Parallel Implementation of Numerov’s Method Based Algorithm for Singularly Perturbed Boundary Value Problems

Rajesh K. Bawa and Amandeep Walia
Department of Computer Science
Punjabi University
Patiala, INDIA 147002

Abstract—In this paper, a parallel computational technique is proposed for solving self-adjoint singularly perturbed boundary value problems. After decomposing the domain into three non-overlapping subdomains, we devise three independent boundary-value problems and then using Numerov’s scheme for the boundary regions and reduced solution for the regular region, we solve them using three different processors. We have implemented the proposed scheme on parallel machine and speed-ups along with maximum errors are computed to show the efficiency of the scheme.

Index Terms—Domain decomposition, Numerov’s scheme, Parallel Computing, Singular Perturbation.

I. INTRODUCTION

Singular Perturbation Problems (SPP’s) arise in several branches of Computational Science and Engineering, which includes fluid dynamics, quantum mechanics, elasticity, chemical reactor theory, reaction-diffusion process etc. The presence of small parameter(s) in these problems prevents us from obtaining satisfactory approximate solutions. The solutions of SPP’s have a multi-scale character. That is, there are thin layer(s) where the solution varies very rapidly, while away from the layer(s) the solution behaves regularly and varies slowly.

In this article, we consider the following singularly perturbed self-adjoint boundary-value problem (BVP):

\[-\epsilon^2 u''(x) + b(x) u(x) = f(x), \ x \in D = (0, 1)\]  \hspace{1cm} (1)

with the boundary conditions

\[u(0) = A, \ u(1) = B, \]  \hspace{1cm} (2)

where,\(\epsilon > 0\) is a small positive parameter, \(A\) and \(B\) are given constants, \(b(x)\) and \(f(x)\) are sufficiently smooth functions.

Many sequential techniques are proposed by authors for solving problems of the form (1)-(2), for details on can refer to Miller, Riordan and Shishkin[1]. Also, many authors have proposed parallel techniques for solving these type of problems. Paprzycki and Gladwell[2] proposed a mesh chopping algorithm for solving SSP’s, Bogloaev [3] proposed a iterative algorithm for domain decomposition applied to singularly perturbed elliptic and parabolic SSP’s. Gracia, Lisbona and Clavero[4] have devised a HODIE scheme for singularly perturbed reaction-diffusion problems. Very recently, Bawa and Natesan [5] have proposed a quintic spline based computational method for such problems on sequential computer, which is suitable for parallelization. Using similar idea of domain decomposition and exploiting the multi-scale nature of solution, It is proposed to decompose the interval according to changing behavior of solution and solve the problem on parallel computer using Numerov’s scheme in order to reduce time complexity and to attain high accuracy.

II. PROPOSED PARALLEL ALGORITHM

The computational domain \(\bar{D} = [0, 1]\) is decomposed into three non-overlapping sub-domains, and then the Boundary Value Problem (BVP) (1), is solved subject to different boundary conditions in each of sub-domain.

Let \(k = \ln (N) > 0\), and \(k\epsilon\) be the width of the boundary layer(s) which is near at \(x = 0\), and \(x = 1\). The domain \(\bar{D}\) is divided into three non-overlapping sub domains, as:

\[D_1 = [0, k\epsilon], \ D_2 = [k\epsilon, 1 - k\epsilon],\ \text{and} \ D_3 = [1 - k\epsilon, 1],\]

such that \(\bar{D} = D_1 \cup D_2 \cup D_3\).

The sub-domains \(D_1\) and \(D_3\) are called the boundary layer regions, whereas \(D_2\) is known as the regular region.

The BVPs correspond to the left and right boundary layers are:

\[-\epsilon^2 u''(x) + b(x) u(x) = f(x), \ x \in D_1 = (0, k\epsilon)\]  \hspace{1cm} (3)

\[u(0) = A, \ u(k\epsilon) = \bar{A},\]  \hspace{1cm} (4)

and

\[-\epsilon^2 u''(x) + b(x) u(x) = f(x), \ x \in D_3 = (1-k\epsilon, 1)\]  \hspace{1cm} (5)

\[u(1-k\epsilon) = \bar{B}, \ u(1) = B,\]  \hspace{1cm} (6)

The width of the boundary layers is \(O(\epsilon)\), in order to magnify the boundary layers, stretching variables for the left and right boundary layers are used which are

\[T = xk\epsilon, \ \text{and} \ \eta = (1-x)/\epsilon,\]

and the transformed BVPs are given as:

...
-U_1''(T) + B(T) U_1(T) = F(T), \quad T \in (0, k) \tag{7}
U_1(0) = A, \quad U_1(k) = \bar{A} \tag{8}
and
-\varepsilon^2 u''(\eta) + B(\eta) U_2(\eta) = F(\eta), \quad \eta \in (0, k) \tag{9}
U_2(0) = B, \quad U_2(k) = \bar{B} \tag{10}

The regular region BVP is given by:

\text{-\varepsilon^2 u''(x) + b(x) u(x) = f(x), \quad x \in D_3 = (k\varepsilon, 1-k\varepsilon)} \tag{11}
u(k\varepsilon) = \bar{A}, \quad u(1-k\varepsilon) = \bar{B} \tag{12}

To determine the boundary conditions at the transition points, we take the zeroth-order asymptotic approximation of the BVP (1-2), given by

\tilde{u}(x) = u_0(x) + v_0(T) + w_0(\eta), \tag{13}

where \( u_0(x) = f(x)/b(x) \) is the reduced problem solution, and \( v_0 \) and \( w_0 \) are respectively the left and right boundary layer correction terms

\( v_0(T) = [A - u_0(0)] \exp(-(v^2 b(0)) x/\varepsilon) \tag{14} \)

\( w_0(\eta) = [B - u_0(1)] \exp(-(v^2 b(1)) (1-x)/\varepsilon) \tag{15} \)

The values \( \bar{A} \) and \( \bar{B} \) are given by

\( \bar{A} = \tilde{u}(k\varepsilon), \quad \text{and} \quad \bar{B} = \tilde{u}(1-k\varepsilon) \tag{16} \)

Here, domain is divided into three sub-domains: Two boundary layer sub-domains, and one regular sub-domain, and we have converted the boundary layer problems to a regular one by proper transformations using stretching variables and then we apply a difference scheme based on Numerov’s method to obtain the numerical solution in the first and last domain, and for the middle domain, solution of the reduced problem is taken as approximate solution. To obtain the reduced solution we put \( \varepsilon = 0 \) in the differential equation (1) and solve it.

To obtain the boundary condition at the transition points, we use an asymptotic approximate solution. By this way, parallel computers can be used to reduce the computation time and maximum error by solving boundary layers and regular subdomain problems on three different processors.

III. IMPLEMENTATION

Here, we discuss the parallel implementation of the proposed algorithm. The computational domain \( \bar{D} \) has been decomposed as

\( \bar{D} = [0, k\varepsilon] \cup [k\varepsilon, 1-k\varepsilon] \cup [1-k\varepsilon, 1]. \)

and the three BVPs (3-4), (11-12), and (5-6) respectively defined on these sub-domains. Further, the BVPs (3-4), and (5-6) are transformed to the BVPs (7-8) and (9-10) respectively.

A. Algorithm:

Step 1. Solve the BVP (7-8) in \( (0, k) \) using Numerov’s difference scheme.

Step 2. Solve the BVP (11-12) in \( [k\varepsilon, 1-k\varepsilon] \) using reduced method.

Step 3. Solve the BVP (9-10) in \( (0, k) \) using Numerov’s difference scheme.

Obviously the parallelism is inherent in the above algorithm. Hence, separate processors can be assigned for Steps 1, 2 and 3. This leads to the following three-processor scheme.

B. Parallel Scheme:

Task 1. Perform Step 1 on Processor \( P_1 \) (Slave 1).

Task 2. Perform Step 2 on Processor \( P \) (Master).

Task 3. Perform Step 3 on Processor \( P_2 \) (Slave 2).

The parallel scheme is presented in tabulated form below in Table I.

<table>
<thead>
<tr>
<th>Table I</th>
<th>COMMUNICATION CHART</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Processor ( P_1 ) (Slave 1)</td>
</tr>
<tr>
<td></td>
<td>Computation</td>
</tr>
<tr>
<td></td>
<td>Communication</td>
</tr>
</tbody>
</table>

IV. NUMERICAL ILLUSTRATION

We have implemented the proposed parallel algorithm on a test problem for different values of \( N \) and \( \varepsilon \). In order to demonstrate the efficiency of the algorithm, the maximum error obtained by using Numerov’s scheme on piece-wise uniform(shishkin mesh) proposed by Sun and Stynes[6] are also presented for comparison purpose.

Example: Consider the non-homogenous self-adjoint SPP

\[-\varepsilon^2 u''(x)+u(x)=-\cos^2(\pi x) -2\varepsilon^2 \pi^2 \cos(2\pi x), \tag{17}\]
\[u(0) = 0, \quad u(1) = 0, \tag{18}\]

The exact solution is given by

\[\tilde{u}(x) = \frac{\exp(-x/\varepsilon) + \exp(-(1-x)/\varepsilon)}{[1-\exp(-1/\varepsilon) - \cos^2(\pi x)}

The transition boundary conditions is given by

\[\tilde{u}(x) = -\cos^2(\pi x) + \exp(-x/\varepsilon) + \exp(-(1-x)/\varepsilon) \tag{19}\]
TABLE II
MAX. ERROR USING PROPOSED SCHEME

<table>
<thead>
<tr>
<th>ε</th>
<th>N</th>
<th>D₁</th>
<th>D₂</th>
<th>D₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>2⁻¹⁰</td>
<td>32</td>
<td>5.356e-03</td>
<td>0.0</td>
<td>5.356e-03</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>8.638e-03</td>
<td>0.0</td>
<td>8.638e-03</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>1.002e-04</td>
<td>0.0</td>
<td>1.002e-04</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>1.099e-05</td>
<td>0.0</td>
<td>1.099e-05</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>1.104e-06</td>
<td>0.0</td>
<td>1.104e-06</td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>1.053e-07</td>
<td>0.0</td>
<td>1.053e-07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ε</th>
<th>N</th>
<th>D₁</th>
<th>D₂</th>
<th>D₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>2⁻²⁰</td>
<td>32</td>
<td>5.356e-03</td>
<td>0.0</td>
<td>5.356e-03</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>8.638e-03</td>
<td>0.0</td>
<td>8.638e-03</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>1.002e-04</td>
<td>0.0</td>
<td>1.002e-04</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>1.099e-05</td>
<td>0.0</td>
<td>1.099e-05</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>1.104e-06</td>
<td>0.0</td>
<td>1.104e-06</td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>1.053e-07</td>
<td>0.0</td>
<td>1.053e-07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ε</th>
<th>N</th>
<th>D₁</th>
<th>D₂</th>
<th>D₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>2⁻³⁰</td>
<td>32</td>
<td>5.356e-03</td>
<td>0.0</td>
<td>5.356e-03</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>8.38e-03</td>
<td>0.0</td>
<td>8.638e-03</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>1.002e-04</td>
<td>0.0</td>
<td>1.002e-04</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>1.099e-05</td>
<td>0.0</td>
<td>1.099e-05</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>1.104e-06</td>
<td>0.0</td>
<td>1.104e-06</td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>1.053e-07</td>
<td>0.0</td>
<td>1.053e-07</td>
</tr>
</tbody>
</table>

TABLE III
MAX. ERROR USING NUMEROV’S SCHEME ON SHISHKIN’S MESH

<table>
<thead>
<tr>
<th>Case</th>
<th>N</th>
<th>ε</th>
<th>2⁻¹⁰</th>
<th>2⁻²⁰</th>
<th>2⁻³⁰</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32</td>
<td>5.356e-03</td>
<td>5.356e-03</td>
<td>5.356e-03</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>64</td>
<td>8.638e-04</td>
<td>8.638e-04</td>
<td>8.638e-04</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>256</td>
<td>1.099e-05</td>
<td>1.099e-05</td>
<td>1.099e-05</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>512</td>
<td>1.104e-06</td>
<td>1.104e-06</td>
<td>1.104e-06</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1024</td>
<td>1.053e-07</td>
<td>1.053e-07</td>
<td>1.053e-07</td>
<td></td>
</tr>
</tbody>
</table>

we have tabulated the results obtained after implementing the Parallel algorithm using domain decomposition, proposed in the earlier section of this article, on a test problems mentioned above. The results are presented in Table II. The results obtained using Piecewise Uniform mesh (Shishkin) are also tabulated in Table III. The results are presented as the maximum absolute nodal error between the exact solution and the approximated solution for the various values of ε and N.

Fig. 1 shows the comparison of timings obtained by applying proposed in parallel and applying Numerov’s scheme on Shishkin’s mesh sequentially, while fig. 2. shows the achieved speed-ups.

V. CONCLUSION
The conventional techniques for solving boundary value problems fails to give satisfactory results for singularly perturbed problems because of multi-scale character of the solution.

The sequential algorithms devised on piecewise meshes gives acceptable results but in the mean time it becomes cumbersome to derive these schemes for these meshes. The implementation of proposed technique avoids these difficulties, decompose the task in such a way that simple higher order schemes, such as, Numerov’s scheme can be applied to give same results (in terms of maximum error) as on that of well known Shishkin’s mesh, but in the meantime reducing the time factor to large extend.
REFERENCES


