



INEEL/CON-02-01471
PREPRINT

**Progress On The Development Of B-Spline
Collocation For The Solution Of Differential
Model Equations: A Novel Algorithm For
Adaptive Knot Insertion**

Richard W. Johnson

May 12, 2003

**Computational Methods and Experimental
Measurements**

This is a preprint of a paper intended for publication in a journal or proceedings. Since changes may be made before publication, this preprint should not be cited or reproduced without permission of the author. This document was prepared as a account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus, product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights. The views expressed in this paper are not necessarily those of the U.S. Government or the sponsoring agency.

Progress on the development of B-spline collocation for the solution of differential model equations: a novel algorithm for adaptive knot insertion.

Richard W. Johnson

Idaho National Engineering and Environmental Laboratory, USA

Abstract

The application of collocation methods using spline basis functions to solve differential model equations has been in use for a few decades. However, the application of spline collocation to the solution of the nonlinear, coupled, partial differential equations (in primitive variables) that define the motion of fluids has only recently received much attention. The issues that affect the effectiveness and accuracy of B-spline collocation for solving differential equations include which points to use for collocation, what degree B-spline to use and what level of continuity to maintain. Success using higher degree B-spline curves having higher continuity at the knots, as opposed to more traditional approaches using orthogonal collocation, have recently been investigated along with collocation at the Greville points for linear (1D) and rectangular (2D) geometries. The development of automatic knot insertion techniques to provide sufficient accuracy for B-spline collocation has been underway. The present article reviews recent progress for the application of B-spline collocation to fluid motion equations as well as new work in developing a novel adaptive knot insertion algorithm for a 1D convection-diffusion model equation.

Introduction

Collocation methods, as applied to the solution of differential equations, employ some form of approximating function, such as a polynomial, to approximate the solution by evaluating the function for sufficient numbers of points in the domain of the solution to provide for the determination of the unknown coefficients that

define the approximating function. The evaluation points are called collocation points. It has been found that using spline curves, or piece-wise polynomials, is more effective in representing the solution to the differential equation than pure polynomials, de Boor [1]. That is, it is more effective to fit polynomials to smaller segments of the solution, than to fit one polynomial to the entire domain of the solution. De Boor [1] shows how to construct and apply a B-spline curve to form the solution to an ordinary differential equation.

Fairweather and Meade [2] provide an extensive bibliography of spline collocation methods and their application to various problems. They indicate that there are four classes of spline collocation methods: nodal, orthogonal, extrapolated/modified and collocation/Galerkin. Nodal collocation employs the junctions of the piece-wise polynomials as the collocation points. Orthogonal collocation, probably the most widely used method, uses the Gauss integration points for the collocation points. The extrapolated/modified method is based on nodal collocation and deferred correction to obtain optimal approximation. The collocation/Galerkin method is a hybrid collocation Galerkin method using the lower continuity between piecewise polynomials of the Galerkin method. Pure collocation methods are said to be more efficient than the hybrid collocation/Galerkin methods, which are more efficient than Galerkin methods [2]. Others workers have employed upwind collocation methods for convection dominated flow problems, see for example, Sun [3], Shapiro and Pinder [4].

The orthogonal collocation method of de Boor [1] employs C^{m-1} B-splines of arbitrary order 'k' to solve an m^{th} order ordinary differential equation. Other researchers, such as Ganesh and Sloan [5] employ C^m B-spline curves/surfaces to solve m^{th} order differential equations. The latter approach reduces the number of unknowns that must be found to produce an approximate solution.

Rubin and Graves [6] were early developers of a cubic spline collocation scheme for solving fluid flow equations. They applied their scheme to the streamfunction-vorticity equations. This approach avoids solving fluid dynamic equations that involve the pressure, which has no boundary conditions. More recently, Lai and Wenston [7] have applied a bi-variate spline collocation method to the streamfunction-vorticity equations, using C^1 cubic splines incorporated into a Galerkin finite element method. Botella [8] applies B-spline collocation to the solution of the Navier-Stokes equations in the velocity-pressure (primitive variable) formulation. Botella employs a fractional step scheme that considers the projection step to be a Div-Grad problem to circumvent the problem of there being no boundary conditions for the pressure.

The present author, Johnson [9], has applied a B-spline collocation method to the solution of the 1D, steady, linear, 2nd-order, convection-diffusion equation using C^3 quartic B-splines with collocation at the Greville abscissae. It was found that Greville collocation using a C^3 quartic B-spline curve yields about the same accuracy as orthogonal collocation using a C^1 cubic B-spline curve. However,

the former method yields significantly lower residuals away from the collocation points and has significantly fewer unknowns than the latter approach.

The present author, Johnson [10], has also applied B-spline collocation at the Greville points to the solution of the 2D, steady, laminar, incompressible Navier-Stokes equations in primitive variable format. The solution for the velocity components and pressure were found by applying an ad-hoc treatment to the boundaries such that the correct pressure field is obtained. It was determined that although the velocity components are specified on the boundaries (Dirichlet boundary conditions), the pressure on the boundaries is critically influenced by the second order derivatives of the velocity components, as indicated by the differential equations. Therefore, it was found that including collocation for the velocity components on the boundaries is essential, even though this leads to an over-specified system of equations. The ad-hoc treatment developed to accomplish this is to sum the absolute values of the residuals for the velocities and the continuity equation in conjunction with finding the unknown de Boor points for the pressure on the boundaries. It was found that C^3 quartic B-spline spline tensor-product surfaces are significantly more accurate than C^2 cubic surfaces for the 2D test problem used (developing channel flow).

The analyst usually specifies the discrete partition or knot vector for the B-spline curve or surface used in the solution of the system of differential equations. However, it is desirable that the partition be found adaptively depending on the nature of the solution curve. Such adaptive refinement must be based on some criterion. Fischer and Bar-Yoseph [11] base an adaptive mesh algorithm on solution gradient and object geometry for solid mechanics. Mohsen and Pinder [12], apply an adaptive mesh algorithm to the unsteady, 1D convection diffusion equation. They use a hybrid spline approach that employs a quintic spline in the region of maximum solution gradient and cubic splines elsewhere. McRae [13] bases his grid adaptation approach on truncation error, especially the solution gradient and solution curvature. These are estimated from finite difference approximations. For a B-spline curve/surface, the curvature can be evaluated exactly at any point.

The present article describes a method used to determine, adaptively, a knot vector to provide suitable accuracy for the 1D, steady, linear convection-diffusion problem. This problem is defined by

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

where D is a constant diffusion coefficient, v is a constant velocity and u represents the transported entity. The problem is defined on the interval $x \in [0, L]$ with Dirichlet boundary conditions given as

$$u(0) = 0, \quad u(L) = 1. \quad (2)$$

The analytical solution to eqn. (1) is

$$\frac{e^{xv/D} - 1}{e^{Lv/D} - 1}. \quad (3)$$

Equations (1-2) describe a problem with a boundary layer at boundary $x = L$. The Peclet number, $Pe = Lv/D$, characterizes the physics of the flow problem: the higher the Peclet number, the more the transport is convection-dominated and the narrower the boundary layer. For the convenience of the reader, a description of the construction and evaluation of B-spline curves is given.

B-spline curves

A B-spline (basis-spline) *curve* is constructed from individual B-splines, on some interval, Hoschek and Lasser [14], and Farin [15]. A k -th order B-spline is a $(k-1)$ degree polynomial. The interval is partitioned into subintervals; the break points between subintervals are called knots. If the knots are given as $t_0 \leq t_1 \leq t_2 \leq \dots \leq t_m$, then a knot vector is defined as $\mathbf{T} = (t_0, t_1, t_2, \dots, t_m)$, where t is a parameter that may or may not coincide with a Cartesian coordinate; for our example, they will coincide. Given a knot vector $\mathbf{T} = (t_0, t_1, t_2, \dots, t_p, t_{p+1}, \dots, t_{p+k})$, a B-spline N_{ik} of order k is defined as:

$$N_{i1} = \begin{cases} 1, & \text{for } t_i \leq t < t_{i+1} \\ 0, & \text{otherwise} \end{cases}, \quad \text{and} \quad (4)$$

$$N_{ik}(t) = \frac{(t - t_i)}{(t_{i+k-1} - t_i)} N_{i,k-1}(t) + \frac{(t_{i+k} - t)}{(t_{i+k} - t_{i+1})} N_{i+1,k-1}(t) \quad (5)$$

for $k > 1$ and $i = 0, 1, \dots, p$. Fourth-order B-splines are referred to as cubic B-splines. Given points \mathbf{d}_i in \mathbf{R}^2 for $i = 0, 1, \dots, p$, and a knot vector \mathbf{T} , a B-spline curve $\mathbf{X}(t)$ in \mathbf{R}^2 is defined as

$$\mathbf{X}(t) = \sum_{i=0}^p \mathbf{d}_i N_{ik}(t) \quad (6)$$

for $p \geq k - 1$. The \mathbf{d}_i are called de Boor points or control points. The B-spline curve has continuity C^{k-m-1} at the knots for knot multiplicity 'm' [14]. The de Boor points \mathbf{d}_i can be located where the user desires. The de Boor points are restricted to lie at the Greville abscissae for the present problem. This means that

the Cartesian coordinate 'x' is coincident with parametric coordinate 't.' The formula for Greville abscissae for an n-degree B-spline curve is [15]

$$\xi_i = \frac{1}{n}(t_i + t_{i+1} + \dots + t_{i+n-1}) \quad (7)$$

where the t_i are the knots. Although B-spline curves can be evaluated using eqns. (4-6), they can also be converted to Bezier spline curves in terms of Bernstein polynomials and evaluated. The latter is employed here because the curves and derivatives are easy to evaluate. The Boehm algorithm [14,15] is used to convert B-spline curves to Bezier curves. The Boehm algorithm inserts temporary knots at existing knot locations. The knot is inserted at an existing knot value t_i using

$$\mathbf{d}_i^\kappa(t) = \frac{t_{i+n-\kappa} - t_I}{t_{i+n-\kappa} - t_{i-1}} \mathbf{d}_{i-1}^{\kappa-1}(t) + \frac{t_I - t_{i-1}}{t_{i+n-\kappa} - t_{i-1}} \mathbf{d}_i^{\kappa-1}(t) \quad (8)$$

for $\kappa = 1, \dots, n-m$, and $i = I - n + \kappa + 1, \dots, I - m + 1$, where \mathbf{d}_i^κ is the new de Boor point at level κ , n is the degree of B-splines used, and m is the multiplicity of the original knot. Sufficient 'temporary' knots are inserted at existing knot locations until multiplicity 'n' is reached. The level κ is equal to zero for the original set of de Boor points. For quartic B-spline curves ($n = 4$), the level 2 and 3 de Boor points become Bezier points, plus the intersection of two adjacent interpolating lines becomes the 5th Bezier point, Hoschek and Lasser [14]. A Bezier curve can be defined in terms of the Bezier points as follows:

$$\mathbf{b}^n(t) = \sum_{j=0}^n \mathbf{b}_j B_j^n(t) \quad (9)$$

where n is the degree of the Bezier curve, $\mathbf{b}^n(t)$ is the Bezier curve, the \mathbf{b}_j are the Bezier points and the $B_j^n(t)$ are Bernstein polynomials. The latter are defined as [15]

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (10)$$

$$\binom{n}{i} = \begin{cases} \frac{n!}{i!(n-i)!} & \text{if } 0 \leq i \leq n \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

The reason for starting with B-spline rather than Bezier curves, is that less information is required to define a B-spline curve. A C^{n-1} continuous spline curve

can automatically be generated using n-degree B-splines. Parametric derivatives of a Bezier curve are obtained from simple formulae. The r^{th} parametric derivative of a Bezier curve defined using Bernstein polynomials is [15]

$$\frac{d^r}{dt^r} \mathbf{b}^n(t) = \frac{n!}{(n-r)!} \sum_{j=0}^{n-r} \Delta^r \mathbf{b}_j B_j^{n-r}(t) \quad (12)$$

where r is the order of the derivative. The difference operator Δ^r for $r = 1, 2$ is

$$\begin{aligned} \Delta^1 \mathbf{b}_i &= \mathbf{b}_{i+1} - \mathbf{b}_i \\ \Delta^2 \mathbf{b}_i &= \mathbf{b}_{i+2} - 2\mathbf{b}_{i+1} + \mathbf{b}_i \end{aligned} \quad (13)$$

Finally, the curvature at a point on a B-spline curve is given by [15]

$$\kappa = \frac{\|\dot{\mathbf{x}} \wedge \ddot{\mathbf{x}}\|}{\|\dot{\mathbf{x}}\|^3} \quad (14)$$

where κ is the curvature, $\mathbf{x}(t)$ is the B-spline curve, \wedge is the cross-product and the dots refer to differentiation with respect to parametric variable 't.'

Adaptive knot insertion

Adaptive knot insertion is the ability to modify the knot vector, i.e. the discrete partition, algorithmically, such that the final solution achieves some desired level of accuracy. The approach used in the present study begins with a knot vector with knots at the endpoints only, that is, one curve segment to fit the whole solution. A solution is obtained, the adaptive knot insertion algorithm is applied, another solution is obtained with the new knot vector, and so forth until a converged knot vector is obtained.

The strategy used in developing an adaptive knot insertion algorithm to limit the size of a knot interval is based on the following criteria:

- the magnitude of the curvature should be incorporated,
- the gradient of the curve should be incorporated, and
- the aspect ratio between adjacent intervals should be limited.

The curvature is important in limiting the size of knot intervals for simple approximation considerations. The curve gradient is important because an approximation to a steeper curve will be more sensitive to a change in the independent variable than if the curve has a smaller gradient. The aspect ratio is thought to be important because the joining of high continuity curve segments is smoother for segments of similar lengths.

The method of employment of the curvature in the determination of the maximum length for a knot interval is the novel aspect of the present knot insertion algorithm. The curvature can be incorporated in a dimensionless way by considering the maximum sum of ‘angular’ segments (in radians) $\delta\theta$ that a curve segment should include. For a circle, an arc length ‘ s ’ is equal to the radius times the angular segment $\Delta\theta$. For an arbitrary curve, the curvature is the rate of change of the angle α between two tangent vectors for two neighboring points on the curve with respect to the arc length s , Farin [15] or $\kappa = d\alpha/ds$. Thus the curvature for a circle is the inverse of the radius. For the present application, the spline curve is divided into small segments ($\Delta x = .0002L$), and the arc lengths computed from:

$$\delta s = \left(\delta x^2 + \delta y^2 \right)^{1/2} \quad (15)$$

The curvature at the center of each curve segment is calculated using eqn. (14) and multiplied by δs , and summed as follows:

$$\Delta\theta = \sum \delta\theta = \sum \kappa \delta s \leq C_{\max} \quad (16)$$

where C_{\max} is a constant determined by numerical experiment. The procedure that has been found to work with the 1D convection-diffusion test problem is to calculate the solution, beginning with a single B-spline curve segment as mentioned earlier, then perform the summation of eqn. (16), inserting a knot each time C_{\max} is reached. The solution is then found based on the new knot vector and the knot insertion procedure is performed again. By the third time the solution is found, a pretty good solution is available. The location of the smallest knot interval is then determined. From the smallest knot interval outward in both directions, a limiting interval expansion is then applied. The limiting expansion formula incorporates the considerations given above concerning both the gradient of the curve and the aspect ratio of adjacent intervals. The limiting expansion formula is

$$\frac{\Delta t_i}{\Delta t_{i-1} \text{ or } i+1} = \frac{1}{E_{\max}} + (E_{\max} - 1)(\cos \beta)^{1/2} \quad (17)$$

where angle β is the angle of the tangent to the curve at the center of the small curve segment with respect to the x-axis. If the tangent to the solution curve becomes parallel to the x-axis, the cosine function becomes unity; the maximum expansion ration in eqn. (17) will then be equal to E_{\max} . As the solution curve tends toward the vertical, the cosine function tends to zero, and the minimum tends to $1/E_{\max}$. Thus, for the limiting expansion formula, the maximum expansion ratio is the inverse of the minimum expansion ratio. Of course, as the gradient of the solution curve increases, the cosine function tends

to zero, limiting the expansion of the knot intervals in the region of a steep solution gradient. The exponent $\frac{1}{2}$ of the cosine function was determined to be effective through numerical experimentation.

Results and discussion

The solution is found using C^3 quartic B-splines. The de Boor points for the B-spline curve are located at the Greville abscissae, eqn. (7). The collocation points are also located at the Greville abscissae. The problem matrix is derived from applying Newton's method to the problem, to obtain $\mathbf{J}\delta\mathbf{x} = -\mathbf{R}$, where \mathbf{J} is the Jacobian, determined numerically, $\delta\mathbf{x}$ is the update vector and \mathbf{R} is the residual vector. The matrix is solved using a direct solver based on LU decomposition.

The present adaptive knot insertion method has been applied to the test problem, eqns. (1,2), for $L = 1$, for three Peclet numbers: 5, 50 and 200. Table 1 provides algorithmic information and results for the three cases. These include the values used for E_{max} , the number of knots in the final solution, the number of solution iterations required for convergence of the discrete partition, and the maximum errors found from error evaluation at 5001 equally spaced points. A value of $C_{max} = 0.2$, see eqn. (16), was found to be suitable based on numerical experiment, for all cases. Figure 1 illustrates the exact solutions for the three Peclet numbers. The adaptive knot insertion algorithm is applied for each case. Figure 2 shows the curvature κ for each of the three cases.

Figure 3 shows the cumulative sum of angular segments for each knot interval for $Pe = 50$. It can be seen that the knot interval at the point of greatest curvature is narrower than adjacent ones, fig. 2. The intervals expand faster to the left than to the right because the gradient is steeper to the right. Also, the intervals near $x = L$ contract. The single parameter-based expansion formula, eqn. (18), with $E_{max} = 1.3$, for $C_{max} = 0.2$, is adequate for $5 \leq Pe \leq 50$ to find accurate solutions. Unfortunately, E_{max} has to be modified for $Pe = 200$. Therefore, additional investigations are called for to determine if a formula can be found that will fit a larger range of Pe numbers. It may be the case that E_{max} should be a function of the gradient. Nevertheless, it is shown that a curvature-based formula for adaptive knot insertion, which includes the influence of solution curve gradient, can provide a suitable partition for an accurate solution for the test problem.

Table 1. Algorithmic information and results for the three test cases

Test Case	E_{max}	knots inserted	solution iter.	max error
Pe = 5	1.3	9	4	3.41×10^{-5}
Pe = 50	1.3	22	4	2.24×10^{-5}
Pe = 200	1.01	182	4	4.06×10^{-5}

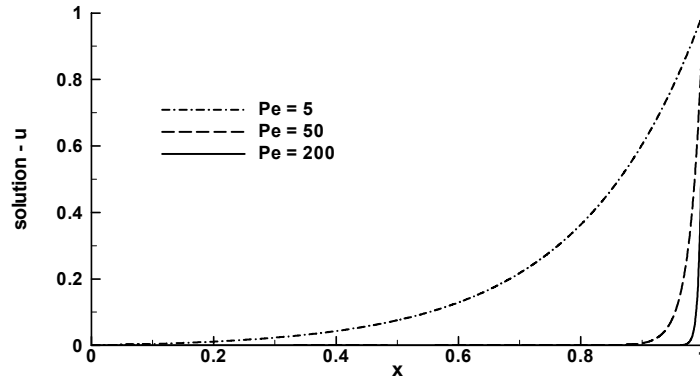


Figure 1. Exact solutions for the test problem for the three Peclet numbers.

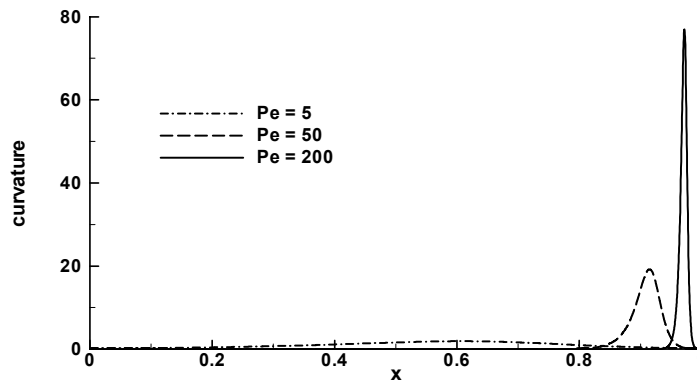


Figure 2. Curvature κ for the test problem for the three Peclet numbers.

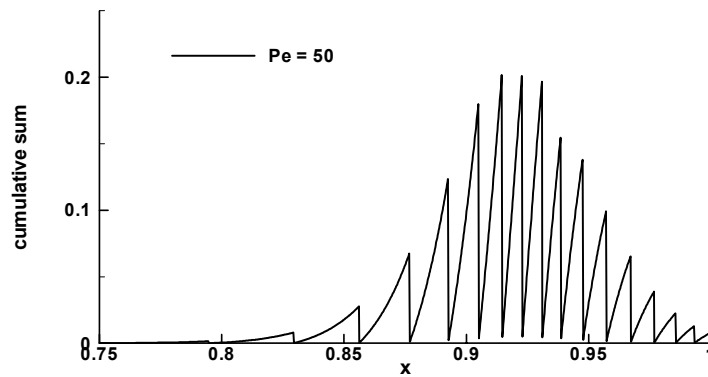


Figure 3. Cumulative $\sum \delta\theta = \sum \kappa\delta s$ for each knot interval for $Pe = 50$.

Acknowledgements

This work was performed at the INEEL under the auspices of the U.S. D.O.E., Idaho Operations, Contract No. DE-AC07-99ID13727 through the LDRD program for the Subsurface Science Initiative.

References

- [1] de Boor, C., *A Practical Guide to Splines*, New York: Springer, 1978.
- [2] Fairweather G., Meade D., A Survey of Spline Collocation Methods for the Numerical Solution of Differential Equations,” in: Diaz JC, ed., *Mathematics for Large Scale Computing*, New York: Marcel Dekker, 1989.
- [3] Sun W., Spectral Analysis of Hermite Cubic Spline Collocation Systems, *SIAM J Num Anal*, **36(6)**, pp. 1962-1975, 1999.
- [4] Shapiro, A., and Pinder, G.F., Analysis of an Upstream Weighted Collocation Approximation to the Transport Equation, *J Comp Phys*, **39**, pp. 46-71, 1981.
- [5] Ganesh, M., and Sloan, I.H., Optimal Order Spline Methods for Nonlinear Differential and Integro-Differential Equations, *Appl Num Math*, **29**, pp. 445-478, 1999.
- [6] Rubin, S.G., and Graves, R.A., Viscous flow solutions with a cubic spline approximation, *Comput & Fluids*, **3**, pp. 1-36, 1975.
- [7] Lai M-J, and Wenston P., Bivariate Spline Method for Numerical Solution of Steady State Navier-Stokes Equations over Polygons in Stream Function Formulation, *Num Meth Part Diff Eq*, **16**, pp. 147-183, 2000.
- [8] Botella, O., On a collocation B-spline method for the solution of the Navier-Stokes equations, *Comput & Fluids*, **31**, pp. 397-420, 2002.
- [9] Johnson, R.W., Higher Order B-Spline Collocation at the Greville Abcissae, submitted for review, 2002.
- [10] Johnson, R.W., A B-spline Collocation Method for Solving the Navier-Stokes Equations, submitted for review, 2002.
- [11] Fischer, A., and Bar-Yoseph, P.Z., Adaptive mesh generation based on multi-resolution quadtree representation, *Int J Num Meth Eng*, **48**, pp. 1571-1582, 2000.
- [12] Mohsen, M.F., and Pinder, G.F., Orthogonal collocation with ‘Adaptive’ finite elements, *Int J Num Meth Eng*, **20**, pp. 1901-1910, 1984.
- [13] McRae, D.S., r-Refinement grid adaptation algorithms and issues, *Comput Meth Appl Mech Eng*, **189**, pp. 1161-1182, 2000.
- [14] Hoschek J., Lasser D., *Fundamentals of Computer Aided Geometric Design*, trans. by Schumaker LL, Wellesley, Mass., A. K. Peters, 1993.
- [15] Farin G., *Curves and Surfaces for Computer Aided Geometric Design*, 4th ed., San Diego, Academic Press, 1997.