JOURNAL OF COMPUTATIONAL PHYSICS 93, 426–448 (1991)

Basis-Spline Collocation Method for the Lattice Solution of Boundary Value Problems

A. S. UMAR

Department of Physics and Astronomy, Vanderbilt University, Nashville, Tennessee 37235

J. WU, M. R. STRAYER, AND C. BOTTCHER

Physics Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831

Received August 10, 1989; revised December 12, 1989

We study a particular utilization of the basis-spline collocation method (BSCM) for the lattice solution of boundary value problems. We demonstrate the implementation of a general set of boundary conditions. Among the selected problems are the Schrödinger equation in radial coordinates, the Poisson, and the generalized Helmholtz equations in radial and three-dimensional Cartesian coordinates. © 1991 Academic Press, Inc.

1. INTRODUCTION

With the advent of computer technology it has now become feasible to perform highly accurate calculations for complex problems commonly encountered in physics. A class of problems that would greatly benefit from the development of new numerical approaches has been recently discussed in Ref. [1]. Typically, large scale calculations in physics involve solutions of complex, non-linear dynamical systems on a space-time lattice. Currently, most of these calculations are performed using low order finite-difference lattice techniques. Some of the more compute-intensive problems contain natural length scales which render the choice of equi-distant grid points computationally expensive. For these problems the typical discretization involves non-equidistant lattice points which require the development of new methods for the accurate representation of the relevant differential operators on the lattice. Thus, it is desirable to investigate higher order interpolation methods which result in the improvement of the overall accuracy and reduction in the total number of lattice points.

The *lattice* solution of differential equations may be viewed to proceed in two steps:

(1) Obtain a discrete representation of the functions and operators on the lattice.

0021-9991/91 $3.00

(2) Solve the resulting lattice equations using iterative or elimination techniques.

Step (1) is an approximation problem for which we could take advantage of the techniques developed using the basis-spline functions [2]. The use of the basis-spline collocation method (BSCM) for the lattice solution of differential equations leads to a matrix–vector representation on the collocation lattice with a metric describing the transformation properties of the collocation lattice. In most physical applications, equations of motion can be obtained via the variation of the lattice representations of the constants of the motion. In this *variation after discretization* approach, equations of motion exactly preserve the constants of the motion [3].

Alternatively, one may use what we will call *continuous* methods which include Galerkin [5], Numerov, and other finite-difference methods. Application of these methods to bound and scattering problems in physics can be found in [6, 7]. An extensive discussion of the mathematical and numerical properties of splines can be found in [2]. A simple discussion of splines and a comprehensive account of iterative methods for the solution of differential equations can be found in [4]. The methods discussed in this paper have been recently applied to the study of low-energy heavy-ion reactions [8] and to the relativistic Dirac equation [9, 10].

The paper is organized as follows: In Section 2 we describe the general properties of basis-splines and the generation of basis-splines and their derivatives. Section 3 deals with the BSCM and the implementation of boundary conditions. Section 3 also contains examples which illustrate typical accuracies obtained for the discrete representation of differential operators. In Section 4 we solve the Schrödinger equation for the well-known Morse potential using the BSCM. Section 5 develops methods for solving the Poisson and generalized Helmholtz equations. These calculations are performed in both radial and three-dimensional Cartesian coordinates. The paper concludes with the discussion of the results in Section 6.

2. Basis-Spline Functions

The general properties of basis-splines and their utilization for the purposes of interpolation and solution of ordinary differential equations is discussed extensively in the literature. A comprehensive account of this work can be found in Ref. [2]. In this section we will not go through the derivation of formulae concerning the generation of basis-splines and their derivatives but rather give a brief description of their properties and explain the formulae used in practical calculations. A reader already familiar with these details may wish to continue with the next section after the following paragraph.

Given a set of points or *knots* denoted by the set $\{x_i\}$ a basis-spline (B-spline denoted by B_i^M) function of order M is constructed from continuous piecewise polynomials of order $M-1$. B-splines have continuous derivatives up to $(M-2)$th derivative and a discontinuous $(M-1)$th derivative. With this terminology, a cubic

B-spline corresponds to $M=4$. In the following discussion we will only consider odd-order splines or even-order polynomials for reasons related to the choice of the collocation points. A typical B-spline of order M is shown in Fig. 1. As we see, the ith B-spline is nonzero only in the interval (x_i, x_{i+M}). This property is commonly referred to as limited support. The knots are the points where polynomials that make up the B-spline join. In the interval containing the tail region B-splines fall off very rapidly to zero.

All of the nonzero splines for a given value of $x_i \leqslant x < x_{i+1}$ can be obtained from a recursion relation [2]

$$B_i^{k+1}(x) = \frac{(x-x_i)}{(x_{k+i}-x_i)} B_i^k(x) + \frac{(x_{k+i+1}-x)}{(x_{k+i+1}-x_{i+1})} B_{i+1}^k(x), \tag{1}$$

where $k=1, ..., (M-1)$. The iteration starts with the initial condition

$$B_i^1(x) = \begin{cases} 1 & \text{for } x_i \leqslant x < x_{i+1} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

This recursion relation makes all of the details of constructing B-splines transparent to the user in addition to being numerically stable and valid for non-equidistant and multiply defined knots. Note that the iteration is not only with respect to B-spline index i but also with respect to the order k. One can use the following triangular iteration scheme for practical calculations [2]

$$\begin{array}{ccccc} & & & & B_{i-M+1}^M \\ & & & & \cdot \\ & & B_{i-2}^3 & & \cdot \\ & B_{i-1}^2 & & & \cdot \\ B_i^1 & & B_{i-1}^3 & & \cdot \\ & B_i^2 & & & \cdot \\ & & B_i^3 & & \cdot \\ & & & & \cdot \\ & & & & B_i^M \end{array} \tag{3}$$

So, for a point $x_i < x < x_{i+1}$ there are only M nonzero B-splines of order M, given in the last column of Eq. (3). In practice, the triangular relation is used by starting from the initial condition (2) and going through the columns of Eq. (3).

The evaluation of various derivatives of B-splines can be obtained from the above recursion relation. For $x_i \leqslant x < x_{i+1}$ all B-splines with non-vanishing pth derivative can be obtained from

$$\partial^p B_i^{k+1}(x) = \left(\frac{k}{k-p}\right)\left[\frac{(x-x_i)}{(x_{k+i}-x_i)} \partial^p B_i^k(x) + \frac{(x_{k+i+1}-x)}{(x_{k+i+1}-x_{i+1})} \partial^p B_{i+1}^k(x)\right], \tag{4}$$

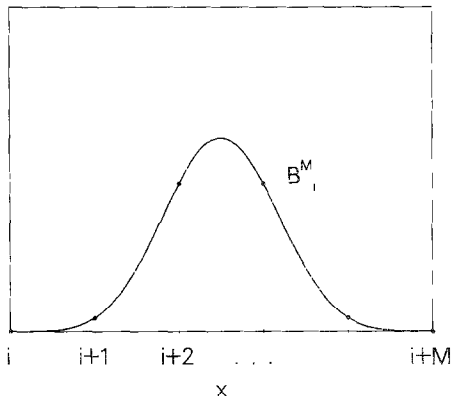

FIG. 1. A B-spline of order M and its extension over the knots x_i (actual curve drawn for $M=5$).

where $k=p+1, ..., (M-1)$. However, the initial condition for the recursion relation (4) is itself a recursive expression. Note that the recursion in Eq. (4) cannot be started for $k \leqslant p$ since $\partial^p B^k$ is zero. To find the lower order derivatives of lower order splines we use the recursion relation

$$C_i^{k+1}(x)=k\left\{\frac{C_i^k(x)}{(x_{k+i}-x_i)}-\frac{C_{i+1}^k(x)}{(x_{k+i+1}-x_{i+1})}\right\}, \tag{5}$$

where $k=1, ..., p$ and we have defined

$$C_i^{k+1}(x)\equiv \partial^k B_i^{k+1}(x). \tag{6}$$

The initial condition for (5) is

$$C_i^1(x)=\begin{cases}1 & \text{for } x_i \leqslant x < x_{i+1}\\ 0 & \text{otherwise.}\end{cases} \tag{7}$$

In practice, we first calculate and store the lower derivatives using Eq. (5) and subsequently solve for the pth-derivative from Eq. (4). Both of these equations have the same structure as Eq. (1) and therefore the same triangular relation is utilized in practical calculations.

Two other useful relations for B-splines are the summation,

$$\sum_i B_i^M(x)=1, \tag{8}$$

and the integration over all space,

$$\int_{-\infty}^{+\infty} B_i^M(x)\,dx \equiv h_i=\frac{x_{i+M}-x_i}{M}. \tag{9}$$

Equations (8) and (9) are generally correct except near the physical boundary, where the B-splines may be incomplete (see below). Equation (8) together with the fact that $B_i^M(x) > 0$ in the interval (x_i, x_{i+M}) makes B-splines a *partition of unity*. B-splines may also be regarded as localized wavepackets having Fourier transforms that are sharply peaked about zero momentum [6].

3. BSCM AND BOUNDARY CONDITIONS

In this section we will use the basis-spline collocation method to set up a lattice representation of functions and differential operators imposing various boundary conditions. The details of Coulomb boundary conditions will be discussed later.

3.1. *Enumeration of B-Splines*

Consider a region of space which has boundaries at $x_{\min}$ and $x_{\max}$. Figure 2 depicts such a region for which we have also drawn some of the B-splines around the boundaries. Since the B-splines have an extension over $M+1$ knots, B-splines beginning outside but at close proximity of the lower physical boundary will stretch into the physical region and B-splines that begin inside of the upper physical boundary will stretch beyond. In order to be able to generate all of the non-vanishing B-splines within the physical boundaries using Eq. (3) we need extra knots, in addition to the knots that are within the boundaries. This knot sequence can be defined as

$$\text{knot sequence} = (x_1, x_2, ..., x_M, x_{M+1}, ..., x_{M+N-1}, x_{M+N}, ..., x_{N+2M-1}), \quad (10)$$

where x_M and x_{M+N} correspond to the physical boundaries $x_{\min}$ and $x_{\max}$, respectively, and $x_i < x_{i+1}$ for $i = M, ..., M+N$. Including the boundary knots the total number of knots within the physical boundaries is $N+1$. The total number of knots

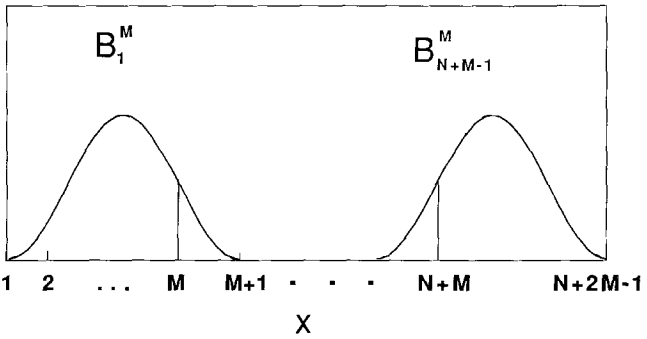

FIG. 2. A region of space with physical boundaries located at knots x_M and x_{M+N}. The B-spline B_1^M which begins at the first knot x_1 has its tail in the physical region. The last B-spline which begins within the physical boundaries is B_{N+M-1}^M. It extends up to the last knot x_{N+2M-1}.

in the entire sequence is $N+2M-1$. From Fig. 2 we also observe that with this knot sequence the splines that fall within the physical boundaries are naturally numbered as

$$B_1^M(x),\ B_2^M(x),\ ...,\ B_{N+M-1}^M. \tag{11}$$

Consequently, the total number of B-splines that fall within the physical boundaries is $N+M-1$.

3.2. *Collocation Representation of Functions*

A continuous function $f(x)$, defined in the interval $(x_{\min}, x_{\max})$, can be approximated in terms of the B-spline functions as (using the covariant notation)

$$f(x) = \sum_{i=1}^{N+M-1} B_i^M(x)\, c^i, \tag{12}$$

where quantities c^i denote the expansion coefficients. Equation (12) defines a spline approximation to the function $f(x)$ on an interval spanned by the subset of knots. In the collocation method, we demand this equation to be satisfied exactly at a set of points $\{x_\alpha\}$ named as *data* or *collocation* points. We than calculate the expansion coefficients c^i by the inversion of the matrix equation. These coefficients give exact values of the function at the collocation points by construction and provide a good interpolant at other values of x. The essence of our method is to transform a differential equation into a matrix equation with operators and functions expressed only in terms of these collocation points. The solution of the resulting matrix–vector equations will thus give the answer evaluated only on the collocation lattice. There are a number of practical ways to choose collocation points [2]; however, for odd-order B-splines an optimal and simple choice is to place one collocation point at the center of each knot interval within the physical boundaries

$$x_\alpha = \frac{x_{\alpha+M-1} + x_{\alpha+M}}{2}, \qquad \alpha = 1,\ ...,\ N. \tag{13}$$

Note that collocation points are denoted by greek subscripts. We can now write a linear system of equations by evaluating (12) at these collocation points

$$f_\alpha = \sum_{i=1}^{N+M-1} B_{\alpha i} c^i, \tag{14}$$

where $f_\alpha \equiv f(x_\alpha)$, and $B_{\alpha i} \equiv B_i^M(x_\alpha)$. Using matrix vector notation

$$(\mathbf{f})_N = (\mathbf{B})_{N\times(N+M-1)}\,(\mathbf{c})_{N+M-1}, \tag{15}$$

where we have also indicated the lengths of the vectors and matrices. In order to solve for the expansion coefficients the matrix $\mathbf{B}$ needs to be inverted. However, as it stands in (15) matrix $\mathbf{B}$ is not a square matrix. In order to achieve this inversion

we need to introduce additional linear equations which represent the boundary conditions imposed on $f(x)$ at the two boundary points, x_M and x_{M+N}. The essence of the lattice method is to eliminate the expansion coefficients c^i using this inverse matrix.

3.3. *Boundary Conditions*

Before we continue with what we will call *fixed boundary conditions* there is another type of boundary condition that can be imposed in a straightforward manner, the *periodic boundary conditions.* For periodic boundary conditions the function is assumed to be folding across one boundary into the other boundary. The terminology becomes more clear for dynamic problems where a function impacting one boundary-wall will appear to be entering back into the physical region from the opposite boundary-wall, thus the name periodic. In this case we form a finite closed space and the periodic boundary conditions can be used when boundary conditions are not important. In practice, periodic boundary conditions can be simply imposed by renumbering B-splines $B_{N+1}, ..., B_{N+M-1}$ by $B_1, ..., B_{M-1}$, respectively (without changing their location). Thus, B_1 seems like a B-spline that begins at knot x_{N+1} stretches up to the upper physical boundary and its tail folds to form the B_1 as drawn in Fig. 2. Consequently, the B-spline index i will stop at N instead of $N+M-1$ and the matrix **B** of Eq. (15) becomes an $N \times N$ matrix which can easily be inverted to solve for the expansion coefficients.

The *fixed boundary conditions* that we will consider can be cast into the form

$$\sum_{p \geqslant 0} K_{rp} \frac{d^p}{dx^p} f(x)|_{\text{boundary}} = 0, \tag{16}$$

where r counts the number of such boundary conditions and the coefficient array K is used to select a subset of terms with constant coefficients in the above summation. We note that p can be at most $M-1$, since the function $f(x)$ is represented in terms of Mth-order B-splines. Using (12), Eq. (16) becomes

$$\sum_{i=1}^{N+M-1} \left(\sum_{p \geqslant 0} K_{rp} \left[\frac{d^p}{dx^p} B_i^M(x) \right]_{\text{boundary}} \right) c^i = 0. \tag{17}$$

Finally, in terms of the matrix β_{ri}, which is defined to be

$$\beta_{ri} \equiv \sum_{p \geqslant 0} K_{rp} \left[\frac{d^p}{dx^p} B_i^M(x) \right]_{\text{boundary}}, \tag{18}$$

the boundary conditions represented by Eq. (16) can be put into the form

$$\sum_{i=1}^{N+M-1} \beta_{ri} c^i = 0. \tag{19}$$

Equation (19) is in the same form as Eq. (14) and can be joined to form a larger

linear system. This new linear system will not only impose the desired boundary conditions but for $r = 1, ..., (M-1)$ will also give a square matrix for inversion. Alternatively, one may wish to define few additional collocation points and only the desired number of boundary conditions. Specifically,

$$\tilde{\mathbf{B}} \equiv \begin{pmatrix} \mathbf{B} \\ \boldsymbol{\beta} \end{pmatrix}_{(N+M-1)\times(N+M-1)}. \tag{20}$$

If we augment the collocation vector on the left-hand side of Eq. (15) by $M-1$ zeroes, we can solve for the expansion coefficients as

$$(\mathbf{c})_{N+M-1} = (\tilde{\mathbf{B}}^{-1})_{(N+M-1)\times(N+M-1)} \begin{pmatrix} \mathbf{f} \\ \hline \mathbf{0} \end{pmatrix}_{N+M-1}. \tag{21}$$

The above choice of collocation points ensures that matrix $\mathbf{B}$ is non-singular and can be inverted [2]. However, due to the presence of the zeroes in the column vector only a part of the inverse is required; we denote this section of the inverse matrix by

$$(\tilde{\mathbf{C}})_{(N+M-1)\times N}. \tag{22}$$

Thus, the inverse matrix $\tilde{\mathbf{C}}$ acts like a metric describing the transformation between the collocation points and knots. As we will see below, this inverse will be used to generate collocation lattice operators which automatically satisfy the above boundary conditions. We also point out that $\tilde{\mathbf{C}}$ satisfies the property

$$\sum_i B_{xi} \tilde{C}^{i\beta} = \delta_x^\beta. \tag{23}$$

By inserting the expansion coefficients represented by Eq. (21) into (14) and using the property (23) one can trivially show that all local functions will have a local representation in the finite-dimensional collocation space

$$f(x) \to f_x. \tag{24}$$

3.4. *Collocation Representation of Operators*

Consider the action of an operator $\mathcal{O}$ onto a function $f(x)$

$$\mathcal{O}f(x) = \sum_i [\mathcal{O}B_i^M(x)]\, c^i. \tag{25}$$

If we evaluate the above expression at the collocation points x_α we can write

$$[\mathcal{O}f]_\alpha = \sum_i [\mathcal{O}B]_{xi}\, c^i. \tag{26}$$

Substituting from Eq. (21) for the coefficients c^i, we obtain

$$[\mathcal{O}f]_\alpha = \sum_{i\beta} [\mathcal{O}B]_{\alpha i} \tilde{C}^{i\beta} f_\beta$$
$$= \sum_\beta O_\alpha^\beta f_\beta, \tag{27}$$

where we have defined the collocation space matrix representation of the operator $\mathcal{O}$ by

$$O_\alpha^\beta = \sum_i [\mathcal{O}B]_{\alpha i} [\tilde{C}]^{i\beta}. \tag{28}$$

Note that the construction of the collocation space operators can be performed once and for all at the beginning of a calculation, using only the given knot sequence and collocation points. Due to the presence of the inverse in Eq. (28) the operator $\mathcal{O}$ is not a sparse matrix. In practice, operator $\mathcal{O}$ is chosen to be a differential operator such as d/dx or d^2/dx^2. In this case the derivatives of B-splines are calculated using the recursion relation (4) and are used in Eq. (28).

3.5. *Collocation Weights*

The appropriate quadrature weights which incorporate the boundary conditions can be obtained by integrating both sides of Eq. (12),

$$I = \int_a^b f(x)\, dx = \sum_i \left[\int_a^b B_i^M(x) \right] c^i. \tag{29}$$

In practice, integration limits a and b usually coincide with the boundaries of the physical region x_M and x_{N+M}, respectively. In this case, for most B-splines the integration in Eq. (29) covers the entire extension of the B-spline for which we can use Eq. (9) to write

$$I = \sum_i h_i c^i. \tag{30}$$

However, for B-splines that extend beyond the physical boundary, the integration limits will not cover the full extent of the B-spline and the integral in Eq. (29) would only give the part of the area that falls within the physical boundaries (for example, the area under B_1^M of Fig. 2). Of course, for periodic boundary conditions all B-splines are present in their full extent (due to folding) and Eq. (9) can be used for all i with any knot sequence. One way to circumvent an explicit integration of the area under the boundary B-splines is to choose multiple knots at both boundaries for fixed boundary conditions. In this case the knot sequence is given by

$$\text{knot sequence} = (x_M, ..., x_M, x_{M+1}, x_{M+2}, ..., x_{N+M-1}, x_{N+M}, ..., x_{N+M}), \tag{31}$$

with the total number of knots remaining the same. This knot sequence produces the correct area under the boundary B-splines and Eq. (9) can be used for all i.

To find the collocation weights we use Eq. (21) to substitute for the expansion coefficients c^i in Eq. (30),

$$I = \sum_{i=1}^{N+M-1} \sum_{\alpha=1}^{N} h_i \tilde{C}^{i\alpha} f_\alpha. \tag{32}$$

This allows us to define the collocation integration weights w^α together with the quadrature formula,

$$\begin{aligned} \int f(x)\,dx &= \sum_\alpha w^\alpha f_\alpha \\ w^\alpha &= \sum_i h_i \tilde{C}^{i\alpha}. \end{aligned} \tag{33}$$

3.6. *Numerical Illustrations*

In this section we will give numerical examples of the formalism discussed above. We will mainly concentrate on problems with fixed boundary conditions. In particular, we will study the action of the first derivative operator on two functions

$$\begin{aligned} f(x) &= \sin(x), \qquad \text{for} \quad 0 \leqslant x < \frac{15\pi}{2} \\ f(x) &= \sin(x)\, e^{-x}, \qquad \text{for} \quad 0 \leqslant x < 20. \end{aligned} \tag{34}$$

Both of these functions vanish at $x = 0$ and have an oscillatory behavior over the physical region. The second function also vanishes at $x = \infty$. On the other hand, the first function has a vanishing first derivative at $x = 15\pi/2$ which has to be incorporated into the boundary conditions. In all of our examples we have tried to use a *reasonable* B-spline order and number of collocation points. It is always possible to make relative errors as small as needed by increasing these parameters.

Knots are chosen at equally spaced intervals within the physical boundaries. At each boundary we add $(M-1)$ multiple knots. Collocation points are located at the middle of each knot interval within the physical region. Table I shows the

TABLE I

Knot and Collocation Sequence for the Case of Order $M = 3$ and the Number of Collocation Points $N = 10$. Equally Distributed in the Interval (0, 10)

Knots	0	0	0	1	2	3	4	5	6	7	8	9	10	10	10
Colloc.			0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5			

Note. There are $N + 2M - 1 = 15$ knots. The knot sequence contains $M - 1$ additional multiple knots at each boundary as discussed in text.

values of knots and collocation points for the simple case of $M=3$ and $N=10$ in the physical interval (0, 10). Various boundary conditions can be implemented via the coefficient array K_{rp} of Eq. (16). For order M we impose a total of $M-1$ boundary conditions in order to form a square matrix as discussed above. Consequently, the index $r=1, ..., (M-1)$, and for each r the index p can go up to $M-1$. We divide the boundary conditions equally between left and right boundaries (i.e., $(M-1)/2$ at each boundary, M being an odd number). For example, to implement a zero boundary condition for a function at a particular boundary point we set $K_{10}=1$ and all others to zero. This way of implementing boundary conditions may introduce redundant conditions for $M>3$. For example, for the $M=5$ we need to introduce four boundary conditions, two at each end. The first two of these conditions may correspond to the real physical conditions and the remaining two may be redundant. In practice, we have found that requiring some high-order derivatives to be zero does not alter the results. For $M=5$ case we set the fourth-order derivatives to zero at each boundary. Alternatively, one may wish to define extra collocation points near the physical boundaries and only one boundary condition at each end.

In Table II we show the error in the difference of the numerical first differentiation constructed on the collocation lattice and the analytic derivative evaluated at the same collocation points for the function $\sin(x)$ in the interval $(0, 15\pi/2)$. The error is defined by

$$\text{error} = \frac{1}{N} \sum_{\alpha=1}^{N} \left| \frac{g_\alpha - \cos(x_\alpha)}{\cos(x_\alpha)} \right|, \tag{35}$$

where

$$g_\alpha = \sum_\beta D_\alpha^\beta f_\beta. \tag{36}$$

TABLE II

Error (see Eq. (35)) in the First Derivative of the $\sin(x)$ Function Computed on the Collocation Lattice as a Function of B-spline Order M and the Number of Collocation Points N

N	$M=3$	$M=5$	$M=7$	$M=9$
10	4.76(−1)	1.93(−1)	9.42(−2)	8.39(−2)
20	7.26(−2)	3.86(−3)	8.63(−4)	7.56(−4)
30	2.85(−2)	5.84(−4)	7.25(−5)	9.02(−5)
50	9.60(−3)	6.53(−5)	6.49(−6)	5.69(−6)
80	3.67(−3)	9.46(−6)	2.22(−8)	7.29(−9)
κ	2.3	4.7	6.9	7.5

Note. The physical region is in the interval $(0, 15\pi/2)$. The last line shows an approximate power law dependence, $N^{-\kappa}$, of the error.

The first derivative matrix **D** is given by (see Eq. (28))

$$D_{\alpha}^{\beta} = \sum_{i} B'_{\alpha i} \tilde{C}^{i\beta}, \tag{37}$$

where B' indicates the first derivative of the B-spline functions. We have used the above error formula for functions that do not have an exponential behavior. We observe the typical improvement as we increase the B-spline order and the number of collocation points. For a few points the improvement with respect to the order seems to saturate at $M = 7$. This saturation will in general depend on the behavior of the functions and on the number of collocation points. We would also like to point out that in the calculations we have imposed the boundary conditions that

change one of these conditions to a wrong boundary condition, our error for the best case in the table becomes 0.20, with most of the error originating from points close to the boundary with the wrong condition. The last line of Table II shows a power law dependence of the error. For each order we have fitted the error using a two-parameter expression, $AN^{-\kappa}$. We observe that the values of κ increase rapidly with the increasing B-spline order.

In Fig. 3 we have plotted the difference $|g^{\text{numerical}}(x) - g^{\text{exact}}(x)|$, where g denotes the first derivative of the function $\sin(x)\exp(-x)$, as a function of x. Calculations employed B-splines with order $M = 7$ and 100 collocation points. Since we are dealing with an exponentially damped function, we have used non-equidistant knots determined by the transformation

$$x = a(e^{b\zeta} - 1), \tag{38}$$

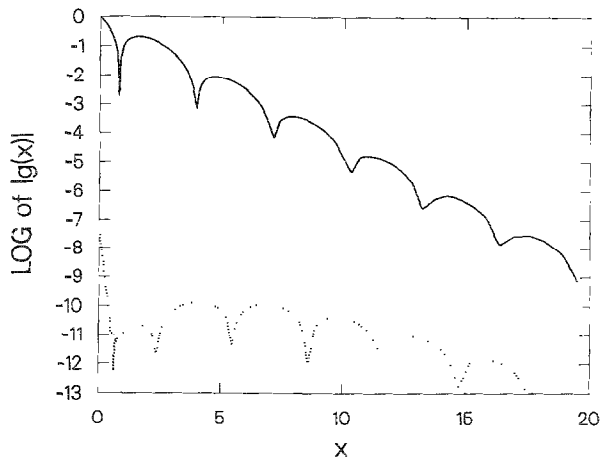

FIG. 3. Logarithm of $|g^{\text{numerical}} - g^{\text{exact}}|$, where g denotes the first derivative of the function $\sin(x)\exp(-x)$, as a function of x (dotted curve). The solid curve is the logarithm of $|g(x)|$. Calculations employed B-splines with order $M = 7$ and $N = 100$ collocation points in the physical interval (0, 20). The non-equidistant grid used in the calculations is given by Eq. (38).

where ζ denotes an equidistant grid in the interval (0, 1). This transformation leads to a knot distribution with a high density of points in the vicinity of the origin and widely spaced points in the tail region. In practical calculations we have used $a = 1.0$ and $b = 3.3$. On the same figure we have also plotted the absolute value of the exact derivative to demonstrate the relative accuracy of the result. We have imposed zero boundary conditions on both ends of the physical interval. The agreement in the tail region is very satisfactory considering the fact that we are dealing with an exponentially damped oscillatory function on a relatively coarse grid. The accuracies obtained for second-order derivatives are commensurate with these results.

4. Radial Schrödinger Equation

In this section we will calculate the bound states of the radial Schrödinger equation ($m = \hbar = 1$, $l = 0$),

$$\left[-\frac{1}{2}\frac{d^2}{dr^2} + V(r) \right] \psi_n(r) = E_n \psi_n(r), \tag{39}$$

for the Morse potential

$$V(r) = 25[e^{-4(r-3)} - 2e^{-2(r-3)}]. \tag{40}$$

In Eq. (39) the wavefunctions ψ_n satisfy the boundary conditions

$$\begin{aligned} &\psi_n(r) \to 0 \qquad \text{for} \quad r \to 0 \\ &\psi_n(r) \to 0 \qquad \text{for} \quad r > r_{\max}. \end{aligned} \tag{41}$$

The second condition is an approximation to the real asymptotic behavior $\psi_n(r) \to Ce^{-\kappa r}$. The reasons for choosing this form for the potential is twofold. First, the four bound state eigenvalues are known analytically

$$E_n = -[5 - \sqrt{2}\,(n + \tfrac{1}{2})]^2, \qquad n = 0, 1, 2, 3; \tag{42}$$

second, there exists an extensive study of the same potential using the *continuous* methods in the literature [6].

Collocation lattice representation of Eq. (39) is obtained by using Eqs. (24) and (27). The result is

$$\sum_{\beta} H_{\alpha}^{\beta} \psi_{n\beta} = E_n \psi_{n\alpha}, \tag{43}$$

where

$$\begin{aligned} H_{\alpha}^{\beta} &\equiv -\tfrac{1}{2} D_{\alpha}^{\beta} + V_{\beta} \delta_{\alpha}^{\beta} \\ D_{\alpha}^{\beta} &\equiv \sum_{i} B''_{\alpha i} \tilde{C}^{i\beta}. \end{aligned} \tag{44}$$

The diagonalization of the matrix **H** will give N eigenvalues, of which the four lowest correspond to the bound state energies E_n, and N eigenvectors, each evaluated on N collocation points. Alternatively, one can use an iterative scheme to find only the lowest four eigenvalues. An efficient damped relaxation method for the Schrödinger equation as well as the Dirac equation can be found in Ref. [11].

In practice, we have chosen an equidistant collocation grid as discussed in the previous section in the interval (0, 12). This grid agrees with the one chosen in Ref. [6]. In Table III we tabulate the quantity $|E_n - E_n^{\text{exact}}|/|E_n^{\text{exact}}|$. The calculations were carried out for $N = 80$ collocation points and various B-spline orders M. We observe that with increasing M we obtain a substantial improvement in the lowest three eigenvalues. The last eigenvalue is very loosely bound ($E_3 = -0.002525$) and dies off at a much larger value of $r_{\max}$. Increasing the value of $r_{\max}$ should lead to a better representation of this state and improve its energy. Another important property is that the Morse potential reaches a very large value at the origin ($V(0) = 4.05 \times 10^6$). In principle, we can move the value of $r_{\min}$ from 0 to 1.0, where the potential is still quite large and wavefunctions satisfy the zero boundary condition (at one part in 10^{-10} level). This should also lead to an improvement in the eigenvalues since more of the collocation points will be located in the region where wavefunctions are appreciably different from zero. In order to test these conjectures we have investigated a non-equidistant grid that was generated by first mapping the zeroes of Legendre polynomials from the interval $(-1, +1)$ to the interval $(y_{\min}, y_{\max})$, with $y_{\min} = 2 \arctan(r_{\min})/\pi$ and $y_{\max} = 2 \arctan(r_{\max})/\pi$. These points were then used in the transformation $r_i = \tan(\pi y_i/2)$. This transformation results in a large density of points near r_{min} and widely spaced points at far distances. The last line of Table III shows the results for $M = 7$ and 64 transformed Gauss points in the physical interval (1, 24). We note a substantial

TABLE III

Error, $|E_n - E_n^{\text{exact}}|/|E_n^{\text{exact}}|$, for the Eigenvalues of the Morse Potential

M \ n	0	1	2	3
3	2.9(−3)	1.7(−2)	6.1(−2)	9.9
5	2.3(−4)	1.6(−3)	6.5(−3)	5.2
7	1.7(−5)	1.2(−4)	5.2(−4)	4.7
9	1.6(−6)	1.1(−5)	5.1(−5)	4.7
7[a]	5.4(−8)	1.6(−5)	1.3(−4)	3.9(−1)

Note. The calculations are performed with 80 collocation points and for various B-splines order M. The physical region is in the interval (0, 12).

[a] Calculated by using 64 points with the non-equidistant transformation described in text. The physical interval is (1, 24).

improvement in the results (in comparison to the $M=7$ case), considering also the fact that the physical interval has almost doubled and the number of collocation points are reduced to 64. In comparison to the $M=9$ case, the states with $n=0$ and 3 show an improvement but not the states with $n=1$ and 2. This is due to the fact that the number of collocation points have been reduced from 80 to 64 in case 7*a* and the points are distributed in a non-equidistant fashion, with most points concentrating in a region where the $n=0$ state extends. The states $n=1$ and 2 are spread over a larger region and the total number of points in that region is less than the 80-point case. The $n=3$ case can only be improved by extending the value of $r_{\max}$ to ensure that the wavefunction becomes very small at that value. So the improvement is mainly due to the large value of $r_{\max}$ used in the case 7*a*.

Our results are in general agreement with those of Ref. [6] where cubic B-splines were utilized. Reference [6] also contains calculations using second-order finite-difference and Numerov algorithms which clearly demonstrate the improvement of the results due to the use of B-spline approximations.

5. Poisson and Generalized Helmholtz Equations

In this section we will provide a method for implementing Coulomb boundary conditions using the BSCM method. The methodology discussed here can be generalized to any differential equation with arbitrary boundary conditions which may or may not be equal to zero.

We are interested in the solution of the two well-known differential equations, namely the Poisson,

$$\nabla^2 \Phi_C(\mathbf{r}) = -4\pi\rho(\mathbf{r}), \tag{45}$$

and the generalized Helmholtz,

$$(\nabla^2 - a^2)\, \Phi_Y(\mathbf{r}) = -4\pi\rho(\mathbf{r}). \tag{46}$$

The formal solution for the Poisson and Helmholtz equations can be written in terms of the corresponding Green's functions as

$$\begin{aligned} \Phi_C(\mathbf{r}) &= \int d^3r' \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} && \text{Poisson} \\ \Phi_Y(\mathbf{r}) &= \int d^3r' \rho(\mathbf{r}') \frac{e^{-a|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r}-\mathbf{r}'|} && \text{Helmholtz.} \end{aligned} \tag{47}$$

However, the three-dimensional integrations are quite difficult and in practice one usually solves the corresponding differential equations. Equations (45) and (46) have different boundary conditions, which is also apparent from the asymptotic behavior of the solutions given in Eqs. (47). The Poisson equation has to be solved

by specifying the values of Φ_C at the boundaries since the solution is finite there. On the other hand, for reasonably large distances, Φ_Y can be considered zero and we simply have to implement a zero boundary condition as discussed in the previous sections. Since the two equations are similar, except for the boundary conditions, below we will only deal with the Poisson equation. The calculations for the Helmholtz equation will also be discussed at the end of each subsection.

5.1. *Radial Equations*

For a spherically symmetric density ρ and potential Φ_C the Poisson equation can be reduced to the form

$$\frac{d^2\phi(r)}{dr^2} = -4\pi r\rho(r), \qquad 0 \leqslant r \leqslant r_{\max}$$
$$\Phi_C(r) = \frac{\phi(r)}{r}. \tag{48}$$

The solution $\phi(r)$ is zero at $r=0$ by virtue of Eq. (48), and it has a finite value at $r_{\max}$. $\phi(r)$ can be expanded in terms of B-spline functions as before.

$$\phi(r) = \sum_{i=1}^{N+M-1} B_i^M(r)\, c^i. \tag{49}$$

Evaluating Eq. (49) at the collocation points r_α gives us the incomplete linear system of equations

$$\phi_\alpha = \sum_{i=1}^{N+M-1} B_{\alpha i} c^i. \tag{50}$$

This system of equations has to be supplemented by linear equations describing the boundary conditions. These equations can be cast into the form

$$\phi_b = \sum_{i=1}^{N+M-1} B_{bi} c^i, \tag{51}$$

where ϕ_b denotes the *known* values of the solution at the *boundary points*, and B_{bi} are the B-splines evaluated at these boundary points. The solutions ϕ_b are most commonly obtained by making a multipole expansion of the formal solution (47) at distant boundary points. The details of this expansion are given in the next section. If we choose $M-1$ such boundary points, Eqs. (51) can be appended to Eqs. (50) to form a square system which can be inverted to solve for the expansion coefficients c^i. The new system can be written as

$$\phi_m = \sum_{i=1}^{N+M-1} \tilde{B}_{mi} c^i, \tag{52}$$

where

$$\tilde{B}_{mi} = \begin{pmatrix} B_{\alpha i} \\ -- \\ B_{bi} \end{pmatrix}, \qquad \phi_m = \begin{pmatrix} \phi_\alpha \\ -- \\ \phi_b \end{pmatrix}. \tag{53}$$

The inversion of Eq. (52) gives

$$c^i = \sum_{m=1}^{N+M-1} (\tilde{B}^{-1})^{im} \phi_m. \tag{54}$$

Expansion coefficients c^i can be substituted into the Poisson equation written in terms of the B-spline expansion on the collocation lattice

$$\sum_i B''_{\alpha i} c^i = -4\pi r_\alpha \rho_\alpha \tag{55}$$

to give

$$\begin{gathered} \sum_m D^m_\alpha \phi_m = -4\pi r_\alpha \rho_\alpha \\ D^{mi}_\alpha \equiv \sum_i B''_{\alpha i} (\tilde{B}^{-1})^{im}. \end{gathered} \tag{56}$$

Here $\rho_\alpha \equiv \rho(r_\alpha)$ and the sum over m runs over the full range, $m = 1, ..., N+M-1$. However, in Eq. (56) the values of ϕ_m at the boundary points are known quantities, thus the summation over m can be divided into two parts

$$\sum_{\beta-1}^{N} D^\beta_\alpha \phi_\beta = -4\pi r_\alpha \rho_\alpha - \sum_{b-N+1}^{N+M-1} D^b_\alpha \phi_b. \tag{57}$$

The right-hand side of Eq. (57) are all known quantities evaluated at the collocation and boundary points and the second term acts like an image charge to fix the appropriate boundary conditions.

In practice, we have used N equidistant collocation points in the interval $(0, r_{\max})$. In addition, we have selected $M-1$ boundary points, half between $r=0$ and the first collocation point and the other half between the last collocation point and $r_{\max}$. These boundary points included $r=0$ and $r=r_{\max}$. The solution ϕ is zero at $r=0$ and was explicitly calculated at the other boundary points using either a multipole expansion or known analytic values as discussed in the example below. The $N \times N$ matrix $\mathbf{D}$ was inverted to solve for ϕ_β in Eq. (57). In Table IV we tabulate the error as a function of the number of collocation points N using B-splines of order five and density (normalized to unity)

$$\rho(r) = \frac{1}{8\pi} e^{-r}. \tag{58}$$

TABLE IV

Total Error (see Eq. (35)) in the Solution of the Poisson Equation in Radial Coordinates as a Function of the Number of Collocation Points N

N	10	20	30	40	50	60
	4.8(−4)	3.3(−5)	6.4(−6)	1.5(−6)	7.6(−7)	2.6(−7)

Note. We have used a fixed B-spline order $M=5$. Physical region is in the interval (0, 6).

In this case the analytic solution is given by

$$\phi(r)=1-\tfrac{1}{2}(2+r)\,e^{-r}. \tag{59}$$

The reason why order five is adequate is due to the behavior of the solutions of the Poisson equation. In general, the solutions are smooth and fall off slowly at large $r(\Phi_C(r)\to 1/r,\ \phi(r)\to 1)$. Increasing the order may still improve the results. Also, at the boundary the exact value of the function calculated by an asymptotic expansion is used. Due to this and the relative constancy of the solutions over the region of r values considered, equally spaced points are the best choice.

The solution of the Helmholtz equation only requires the implementation of a zero boundary condition at $r=0$ and $r=r_{\max}$. In this case,

$$\left(\sum_{\beta=1}^{N} D_\alpha^\beta - a^2\delta_\alpha^\beta\right)\phi_\beta = -4\pi r_\alpha \rho_\alpha, \tag{60}$$

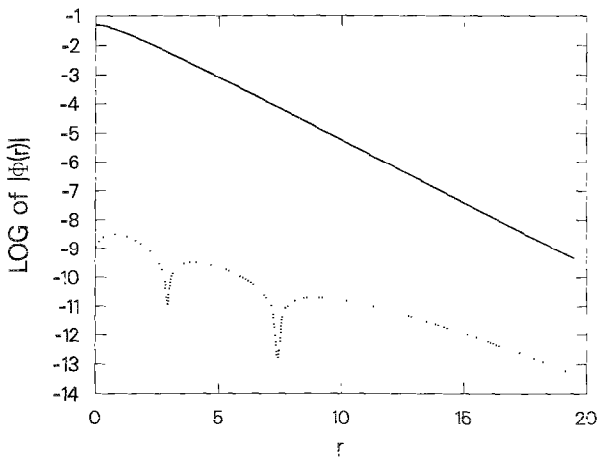

FIG. 4. Logarithm of $|\Phi^{\text{numerical}}-\Phi^{\text{exact}}|$, for the Helmholtz equation as a function of the radial coordinate r (dotted curve). The solid curve is the logarithm of $\Phi(r)$ Calculations employed B-splines with order $M=5$ and $N=100$ collocation points in the physical interval (0, 20). The non-equidistant grid used in the calculations is given by Eq. (38).

where the second derivative matrix **D** is calculated using the fixed boundary conditions discussed in the previous sections. In Fig. 4 we show the error in the solution of Eq. (60) for the same spherical density used for the Poisson equation. The analytic solution for the Helmholtz equation is given by

$$\phi(r) = \frac{1}{(1-a^2)^2}\{e^{-ar} - e^{-r}[1 + \tfrac{1}{2}(1-a^2)r]\}. \tag{61}$$

We have used B-splines of order five and 100 collocation points distributed according to the transformation given by Eq. (38). The solid curve is the logarithm of the analytic solution, whereas the dotted curve denotes the logarithm of the difference $|\Phi^{\text{numerical}} - \Phi^{\text{exact}}|$. We observe for both Poisson and Helmholtz equations that very reliable solutions are obtained with relatively low order B-splines and few collocation points.

5.2. *Three-Dimensional Equations*

The methods described in the context of the radial Poisson equation can be easily generalized to three dimensions. In Cartesian coordinates the Poisson equation is given by

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} + \frac{\partial^2 \Phi}{\partial z^2} = -4\pi\rho(x, y, z). \tag{62}$$

The potential $\Phi(x, y, z)$ can be expanded in terms of the B-spline basis as

$$\Phi(x, y, z) = \sum_{ijk} B_i(x)\, B_j(y)\, B_k(z)\, c^{ijk}, \tag{63}$$

where we have assumed that all B-splines are of order M (they can have different orders) and the indices i, j, k correspond to B-splines in x, y, z directions, respectively. Inserting Eq. (63) into (62) and evaluating the expression at a set of collocation points denoted by subscripts α, β, γ, we obtain

$$\sum_{ijk} [B''_{\alpha i} B_{\beta j} B_{\gamma k} + B_{\alpha i} B''_{\beta j} B_{\gamma k} + B_{\alpha i} B_{\beta j} B''_{\gamma k}]\, c^{ijk} = -4\pi\rho_{\alpha\beta\gamma}. \tag{64}$$

Here, like i, j, k, the indices α, β, γ correspond to the collocation points in x, y, z, directions, respectively. As it stands, we cannot solve for the expansion coefficients c^{ijk} since the three B-matrices are not square matrices. The generalization for introducing boundary potentials can be achieved if we consider the following expansion for the potential

$$\Phi_{mnp} = \sum_{ijk} B_{mi} B_{nj} B_{pk}\, c^{ijk}. \tag{65}$$

In this expression the indices m, n, p not only include the collocation points but

also a set of extra boundary points. Specifically, in each dimension we choose an additional set of boundary points in exactly the same way for the radial equation. However, in the three-dimensional case these extra boundary points lead to the specification of the boundary potentials on the surfaces of the cube. This is apparent from Eq. (65) if one considers one of the points, say m, to be a boundary point. In this case, the other two indices n, p correspond to all possible collocation points in y, z directions. This describes a surface for which the exact potential values need to be calculated. If n, p correspond to boundary points, we represent the corners and lines joining the corners of the cube. Finally, for $M > 3$ we choose more than one point at each side of the Cartesian axis: in this case we have to specify boundary potentials at adjacent surfaces of the cube.

In Eq. (65) all of the B-matrices are square and they can be inverted to solve for the expansion coefficients c^{ijk},

$$c^{ijk} = \sum_{mnp} (B^{-1})^{im} (B^{-1})^{jn} (B^{-1})^{kp} \Phi_{mnp}. \tag{66}$$

Substitution of Eq. (66) into the Poisson equation (63) and making use of the relation

$$\sum_{i} B_{\alpha i} (B^{-1})^{im} = \delta_{\alpha}^{m} \tag{67}$$

yields the collocation lattice representation of the Poisson equation

$$\begin{aligned} &\sum_{\alpha'} D_{\alpha}^{\alpha'} \Phi_{\alpha'\beta\gamma} + \sum_{\beta'} D_{\beta}^{\beta'} \Phi_{\alpha\beta'\gamma} + \sum_{\gamma'} D_{\gamma}^{\gamma'} \Phi_{\alpha\beta\gamma'} \\ &\quad = -4\pi\rho_{\alpha\beta\gamma} - \sum_{a} D_{\alpha}^{a} \Phi_{a\beta\gamma} - \sum_{b} D_{\beta}^{b} \Phi_{\alpha b\gamma} - \sum_{c} D_{\gamma}^{c} \Phi_{\alpha\beta c}, \end{aligned} \tag{68}$$

where the points a, b, c denote the boundary points in x, y, z directions, respectively. The definition of the second derivative matrix $\mathbf{D}$ is given by Eq. (56). The boundary potentials needed for the right hand side of Eq. (68) are calculated by using an asymptotic expansion of the denominator in Eq. (47),

$$\begin{aligned} \frac{1}{|\mathbf{r} - \mathbf{r}'|} &= e^{-\mathbf{r}' \cdot \nabla} \left(\frac{1}{|\mathbf{r}|} \right) \\ &= \left[1 - \mathbf{r}' \cdot \nabla + \frac{1}{2} (\mathbf{r}' \cdot \nabla)^2 - \frac{1}{6} (\mathbf{r}' \cdot \nabla)^3 + \frac{1}{24} (\mathbf{r}' \cdot \nabla)^4 - \cdots \right] \frac{1}{r}; \end{aligned} \tag{69}$$

here $r \gg r'$, i.e., no density on the boundary. We have kept the expansion up to the fourth order including all of the odd moments.

Equation (68) is in the form $\mathbf{A} \cdot \mathbf{\Phi} = \mathbf{b}$. However, in this case the dimensions of

matrix **A** can be very large and iterative algorithms become more suitable for the solution of the linear system. We have used the conjugate gradient algorithm

$$\mathbf{r}_0 = \mathbf{b} - \mathbf{A} \cdot \mathbf{\Phi}_0$$

$$\mathbf{p}_0 = \mathbf{r}_0$$

do for $i = 0, 1, 2, \ldots$ until convergence

$$\alpha_i = (\mathbf{r}_i, \mathbf{r}_i)/(\mathbf{p}_i, \mathbf{A}\mathbf{p}_i)$$

$$\mathbf{\Phi}_{i+1} = \mathbf{\Phi}_i + \alpha_i \mathbf{p}_i$$

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{A}\mathbf{p}_i$$

$$\beta_{i+1} = (\mathbf{r}_{i+1}, \mathbf{r}_{i+1})/(\mathbf{r}_i, \mathbf{r}_i)$$

$$\mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1}\mathbf{p}_i$$

end do.

The residual was required to satisfy the condition $\|\mathbf{r}_i\| < 10^{-10}$ for convergence. As the initial guess for Φ we have used the potential of a uniform charge distribution.

Table V illustrates the three-dimensional results for the Poisson equation. In calculations we have used B-splines of order $M = 5$ and a different number of equidistant collocation points distributed in the physical interval $(-12, +12)$ in each Cartesian dimension. We compare the total electrostatic energy

$$E_s = \int dx\, dy\, dz \rho(x, y, z)\, \Phi(x, y, z) \to \sum_{\alpha\beta\gamma} w^\alpha w^\beta w^\gamma \rho_{\alpha\beta\gamma} \Phi_{\alpha\beta\gamma}, \tag{70}$$

with the analytic value of 5/16. This quantity also incorporates the errors arising from the three-dimensional volume integral. The number of iterations required for

We have also used the same algorithm for the solution of the Helmholtz equation with accuracies that are somewhat better than the Poisson equation. In this

TABLE V

The Error, $|E_s - 0.3125|/0.3125$, in the Total Electrostatic Energy (70) for the Solution of the Poisson Equation in Three-Dimensional Cartesian Coordinates.

N	$(10)^3$	$(20)^3$	$(30)^3$	$(40)^3$	$(50)^3$
	4.2(−2)	4.5(−3)	8.3(−4)	1.6(−4)	3.1(−5)

Note. The results are given as a function of the number of collocation points for a fixed B-spline order $M = 5$. The physical region is in the interval $(-12, +12)$ in each dimension.

case the conjugate gradient iteration takes only 15 iterations for convergence. This is due to the intrinsic difference in the behavior of the solutions for the Poisson and Helmholtz equations (long-range vs finite-range). However, in practical calculations such as the nuclear and atomic Hartree–Fock studies, where the self-consistent equations are solved iteratively, one can use the previous solution as an initial guess for the next step. In this case the number of subsequent iterations rapidly decreases.

6. Summary and Discussion

We have developed a methodology for the lattice solution of the boundary value differential equations using the basis-spline collocation method. A large class of problems in physics which currently employ low order finite-difference techniques can benefit by exploiting higher order interpolation methods such as the BSCM. We have given a number of examples of the method which demonstrate the implementation of various types of boundary conditions. In most applications the lattice equations can be written in terms of matrix–vector operations. In BSCM the matrix representation of operators are not sparse and require a larger computational effort. On the other hand, *continuous* methods such as the Galerkin method using B-splines, result in a system of generalized linear equations with banded matrices. The stability of these equations and their potential use for the study of complex dynamical systems using higher order B-splines remains for investigation. For the calculations performed in this manuscript we have developed a library of fortran subroutines called BSLIB. This library includes all of the basic B-spline and derivative generation routines for fixed or periodic boundary conditions and is available from the authors.

Acknowledgments

This research was sponsored in part by the U.S. Department of Energy under Contract DE-AC05-84OR21400 with Martin Marietta Energy Systems, Inc. and under Contract DE-FG05-87ER40376 with Vanderbilt University. The numerical calculations were carried out on CRAY-2 supercomputers at the National Center for Supercomputing Applications, Illinois, and at the National Magnetic Fusion Energy Computer Center, Livermore, CA.

References

1. C. Bottcher and M. R. Strayer, *Ann. Phys (N.Y.)* **175**, 64 (1987).
2. C. De Boor, *Practical Guide to Splines* (Springer-Verlag, New York, 1978); and references therein.
3. A. S. Umar, M. R. Strayer, P.-G. Reinhard, K. T. R. Davies, and S.-J. Lee, *Phys. Rev. C* **40**, 706 (1989); and references therein.
4. G. I. Marchuk, *Methods of Numerical Mathematics* (Springer-Verlag, New York, 1975).
5. C. A. J. Fletcher, *Computational Galerkin Methods* (Springer-Verlag, New York, 1984).
6. B. W. Shore, *J. Chem. Phys.* **58**, 3855 (1973).

7. C. F. FISCHER AND M. IDREES, *Comput. Phys.* **3**, 53 (1989).
8. A. S. UMAR, in *Proceedings of the Summer School of Computational Atomic and Nuclear Physics* (World Scientific, New York, 1990), p. 377.
9. M. R. STRAYER, C. BOTTCHER, V. E. OBERACKER, AND A. S. UMAR, *Phys. Rev. A* **41**, 1399 (1990).
10. C. BOTTCHER AND M. R. STRAYER, *Nucl. Instrum. Methods B* **31**, 122 (1988).
11. C. BOTTCHER, M. R. STRAYER, A. S. UMAR, AND P.-G. REINHARD, *Phys. Rev. A* **40**, 4182 (1989); A. S. UMAR, M. R. STRAYER, R. Y. CUSSON, P.-G. REINHARD, AND D. A. BROMLEY, *Phys. Rev. C* **32**, 172 (1985).