frac{-A_j}{(C_j + B_j) + (C_j - B_j)\lambda_j}.$$

Algebraic manipulation of (20) leads to

$$r_{j,0} = \rho_j \frac{q_{j,m_j} - q_{j,0}}{\lambda_j^{m_j} - 1},$$

where

$$(21) \quad \rho_j = \frac{2\beta_j}{h_j} \frac{1 + \beta_j \zeta_j}{\beta_j^2 \zeta_j^2 + 4\beta_j \zeta_j + 2}.$$

Thus, invoking (16), we obtain

$$(22) \quad r_{j,i} = \rho_j (q_{j,m_j} - q_{j,0}) \frac{\lambda_j^i}{\lambda_j^{m_j} - 1},$$

which gives the value for each  $r_{j,i}$ ,  $i = 0, 1, 2, \dots, m_j$ , in terms of  $q_{j,0}$  and  $q_{j,m_j}$ , the values of the interpolating polynomial (5) at the endpoints of the interval  $[X_{j-1}, X_j]$ .

Finally, introducing (20) into (18) and substituting the result into (19) yields

$$(23) \quad q_{j,i} = q_{j,0} + (q_{j,m_j} - q_{j,0}) \frac{\lambda_j^i - 1}{\lambda_j^{m_j} - 1},$$

which, much like (22), gives the value for each  $q_{j,i}$ ,  $i = 0, 1, 2, \dots, m_j$ , in terms of  $q_{j,0}$  and  $q_{j,m_j}$ , the values of the interpolating polynomial (5) at the endpoints of the interval  $[X_{j-1}, X_j]$ .





#### 4. SELECTION OF THE UPSTREAM PARAMETERS AND GRID SIZE

In this section we describe an algorithm for selecting the values of the upstreaming parameters  $\zeta_j$ ,  $j = 1, 2, \dots, p$ , that provides, in all test cases discussed in Section 5, extremely accurate collocation solutions of (1)(2).

We first describe the permissible range of values of  $\zeta_j$ , given  $\beta_j$ . (This was done by Brill [3] for the case of upstreaming (i.e.,  $\zeta_j \geq 0$ ) only. In this paper we permit downstreaming (i.e.,  $\zeta_j < 0$ ) as well.) Let us begin by recalling that by the geometry of upstream collocation, we have (11). A glance at the derivative of (28),

$$(33) \quad u'_j(x) = \frac{\frac{v_j}{D}[u_j(X_j) - u_j(X_{j-1})] \exp(\frac{v_j}{D}x)}{\exp(\frac{v_j}{D}X_j) - \exp(\frac{v_j}{D}X_{j-1})},$$

reveals that (33) does not change sign as  $x$  varies; thus (28) is monotone. To ensure that the corresponding collocation solution also maintains monotonicity on each subinterval  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ , for which it is defined, we require (see (23)) that  $\lambda_j > 0$ ,  $j = 1, 2, \dots, p$ . Observing (14), we see that  $\lambda_j$  can change sign only across curves in the  $\beta_j$ - $\zeta_j$  plane where either the numerator or denominator of  $\lambda_j$  vanishes.

The numerator of  $\lambda_j$  vanishes on the curves

$$\zeta_j^\pm = -\frac{2}{\beta_j} - \frac{1}{2} \pm \frac{1}{6\beta_j} \sqrt{72 + 36\beta_j + 3\beta_j^2},$$

which can be shown to never intersect the region of interest, namely (11) combined with  $\beta_j > 0$ . On the other hand, the denominator of  $\lambda_j$  vanishes on the curves

$$(34) \quad \zeta_j^\pm = -\frac{2}{\beta_j} + \frac{1}{2} \pm \frac{1}{6\beta_j} \sqrt{72 - 36\beta_j + 3\beta_j^2}.$$

These latter curves do intersect the region of interest; this situation is depicted in Figure 1, where the horizontal lines are the graphs of  $\zeta_j = \pm\zeta_{\max}$ . The other curves in Figure 1 correspond to where the denominator of  $\lambda_j$  is zero. To avoid negative values of  $\lambda_j$ , we must stay below the curves  $\zeta_j = \zeta_{\max}$  and  $\zeta_j^-$  in (34) and stay above the curves  $\zeta_j = -\zeta_{\max}$  and  $\zeta_j^+$  in (34). Since, while in the region of interest, the individual curves of (34) stay very close to the nearby limiting lines  $\zeta_j = \pm\zeta_{\max}$ , it is clear that enforcing the monotonicity of the upstream collocation solution is rather unrestrictive.

As we saw in Section 3, the collocation and exact solutions are functions of the solutions of the tridiagonal matrix equations (25) and (30), respectively. So a reasonable strategy is to enforce that these two tridiagonal matrices are equal. However, it is more convenient to pursue the following equivalent formulation. We scale all but the first and last rows in each of (25) and (30), then set the scaled versions of (25) and (30) equal to each other, obtaining

$$(35) \quad \begin{bmatrix} \frac{-c_i^*}{b_{i+1}^*} & \frac{c_i^*}{b_{i+1}^*} + 1 & -1 \end{bmatrix} = \begin{bmatrix} \frac{-c_i}{b_{i+1}} & \frac{c_i}{b_{i+1}} + 1 & -1 \end{bmatrix},$$

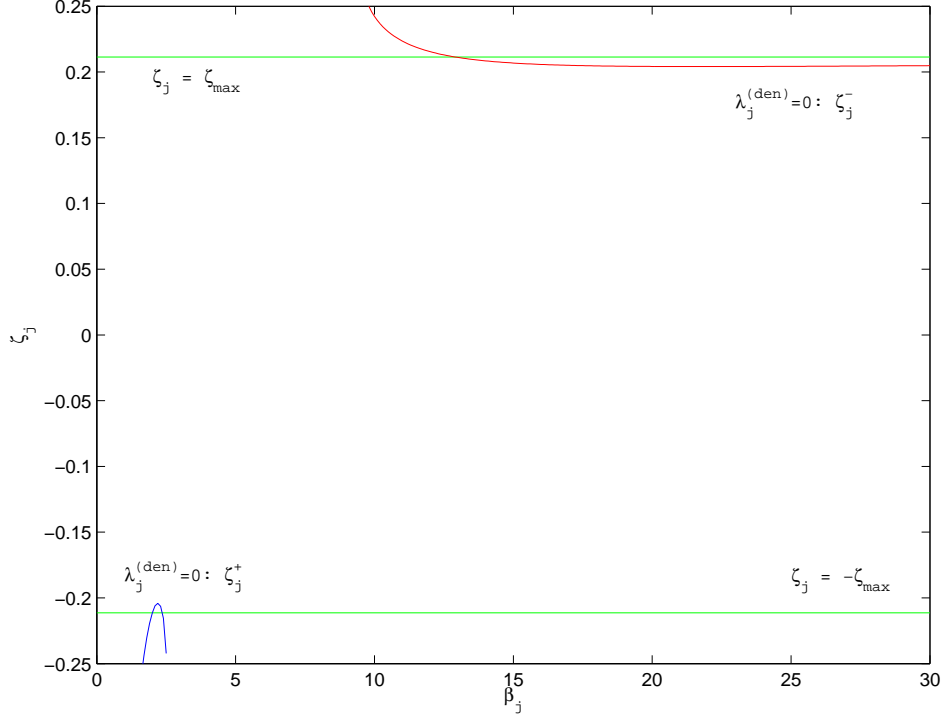


FIGURE 1. Limiting the region of interest to enforce monotonicity. To obtain a monotonic collocation solution, the ordered pair  $(\beta_j, \zeta_j)$  must lie below the curves at the top of this figure and above the curves at the bottom of this figure.

$i = 1, 2, \dots, p - 1$ . Our requirement is therefore that

$$(36) \quad \frac{c_i^*}{b_{i+1}^*} = \frac{c_i}{b_{i+1}}$$

$i = 1, 2, \dots, p - 1$ . Notice that with an obvious bit of algebra we may transform (36) into a polynomial equation in the variables  $\zeta_i$  and  $\zeta_{i+1}$ , where the degrees of  $\zeta_i$  and  $\zeta_{i+1}$  are determined respectively by  $m_i$  and  $m_{i+1}$ . So we see that the system (36),  $i = 1, 2, \dots, p - 1$ , is underdetermined and consists of  $p - 1$  equations in the  $2p$  (as yet) undetermined coefficients  $\zeta_i$  and  $m_i$ ,  $i = 1, 2, \dots, p$ .

We now describe an algorithm that solves (nonuniquely) the underdetermined system mentioned immediately above. To begin, we note that experimentation reveals that when the the values  $v_j$ ,  $j = 1, 2, \dots, p$ , in (4) vary widely (as they do in our numerical experiments of Section 5) the exact solution  $u(x)$  (see (29)) appears to be almost linear on all of its  $p$  “pieces” except for a single piece on which  $u(x)$  exhibits considerable turning (i.e., changes in curvature).

With one exception (Example 8 in Section 5), this is the piece on which  $v$  is greatest. Thus we want the mesh to be sufficiently refined on this piece to resolve this turning. We achieve this goal in the following manner.

Let  $k$  be the index for which  $u_j(x)$ ,  $j = 1, 2, \dots, p$ , (see (29)) has the greatest curvature. This is easily determined once we have solved the tridiagonal matrix equation (30). The curvature  $\kappa$  at a point on the curve  $u = u(x)$  is defined as (Stewart [9])

$$(37) \quad \kappa = \frac{\left| \frac{d^2 u}{dx^2} \right|}{\left[ 1 + \left( \frac{du}{dx} \right)^2 \right]^{3/2}}$$

where (37) is evaluated at the point in question. On each of the  $p$  “pieces”  $[X_{j-1}, X_j]$  of the solution, (37) is a continuous function defined on a closed interval and thus achieves a maximum somewhere in that interval. This location is at one of the two endpoints of that interval, or at an interior point whose coordinate is easily seen to be

$$(38) \quad \frac{D}{v_j} \log \left( \frac{D}{\sqrt{2}v_j|U_j - U_{j-1}|} \left[ \exp \left( \frac{v_j}{D} X_j \right) - \exp \left( \frac{v_j}{D} X_{j-1} \right) \right] \right).$$

(Of course, there is no guarantee that (38) actually lies in  $[X_{j-1}, X_j]$ ). Since we have finitely many ( $p$ ) such intervals, (37) attains a global maximum somewhere in the interval  $[0, 1]$  and therefore on some particular “piece”  $[X_{k-1}, X_k]$ . (In the case where two or more pieces attain the same maximum curvature, we choose the piece with the largest index to define  $k$ .)

On this  $k$ th piece, we select  $m_k$  to be the smallest positive integer that is sufficiently large (this forces  $h_k$  and thus  $\beta_k$  to be sufficiently small) so that there exists  $\zeta_k$  in the region of monotonicity depicted in Figure 1 such that

$$(39) \quad \lambda_k = e^{\beta_k}$$

(see discussion at the end of Section 3). Note that enforcing (39) and (36) for  $i = 1, 2, \dots, p-1$ , results in the collocation solution agreeing exactly with the exact solution at all nodes in the refined  $k$ th piece. That we can find such a  $\zeta_k$  is evident if we solve (39) for  $\zeta_k$ , obtaining

$$(40) \quad \zeta_k^\pm = \frac{12 - 12e^{\beta_k} + 3\beta_k + 3\beta_k e^{\beta_k} \pm \sqrt{Q}}{6\beta_k(e^{\beta_k} - 1)},$$

where

$$Q = 72e^{2\beta_k} - 144e^{\beta_k} - 36\beta_k e^{2\beta_k} + 72 + 36\beta_k + 3\beta_k^2 + 30\beta_k^2 e^{\beta_k} + 3\beta_k^2 e^{2\beta_k}.$$

We now graph  $\zeta_k^+$  in (40) (i.e., a curve on which  $\lambda_k = \exp \beta_k$ ) in Figure 2, where the curves from Figure 1 are included. This  $\zeta_k^+$  curve begins at  $\beta_k = 0$  and ends at

$$\beta_k = \beta_{\text{crit}} \approx 2.862473215,$$

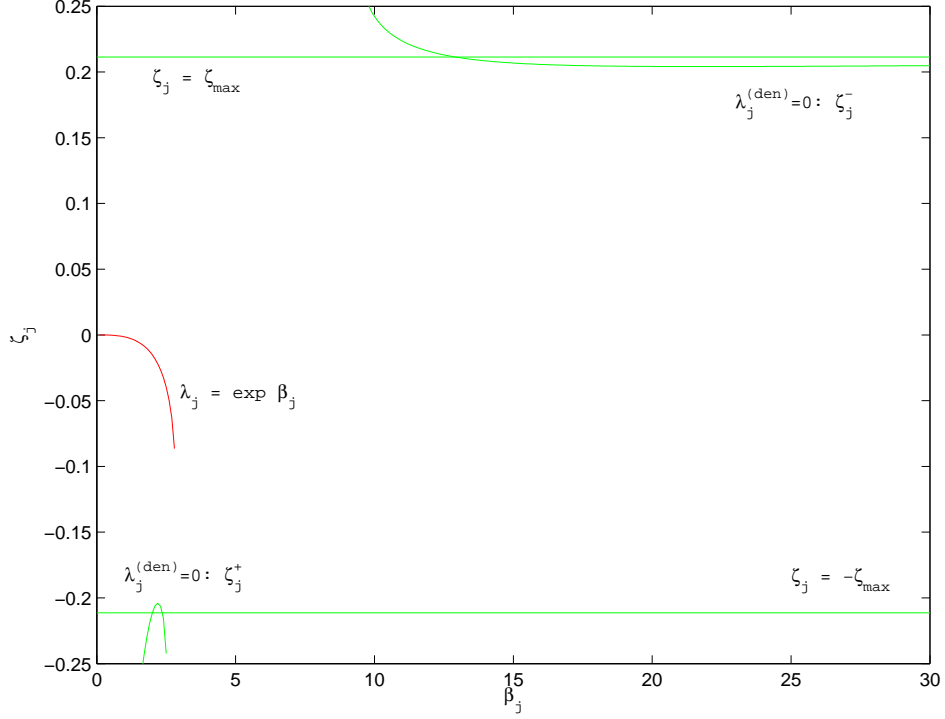


FIGURE 2. The region of monotonicity and a curve on which  $\lambda_j = e^{\beta_j}$

because  $Q$  changes from positive to negative as  $\beta_k$  increases through this value. And clearly this curve lies in the desired region of monotonicity. Thus, if  $0 < \beta_k < \beta_{\text{crit}}$ , we can find  $\zeta_k = \zeta_k^+$  so that the ordered pair  $(\beta_k, \zeta_k)$  lies in the region of monotonicity and such that this ordered pair provides (39). Thus we choose  $m_k$  such that

$$(41) \quad m_k = \left[ \frac{v_k}{pD\beta_{\text{crit}}} \right].$$

This choice defines a value for  $h_k$  and thus for  $\beta_k$  such that  $0 < \beta_k < \beta_{\text{crit}}$ . And using this resulting value of  $\beta_k$  in (40) gives a value of  $\zeta_k^+$  that satisfies our specified criteria. Furthermore, it should be noted that any value of  $m_k$  greater than (41) will also produce a suitable  $\zeta_k^+$ . Thus the value (41) should be considered to be an initial value for  $m_k$  and not necessarily its value at the conclusion of our algorithm.

We presently have values for  $m_k$  and  $\zeta_k$  and now setting  $m_j = 1$  for all  $j = 1, 2, \dots, p$  different from  $k$  provides that the system represented by (36)

is now a system of  $p - 1$  equations in  $p - 1$  unknowns. (Note that this choice  $m_j = 1$  means that there are precisely two mesh points (namely  $X_{j-1}$  and  $X_j$ ) for each subinterval  $[X_{j-1}, X_j]$ , a choice which is consistent with the almost linear nature of the exact solution away from the piece of maximum curvature.). Furthermore, since the  $\zeta$ 's that appear in (36) (with index  $i$ ) are  $\zeta_i$  and  $\zeta_{i+1}$ , our system may be solved sequentially in a straightforward manner, much in the way one solves triangular matrix equations by using "back substitution" or "forward substitution". Unless  $k = 1$  or  $p$ , both forward and back substitution will be necessary. Recall that each equation in (36) may be viewed as a polynomial equation with real coefficients; thus each of the  $p - 1$  equations is easily solved numerically over the field of real numbers via Newton's method. (If convergence of Newton's method is not reached in 50 iterations, then the final estimate of the solution is taken as the solution. Since this final estimate (in the absence of convergence) is typically quite large, this value for the upstreaming parameter will fall outside of the region of monotonicity, and thus this solution is rejected, as described immediately below.) For each of the  $p - 1$  equations, we seek the solution  $\zeta_j$  of minimum absolute value so as to maximize the likelihood that the corresponding ordered pair  $(\beta_j, \zeta_j)$  lies in the region of monotonicity. If we complete the forward/back substitution process with each ordered pair  $(\beta_j, \zeta_j)$ ,  $j = 1, 2, \dots, p$ , in the region of monotonicity, then we are done, as we have found values  $\zeta_j$  and  $m_j$ ,  $j = 1, 2, \dots, p$ , that satisfy (36),  $i = 1, 2, \dots, p - 1$ . If not, we refine our mesh and restart the algorithm with the newly refined mesh. The description of this refinement immediately follows.

When the present values of  $m_j$ ,  $j = 1, 2, \dots, p$ , do not yield satisfactory values for  $\zeta_j$ ,  $j = 1, 2, \dots, p$ , we refine the mesh (i.e., increase some of the values  $m_j$ ,  $j = 1, 2, \dots, p$ , as described below), in the hope that the algorithm described above, when applied to the refined mesh, will decrease the values of  $|\zeta_j|$ ,  $j = 1, 2, \dots, p$ , so that they will all lie in the region of monotonicity. We hope that increasing only one of the values of  $m_j$ ,  $j = 1, 2, \dots, p$ , will lead to improvement in the values of  $\zeta_j$ ,  $j = 1, 2, \dots, p$ , where improvement means a decrease in the absolute value of the  $\zeta_j$ 's that fall outside the region of monotonicity. For  $i, j = 1, 2, \dots, p$ , we define

$$m_{ij}^{\text{temp}} = \begin{cases} m_j + \mu, & \text{if } i = j \\ m_j, & \text{if } i \neq j. \end{cases}$$

Then, for each  $i = 1, 2, \dots, p$ , we use the values  $m_{ij}^{\text{temp}}$ ,  $j = 1, 2, \dots, p$ , in the algorithm described above to find  $\zeta_j^{(i)}$ . Here  $\zeta_j^{(i)}$  means the value of  $\zeta_j$  with the proposed values of  $m_{ij}^{\text{temp}}$ ,  $j = 1, 2, \dots, p$ , for each  $m_i$ ,  $i = 1, 2, \dots, p$ . (Initially  $\mu = 1$ .) We then compute, for each  $i = 1, 2, \dots, p$ , the quantities

$$(42) \quad w_i = \max_{j=1}^p |\zeta_j^{(i)}|$$

and

$$(43) \quad W = \min_{i=1}^p w_i.$$

If  $W$  is sufficiently smaller than

$$(44) \quad \max_{j=1}^p |\zeta_j|$$

then we change the value of  $m_i$  to  $m_{ii}^{\text{temp}}$  and the values of  $\zeta_j$  to  $\zeta_j^{(i)}$ ,  $j = 1, 2, \dots, p$ , where  $i$  is the index in the “minimax” operation (42)(43) which produced  $W$ . In other words, refining the subinterval  $[X_{i-1}, X_i]$  by increasing  $m_i$  by  $\mu$  led to a significant decrease in (44).

Now, what constitutes  $W$  being “sufficiently smaller” than (44), as mentioned in the previous paragraph? Let  $\varphi$  assume the initial value 0.1. Then “sufficiently smaller” means

$$(45) \quad \frac{\left| W - \max_{j=1}^p |\zeta_j| \right|}{\max_{j=1}^p |\zeta_j|} < \varphi.$$

If  $W$  fails to satisfy (45), then  $\varphi$  is multiplied by 0.1 and  $\mu$  is incremented by unity and we find a new  $W$ . When a suitable  $W$  is found (i.e., one that satisfies (45)), then  $\varphi$  and  $\mu$  return to their original values.

The only exception to this algorithm is when we encounter the circumstance that  $W$  and  $Z$  have opposite signs and  $Z$  is outside the region of monotonicity (here  $Z$  is the  $\zeta_j$  that produces  $\max_{j=1}^p |\zeta_j|$ ). Let  $\ell$  be the index that produces this maximum. In this case,  $\zeta_\ell$  was above (respectively, below) the region of monotonicity at one iteration of trying to find a suitable  $W$  but below (respectively, above) it at the following iteration. Thus the mesh is too coarse to obtain a suitable  $W$ . To handle this, we increment each  $m_j$ ,  $j = 1, 2, \dots, p$ , by 1, reset  $\varphi$  and  $\mu$  to their original values, and begin the algorithm anew.

The philosophy behind the algorithm lies in the observations made at the end of Section 3, namely that we obtain approximate equality of (26) and (31) and of (27) and (32) by enforcing  $\beta_j \approx 0$  and  $\zeta_j \approx 0$ . Each time the algorithm fails to find a complete set of suitable  $\zeta_j$ 's,  $j = 1, 2, \dots, p$ , the mesh is refined, decreasing at least one of the  $h_j$ 's which provides a corresponding decrease in  $\beta_j$ . This in turn drives the  $\zeta_j$ 's to decrease in absolute value, since they must satisfy the system given by (36).

## 5. COMPUTATIONAL EXAMPLES

In this section, we provide and discuss computational examples that illustrate the theory and algorithm described above. In all cases, the boundary conditions (2) are

$$\begin{aligned} u(0) &= 1 \\ u(1) &= 0. \end{aligned}$$

and the domain  $[0, 1]$  is initially subdivided into  $p = 4$  pieces. The diffusion coefficient  $D$  assumes two different values: 5 on some examples and 1 on others. The convection coefficients  $v_j$ ,  $j = 1, 2, \dots, p$ , take on the values from the set  $\{0.1, 1, 10, 100\}$ . Specifically, we will report on twelve examples, where the values of the  $v_j$ 's are given in Table 1 for  $D = 5$  and in Table 2 for  $D = 1$ . Note that the examples numbered 1-6 are almost identical to those numbered 7-12, with the only difference being the value of the coefficient  $D$ . Note also that in each example, the  $v_j$ 's vary over four orders of magnitude. For each  $v_j$ , the corresponding values  $m_j$  and  $\zeta_j$ , as determined by our algorithm, are reported. The penultimate column of the tables, labeled "max error" is computed by

$$(46) \quad \max_{i=0}^M |u(x_i) - q_i|.$$

Finally, in the last column, we give an other version of "max error," this one using central finite differences instead of collocation, as the method of discretization. As in the collocation results, we force the finite difference approximation to provide perfect agreement with the exact solution at the breakpoints (3). The finite difference approximations for the derivatives in (1) are

$$(47) \quad \frac{du}{dx}(x_j) \approx \frac{-\bar{h}_j^R}{\bar{h}_j^L \Sigma_j} u(x_j - \bar{h}_j^L) + \frac{\bar{h}_j^R - \bar{h}_j^L}{\bar{h}_j^L \bar{h}_j^R} u(x_j) + \frac{\bar{h}_j^L}{\bar{h}_j^R \Sigma_j} u(x_j + \bar{h}_j^R)$$

and

$$(48) \quad \frac{d^2u}{dx^2}(x_j) \approx \frac{2}{\bar{h}_j^L \Sigma_j} u(x_j - \bar{h}_j^L) - \frac{2}{\bar{h}_j^L \bar{h}_j^R} u(x_j) + \frac{2}{\bar{h}_j^R \Sigma_j} u(x_j + \bar{h}_j^R)$$

where  $\bar{h}_j^L = x_j - x_{j-1}$ ,  $\bar{h}_j^R = x_{j+1} - x_j$ , and  $\Sigma_j = \bar{h}_j^L + \bar{h}_j^R$ . The expressions (47) and (48) minimize local discretization error for 3-point approximations on our non-uniform mesh and reduce to familiar approximations for uniform meshes found, for example, in Burden and Faires [4]. When we compare the last two columns of Tables 1 and 2, we see that implementing collocation with the optimal upstreaming parameters obtained by utilizing our algorithm provides much better accuracy than using finite differences, often by many orders of magnitude.

In each of the figures, three different solutions are given. One is the exact solution; another is the solution based on the algorithm as described in Section 4 (given in the legends as the "optimal" solution); the last is the solution obtained if we utilize the algorithm only to the point where we compute (41) and its corresponding upstream parameter  $\zeta_k$ , with all other  $\zeta_j$ 's being zero (given in the legends as the "original" solution). In all examples we note that the exact solution exhibits significant changes in curvature in one subinterval  $[X_{j-1}, X_j]$ ; on the remaining regions we see that the exact solution is almost linear.

Because the exact solution is almost linear on all but one of its pieces and because the "original" solution has sufficient refinement to obtain (39), it is natural to wonder whether the complete algorithm is necessary (i.e., is the

TABLE 1. Values of parameters in the computational examples.  $D = 5$

Example number	Figure number	$D$	$j$	$v_j$	$m_j$	$\beta_j$	$\zeta_j$	max error	fn. diff. max error
1	3	5	1	100.0	41	1.2195e-01	-2.5212e-06	2.9961e-08	2.0472e-05
			2	0.1	8	6.2500e-04	1.9665e-01		
			3	10.0	8	6.2500e-02	-1.9732e-05		
			4	1.0	8	6.2500e-03	1.8982e-03		
2	4	5	1	0.1	1	5.0000e-03	-2.0970e-01	1.3877e-12	7.8680e-05
			2	100.0	24	2.0833e-01	-1.2591e-05		
			3	10.0	1	5.0000e-01	-8.0852e-04		
			4	1.0	1	5.0000e-02	1.2876e-01		
3	5	5	1	10.0	1	5.0000e-01	6.1490e-04	3.5194e-14	2.0327e-04
			2	0.1	1	5.0000e-03	-2.0970e-01		
			3	100.0	24	2.0833e-01	-1.2591e-05		
			4	1.0	1	5.0000e-02	2.0745e-03		
4	6	5	1	0.1	2	2.5000e-03	-7.5084e-02	3.3602e-08	2.3747e-03
			2	1.0	2	2.5000e-02	7.4337e-04		
			3	10.0	2	2.5000e-01	-1.0904e-04		
			4	100.0	17	2.9412e-01	-3.5520e-05		
5	7	5	1	0.1	1	5.0000e-03	-2.0794e-01	1.3071e-09	4.1630e-04
			2	100.0	43	1.1628e-01	2.8183e-06		
			3	0.1	1	5.0000e-03	-2.4613e-02		
			4	100.0	41	1.2195e-01	-2.5212e-06		
6	8	5	1	100.0	9	5.5556e-01	9.0388e-06	5.7117e-08	6.7821e-05
			2	0.1	9	5.5556e-04	-2.0116e-01		
			3	100.0	42	1.1905e-01	-2.3453e-06		
			4	0.1	9	5.5556e-04	2.0089e-01		

TABLE 2. Values of parameters in the computational examples.  $D = 1$

Example number	Figure number	$D$	$j$	$v_j$	$m_j$	$\beta_j$	$\zeta_j$	max error	fn. diff. max error
7	9	1	1	100.0	108	2.3148e-01	-1.7282e-05	5.1206e-07	1.2162e-04
			2	0.1	16	1.5625e-03	2.0408e-01		
			3	10.0	16	1.5625e-01	-2.3114e-05		
			4	1.0	16	1.5625e-02	1.4705e-03		
8	10	1	1	0.1	1	2.5000e-02	-2.0601e-01	1.9519e-09	2.6969e-04
			2	100.0	97	2.5773e-01	-1.4422e-05		
			3	10.0	11	2.2727e-01	-1.6355e-05		
			4	1.0	1	2.5000e-01	-7.4896e-05		
9	11	1	1	10.0	1	2.5000e+00	8.5674e-03	2.1094e-15	2.2833e-04
			2	0.1	1	2.5000e-02	-2.0610e-01		
			3	100.0	54	4.6296e-01	-1.3961e-04		
			4	1.0	1	2.5000e-01	1.7983e-03		
10	12	1	1	0.1	1	2.5000e-02	-1.7290e-01	5.3291e-15	1.4816e-02
			2	1.0	1	2.5000e-01	1.4758e-03		
			3	10.0	4	6.2500e-01	-5.8547e-04		
			4	100.0	36	6.9444e-01	-4.7904e-04		
11	13	1	1	0.1	1	2.5000e-02	-2.0971e-01	6.6613e-15	9.9212e-04
			2	100.0	145	1.7241e-01	8.5101e-06		
			3	0.1	1	2.5000e-02	-4.6766e-03		
			4	100.0	139	1.7986e-01	-8.0962e-06		
12	14	1	1	100.0	18	1.3889e+00	2.6028e-06	8.5851e-07	5.9123e-05
			2	0.1	18	1.3889e-03	-2.0725e-01		
			3	100.0	111	2.2523e-01	-1.5916e-05		
			4	0.1	18	1.3889e-03	2.0571e-01		

“original solution” sufficiently accurate?). A glance at Figures 6 and 10 (and, to a lesser degree, Figure 9) reveals that the complete algorithm is required to obtain highly accurate solutions. Here we clearly see some significant differences between the exact solution and the “original” solution. Furthermore, if we linearly interpolate the data points of the “original” solution, we often fail

to adequately capture the changes in curvature in the exact solution; this is evident to some degree in Figures 3 through 12.

In contrast to the “original” solution, the “optimal” solution, as determined by our algorithm, does an excellent job of capturing the profile of the exact solution in all examples given. We note that the algorithm converges to a solution regardless of where the maximum value of  $v$  or curvature  $\kappa$  occurs.

It is interesting to note (see Tables 1 and 2) that in all examples (except for Example 4, Figure 6), there is at least one subinterval  $[X_{j-1}, X_j]$ ,  $j = 1, 2, \dots, p$ , for which the corresponding upstream parameter  $\zeta_j$  is quite close to  $\pm\zeta_{\max} \approx \pm 0.21132$ . This is an artifact of our algorithm, which halts when all  $\zeta_j$ 's,  $j = 1, 2, \dots, p$ , lie in the region of monotonicity. The last  $\zeta_j$  to achieve this goal typically approaches  $\pm\zeta_{\max}$  rather slowly (with respect to iteration of the algorithm) and thus is rather close to  $\pm\zeta_{\max}$  when the algorithm's stopping criteria are achieved.

We also note that Example 5 (Figure 7) and Example 11 (Figure 13) presented the greatest challenge to our algorithm. These examples are identical except for different values of  $D$ . The  $v_j$ 's in these examples change from 0.1 to 100 to 0.1 to 100. In these examples, the large changes in curvature occur on the subinterval  $[0.75, 1]$  (one of the subintervals on which  $v = 100$ ) and thus significant refinement is required to resolve this. Yet the other interval on which  $v = 100$ , namely  $[0.25, 0.5]$ , is similarly refined, which is wasteful from the point of view that  $[0.25, 0.5]$  does not require the large number of nodes that our algorithm produces to resolve its almost linear nature. It is curious to note that Examples 6 and 12 (Figures 8 and 14, respectively), did not suffer from this drawback even though the change in values of  $v$  for these two examples is very similar to that found in Examples 5 and 11.

## 6. SUMMARY AND CONCLUSIONS

In this work we have derived closed-form formulas for the solution of the matrix equations arising from the Hermite collocation solution of the boundary value problem (1), (2) with piecewise constant convection coefficient and with upstream/downstream weighting implemented on the convective term of (1). These formulas have a number of free parameters (the upstream/downstream coefficients  $\zeta_j$  and refinement coefficients  $m_j$  ( $j = 1, 2, \dots, p$ )). An algorithm is presented in which the values of these free parameters are chosen so as to obtain extremely accurate numerical solutions that capture the significant changes in curvature that characterize the exact solution to our boundary value problem.

## REFERENCES

- [1] M. B. Allen, How Upstream Collocation Works, *Int. J. Num. Meth. Eng.* **19** (1983) 1753–1763.
- [2] S. H. Brill, Analytical Solution of Hermite Collocation Discretization of the Steady-State Convection-Diffusion Equation, *International Journal of Differential Equations and Applications* **4** (2002) 141–155.

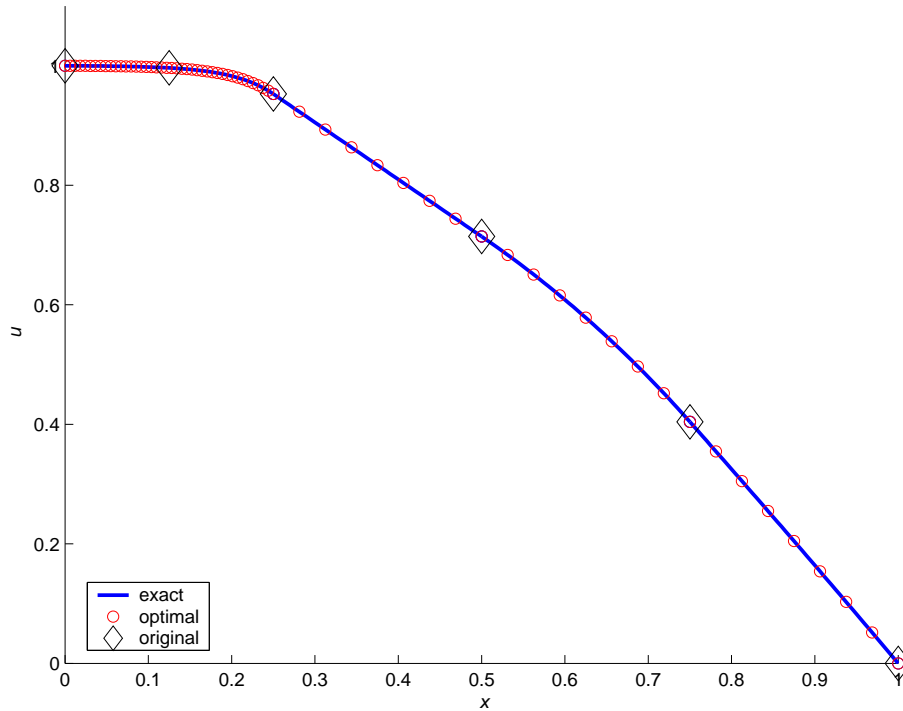


FIGURE 3. Example 1.

- [3] S. H. Brill, Optimal Upstream Collocation Solution of the One-Dimensional Steady-State Convection-Diffusion Equation, *International Journal of Applied Mathematics* **10** (2002) 197–215.
- [4] R. L. Burden and J. D. Faires, *Numerical Analysis*, 8th ed. (Thomson Brooks/Cole, Belmont, CA, 2005).
- [5] C. deBoor, *A Practical Guide to Splines* (Springer-Verlag, Berlin, 1978).
- [6] K. W. Morton, *Numerical Solution of Convection-Diffusion Problems* (Chapman & Hall, London, 1996).
- [7] G. F. Pinder and A. Shapiro, A New Collocation Method for the Solution of the Convection-Dominated Transport Equation, *Water Resources Research* **15** (1979) 1177–1182.
- [8] P. M. Prenter, *Splines and Variational Methods* (John Wiley & Sons, New York, 1975).
- [9] J. Stewart, *Calculus, Early Transcendentals*, 5th ed. (Thomson Brooks/Cole, Pacific Grove, CA, 2003).
- [10] A. Shapiro and G. F. Pinder, Analysis of an Upstream Weighted Collocation Approximation to the Transport Equation, *J. Comp. Phys.* **39** (1981) 46–71.

DEPARTMENT OF MATHEMATICS, BOISE STATE UNIVERSITY, BOISE, IDAHO, USA  
*E-mail address:* `brill@math.boisestate.edu`

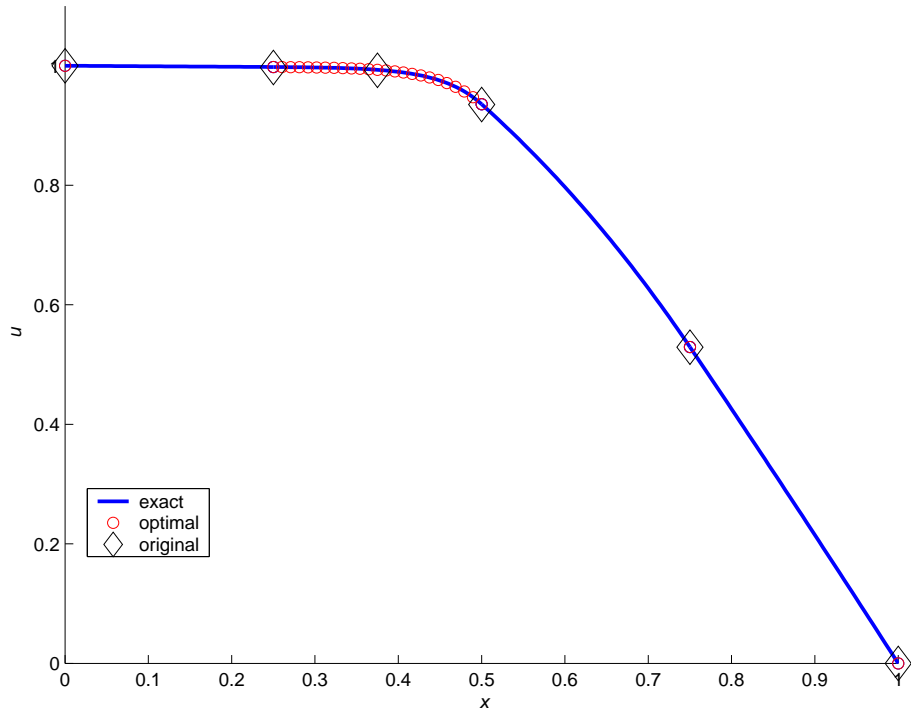


FIGURE 4. Example 2.

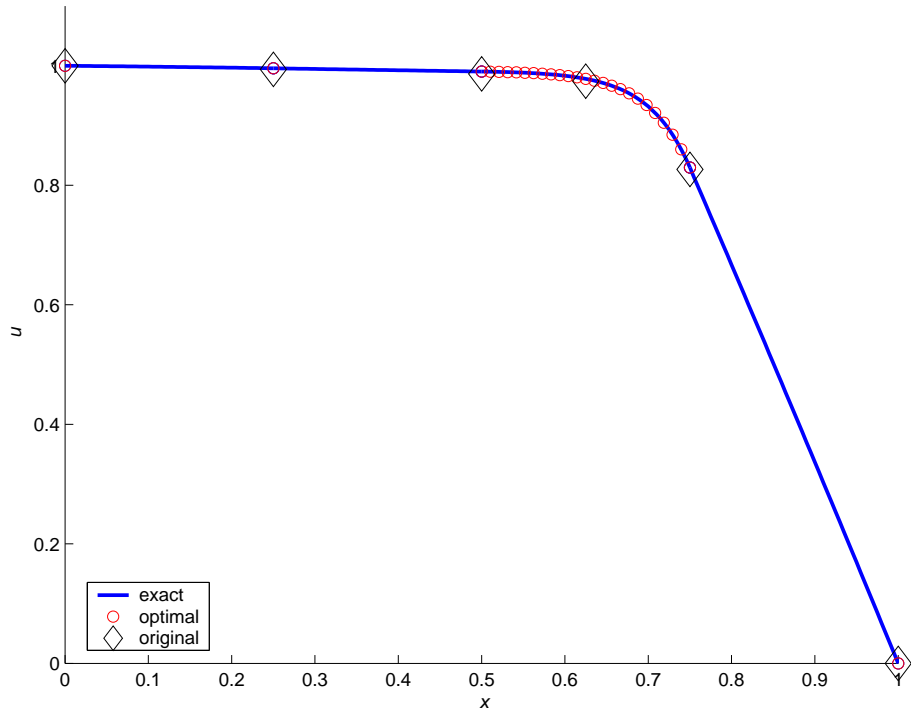


FIGURE 5. Example 3.

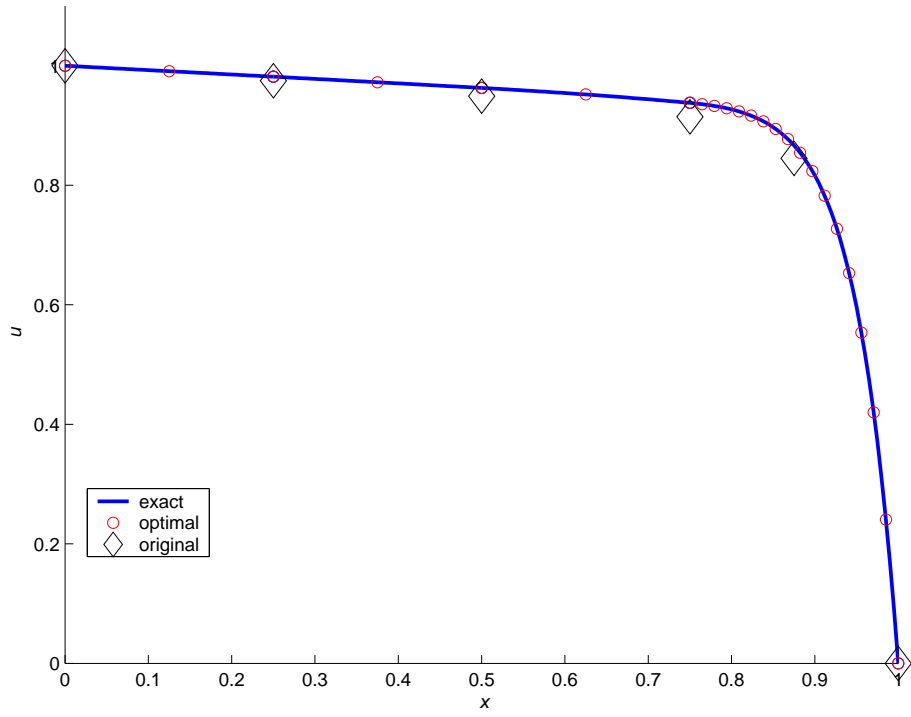


FIGURE 6. Example 4.

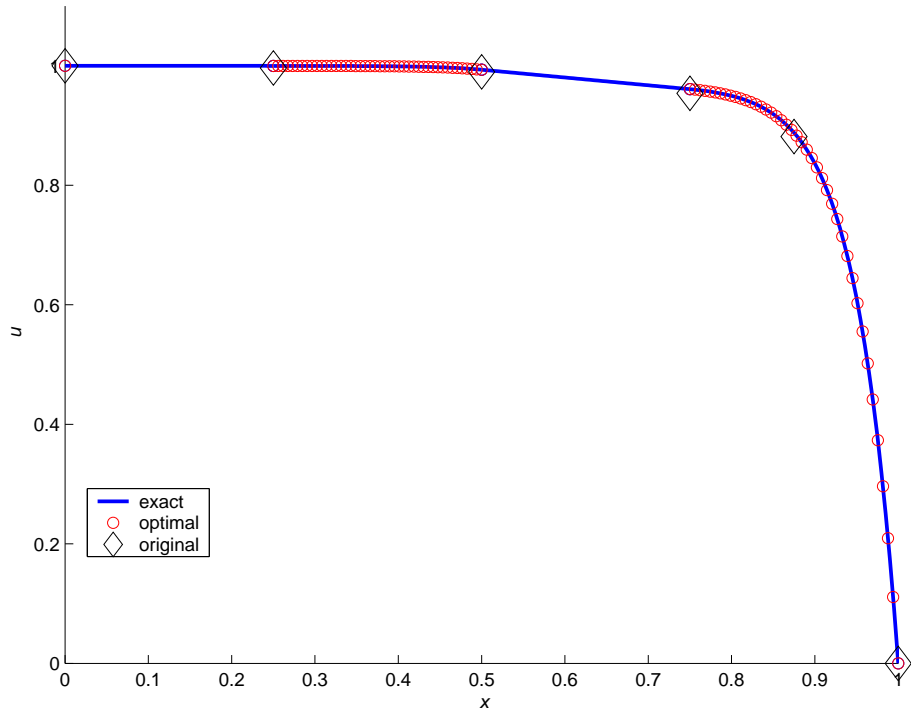


FIGURE 7. Example 5.

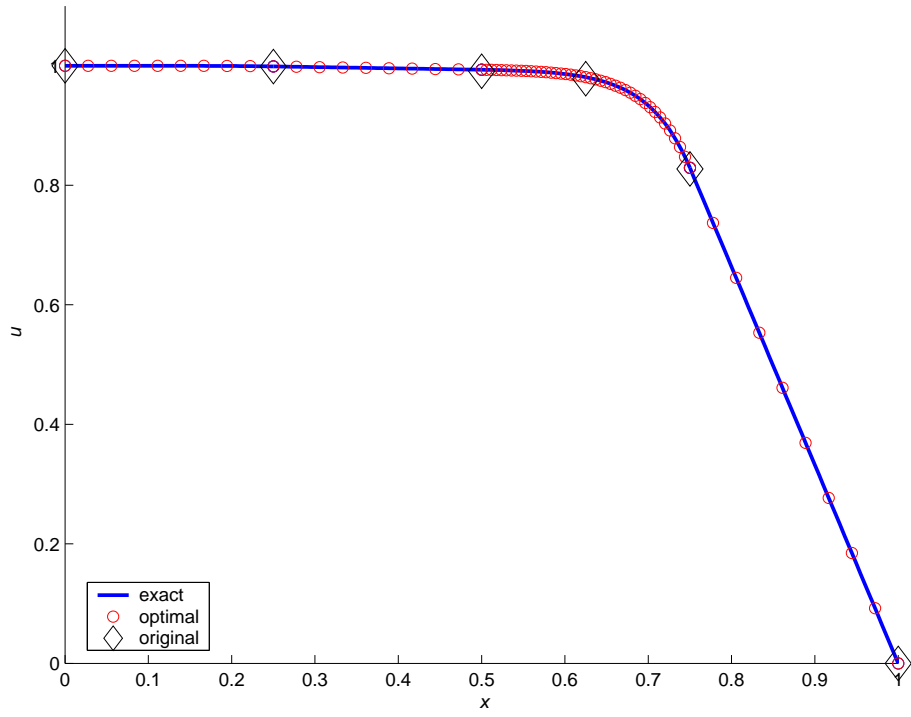


FIGURE 8. Example 6.

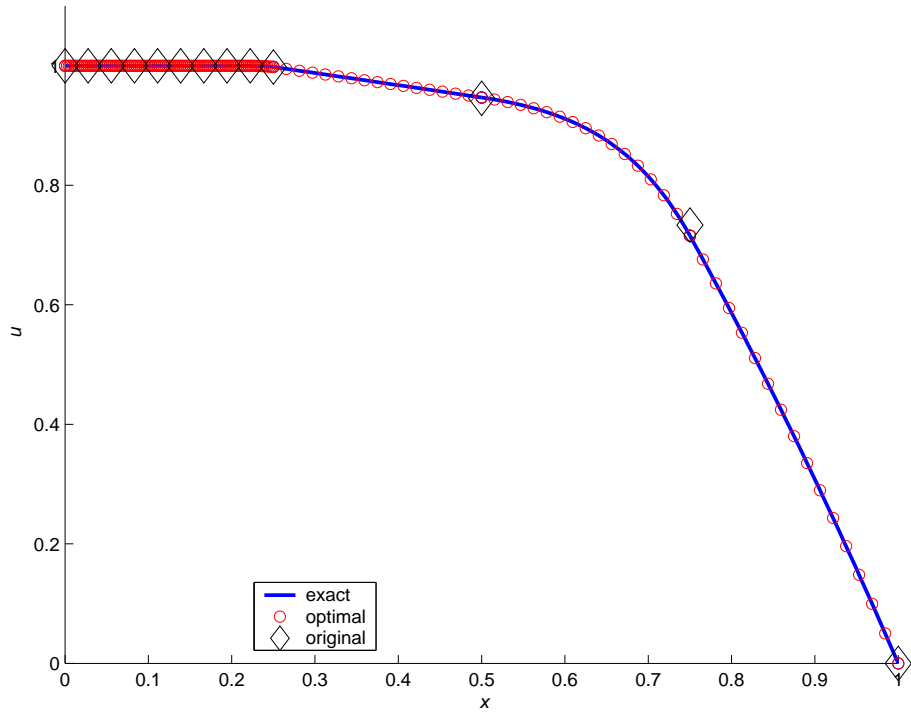


FIGURE 9. Example 7.

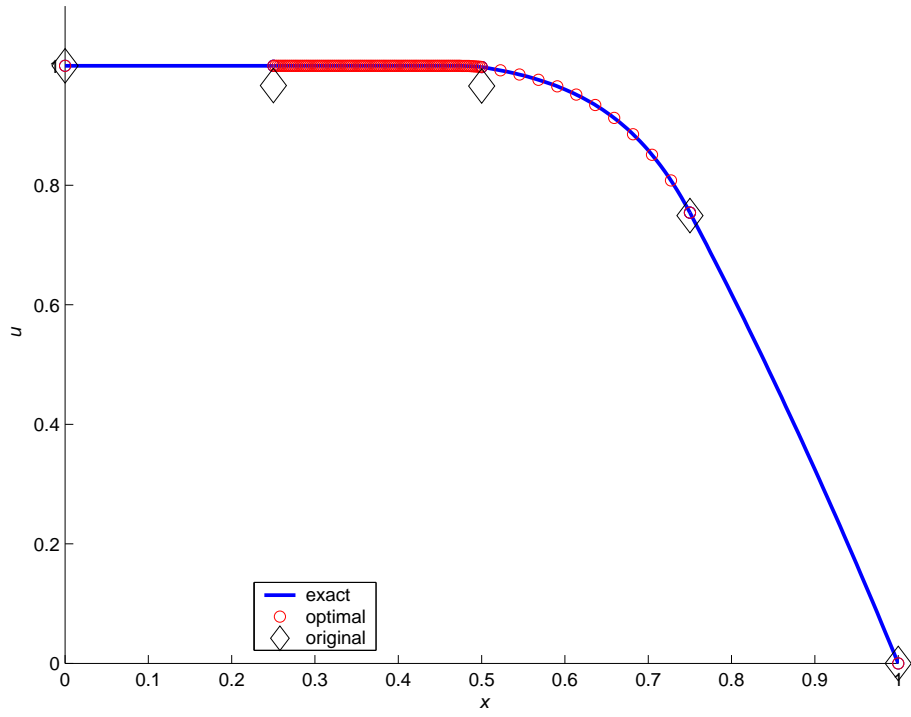


FIGURE 10. Example 8.

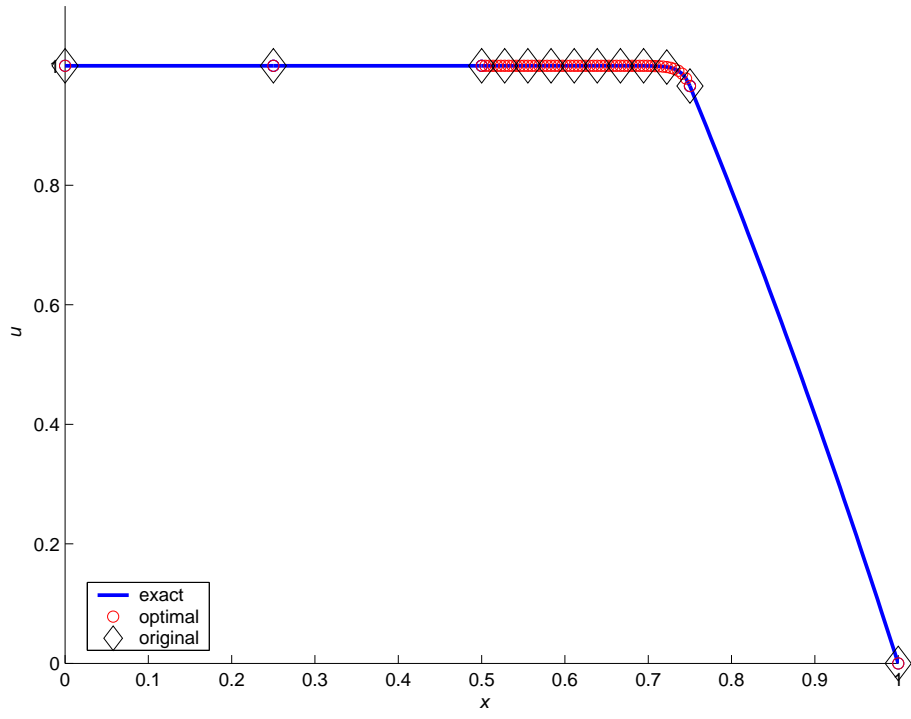


FIGURE 11. Example 9.

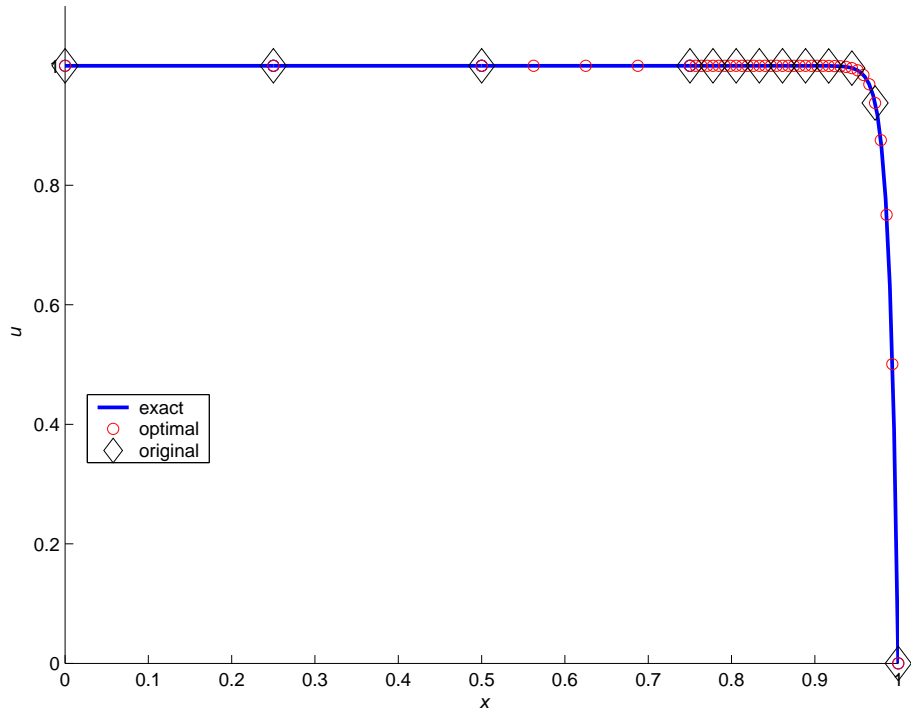


FIGURE 12. Example 10.

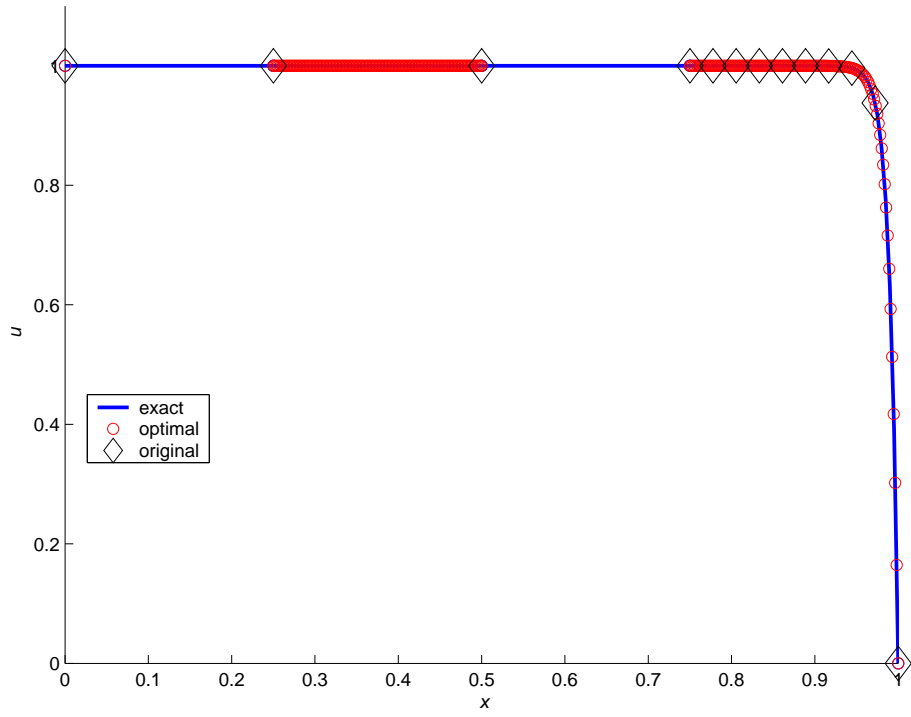


FIGURE 13. Example 11.

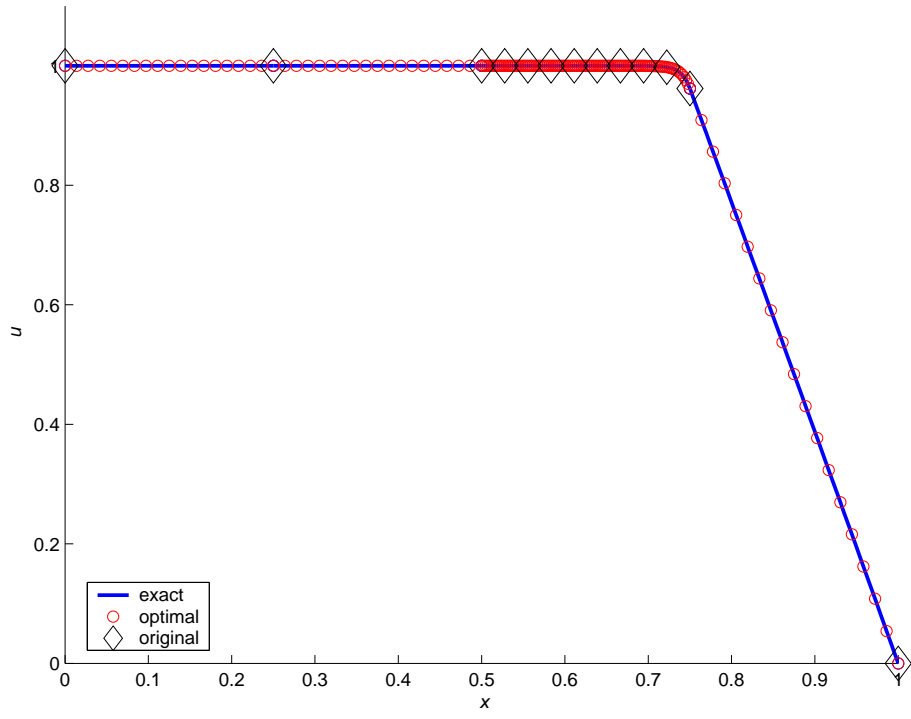


FIGURE 14. Example 12.