A Parallel Numerical Algorithm For Boundary - Value FIDEs on a PC Cluster

R.E.Shaw†, L.E.Garey† and D.J.Lizotte
University of New Brunswick
P.O. Box 5050, Saint John, NB,
Canada E2L 4L5

Abstract

An algorithm for parallel processing the discrete nonlinear system for solving Fredholm integro-differential equations with two-point boundary conditions on a PC cluster is considered. The cost of calculating the history terms is expensive and improvements are motivated by considering different architectures. The algorithm has been modified to minimize the communication overhead inherent in a distributed application and tested using two different messaging protocols: GAMMA, an efficient message system for clusters of PCs [1] with low latency and high throughput and TCP/IP protocol with MPI. Numerical examples illustrate the results.

Keywords: FIDEs, PC Clusters, parallel processing, Toeplitz systems

C.R. Category : G.1.9

1 Introduction

In a paper [7] by Shaw, Garey and Lizotte, numerical methods for a nonlinear second order Fredholm integro-differential equations with two-point boundary conditions were developed. An unresolved problem was that the history term portion of the solution was re-evaluated with each iteration. The coefficient matrix in the numerical approximation is nonsymmetric but is identified with a banded symmetric coefficient matrix and an additional sparse matrix. Taking advantage of this led to the application of a parallel algorithm for the numerical solution.

Consider a nonlinear second order Fredholm integro-differential equation (FIDE) of the form

\[ y'' = f(x, y, z), \]

\[ y(0) = y_0, \quad y(a) = y_a \]

\[ z(x) = \int_0^a K(x, t, y(t))dt, 0 \leq x \leq a \]

The function \( y(x) \) is unknown. For equation (1) defined for points in

\[ S = \{(x, y, z) : 0 \leq x \leq a, |y| < \infty, |z| < \infty\} \]

and

\[ T = \{(x, t, y) : 0 \leq t, x \leq a, |y| < \infty\}, \]

we assume:

i) \( f \) and \( K \) are uniformly continuous in each variable

ii) for all \( (x, y, z), (x, \bar{y}, \bar{z}), \) and \( (x, y, \bar{z}) \) in \( S \), \( f \) satisfies

\[ |f(x, y, z) - f(x, \bar{y}, \bar{z})| < L_1|y - \bar{y}| \]

\[ |f(x, y, z) - f(x, y, \bar{z})| < L_2|z - \bar{z}| \]

iii) for all \( (x, t, y) \) and \( (x, t, \bar{y}) \) in \( T \), \( K \) satisfies

\[ |K(x, t, y) - K(x, t, \bar{y})| \leq L_0|y - \bar{y}| \]

iv) the functions \( f_y, f_z \) and \( K_y \) are continuous and satisfy \( f_y \geq 0, f_z \geq 0 \) and \( K_y \geq 0 \) for all points in \( S \) and \( T \).

Let \([0, a]\) be a partition with

\[ I_N = \{x_n : x_n = nh, n = 0(1)N, h > 0, Nh = a\}. \]
A general k-step method of solution is defined by

\[ \sum_{i=0}^{k} a_i y_{n+i} = h^2 \sum_{i=0}^{k} \beta_i f(x_{n+i}, y_{n+i}, z_{n+i}), \]

\[ n = 0(1)N - k \]

with \( z_n = h \sum_{j=0}^{N} w_j K(x_n, x_j, y_j), n > s, z_0 = 0 \) where \( y_t \) denotes an approximation to \( y(x_t) \) and where \( \{w_j\} \) denote the weights of the quadrature rule, \( s \) is related to the order of the method and \( \{a_i\} \) and \( \{\beta_i\} \) are, the coefficients of the polynomials \( (p, \sigma) \). For equation \( (2) \), \( \rho(z) = \sum_{i=0}^{k} a_i z^i, \sigma(z) = \sum_{i=0}^{k} \beta_i z^i \). Methods for integro-differential equations are denoted by triples \( ((p, \sigma), Q, IP) \) where \( Q \) denotes the quadrature rule to approximate the integral and \( IP \) is the iteration procedure chosen. Here we consider, as an example, the five point rule

\[ \rho(z) = z^4 - 16z^3 - 30z^2 - 16z + 1, \]
\[ \rho(z) = -12z^2 \]

A Newton-Cotes or Newton-Gregory quadrature rule which is compatible with the order of the method defined by \( (\rho, \sigma) \) is used. The two natural auxiliary conditions [7] for this method are given by

\[ 10y_0 - 15y_1 - 4y_2 + 14y_3 - 6y_4 + y_5 \]
\[ = 12h^2 f(x_1, y_1, z_1) \]

and

\[ y_{N-5} - 6y_{N-4} + 14y_{N-3} - 4y_{N-2} - 15y_{N-1} + 10y_N \]
\[ = 12h^2 f(x_{N-1}, y_{N-1}, z_{N-1}). \]

Adding these, respectively, as the first and \( (N - 1) \)th equations in the system, we have

\[ CY = 12h^2 F(Y) + R, \]

\[ C = \begin{pmatrix} -15 & -4 & 14 & -6 & 1 \\ 16 & -30 & 16 & -1 & 0 \\ -1 & 16 & -30 & 16 & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & -6 & 14 & -4 & -15 \end{pmatrix} \]

\[ F(Y) = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N-5} \\ f_{N-1} \end{pmatrix}, \quad R = \begin{pmatrix} -10y_0 \\ y_0 \\ \vdots \\ y_{N-5} \\ -10y_N \end{pmatrix} \]

With the boundary conditions substituted, we have an \( N - 1 \times N - 1 \) system leading to the terms in \( R \). This method solves the problem in its given form. For other solutions, see [3]. System (3) is solved iteratively (IP). The coefficient matrix \( C \) is banded and near Toeplitz. Yan and Chung [8] provide a Toeplitz LU factorization of Toeplitz matrices following a perturbation of the given system. In [2], we have a treatment of five-band systems which can be factored. Shaw [6] provides insight into the processing of the right hand side of system (3). We consider improvements to the iterative solution of system (3) by processing the approximations to the integral terms and by implementing a parallel version of the method in [8] as part of the iterative solution of the factored system. We conclude with two numerical examples.

# 2 Parallel Algorithm

Let us consider a system to be solved iteratively of the form

\[ A z^{(i+1)} = b_i \equiv b(z^{(i)}) \]

\[ A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ \beta & \alpha & \beta & \gamma & 0 \\ \gamma & \beta & \alpha & \beta & \gamma \\ \vdots & \ddots & \ddots & \ddots & \ddots \\ 0 & \gamma & \beta & \alpha & \beta \\ a_{10} & a_{14} & a_{13} & a_{12} & a_{11} \end{pmatrix} \]

where \( A \) is nonsingular. The matrix \( A \) can be written as \( A = B + E \) with

\[ B = \begin{pmatrix} \alpha' & \beta & \gamma & \beta & \gamma \\ \beta & \alpha & \beta & \gamma & 0 \\ \gamma & \beta & \alpha & \beta & \gamma \\ \vdots & \ddots & \ddots & \ddots & \ddots \\ 0 & \gamma & \beta & \alpha & \beta \\ \gamma & \beta & \alpha & \beta & \gamma \end{pmatrix} \]

and

\[ E = \begin{pmatrix} a_{11} - a' & a_{12} - \beta & a_{13} - \gamma & a_{14} & a_{15} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ a_{10} & a_{14} & a_{13} & a_{12} & a_{11} \end{pmatrix} \]

where \( \alpha' = \alpha - \gamma \). \( B \) is a near Toeplitz, symmetric, sparse and banded matrix. In our example, \( \gamma = 1 \). In general, for \( \gamma \neq 0 \), let \( \gamma D = B \) and write

\[ D z^{(i+1)} = \frac{1}{\gamma} b_i \equiv \hat{b}_i \]
where

\[
D = \begin{pmatrix}
\alpha' & \beta & 1 \\
\beta & \alpha' & \beta & 1 \\
& \ddots & \ddots & \ddots \\
1 & \beta & \alpha' & \beta & 1
\end{pmatrix}
\]

D can be factored [2] into \( D_1 = \{1, d_{11}, 1\} \) and \( D_2 = \{1, d_{22}, 1\} \) with

\[
d_{1,2} = \frac{\beta \pm \sqrt{\beta^2 - 4(\alpha - 2)}}{2}.
\]

For the coefficient matrix C in system (3), we have \( C_1 = \{-1, 2, -1\} \) and \( C_3 = \{-1, 14, 1\} \). For the matrix D, let \( D_2 \) be a diagonally dominant Toeplitz matrix. Then \( D_2 D_1 x^{(i+1)} = \hat{b}_i \) and we solve a pair of systems

\[
D_2 x^{(i+1)} = \hat{b}_i \quad \text{and} \quad D_1 x^{(i+1)} = x^{(i+1)}
\]

Let \( D_2 = D_{21} + D_{22} \) where

\[
D_{22} = \begin{bmatrix}
-b & 0 \\
1 & -b \end{bmatrix}
\]

and \( D_{21} \) is a block diagonal matrix with two tridiagonal matrices \( D_{21,j} \). Each diagonal block in \( D_{21} \) is taken to be of size \( k = n/2 \) for \( n \) even. Otherwise, the sizes are \( m = k + 1 \) and \( k \). Again, a perturbed system is to be solved

\[
D_{21} x^{(i+1)} = \hat{b}_i
\]

This can be written as two independent systems

\[
D_{21,m} x^{(i+1)} = \hat{b}_{i,m}
\]

and

\[
D_{21,k} x^{(i+1)} = \hat{b}_{i,k}
\]

and solved in parallel. \( D_{21,j} \), where \( j = m \) or \( k \), is given by

\[
D_{21,j} = \begin{bmatrix}
a & 1 & 1 \\
1 & d & 1 \\
& \ddots & \ddots & \ddots \\
1 & d & 1 \\
1 & d & 1
\end{bmatrix}
\]

With reference to the FIDE, recall that the \( \nu \)th component of \( \hat{b}_i \) is given by

\[
\hat{b}_\nu = \frac{1}{\gamma} h^2 \sum_{j=0}^{k} \beta_j f(x_{\nu+j-1}, y^{(i)}_{\nu+j-1}, z_{\nu+j-1})
\]

\[
z_{\nu+j-1} = h \sum_{m=0}^{N} w_m K(x_{\nu+j-1}, x_m, y^{(i)}_m)
\]

Note that \( y^{(i)}_0 = y_0 \) and \( y^{(i)}_m = y_m \forall t \).

The right hand side of each subsystem is solved in parallel using block decomposition. Collective communication is used to gather the most recent version of the solution vector from each process at the end of each iteration. This idea can be extended to \( p \) subsystems.

### 3 Correcting Perturbed System

Two systems have been perturbed. We consider first correcting for \( D_{22} \). Denoting the solution for the pair of systems (8), by \( x^{(i+1)} \), we have

\[
D_{22} x^{(i+1)} = D_{21} x^{(i+1)} + D_{22} \tilde{x}^{(i+1)}
\]

and

\[
D_{21} x^{(i+1)} = D_{21} x^{(i+1)} + D_{22} \tilde{x}^{(i+1)}.
\]

When this method is used, we express \( x^{(i+1)}_1 \) in terms of \( x^{(i+1)} \) and a correction term before the second system involving \( D_1 \) is solved. Multiplying by \( D_{21}^{-1} \) yields

\[
x^{(i+1)}_1 = x^{(i+1)}_1 - D_{21}^{-1} D_{22} x^{(i+1)}.
\]

The matrix \( D_{22} \) has only four nonzero elements so we can write

\[
x^{(i+1)}_1 = x^{(i+1)}_1 - D_{21}^{-1} D_{22} x^{(i+1)}.
\]

To avoid calculating \( D_{21}^{-1} \), vectors \( p, q \) and \( r \) need to be determined such that

\[
D_2 p = e_1,
\]

\[
D_2 q = e_m
\]

and

\[
D_2 r = e_{m+1}
\]

Let \( p = \{b, b^2, \ldots, b^t, 0, 0, \ldots, 0\}^T \) where \( |b| < 1 \). Then [8]

\[
D_2 p = -e_1 + c_1 (e_{t+1} - be_{t}).
\]

Let

\[
q_n = \{0, \ldots, 0, b^t, b^t-1, \ldots, b^0, 0, \ldots, 0\}^T
\]
and 

\[ r = \{0, \ldots, 0, b, b^2, \ldots, b^t, 0, \ldots, 0\}^T. \]

From [5], we find

\[ D_2 q = -e_m + b e_{m+1} + b^t (e_{m-t} - b e_{m-t+1}) \]

and

\[ D_2 r = -e_{m+1} + b e_m + b^t (e_{m+t+1} - b e_{m+t}). \]

Considering expressions for

\[ D_2 \mathbf{q} = -e_m + b e_{m+1} + b^t (e_{m-t} - b e_{m-t+1}) \]

and

\[ D_2 \mathbf{r} = -e_{m+1} + b e_m + b^t (e_{m+t+1} - b e_{m+t}). \]

we can write

\[ x_1^{(i+1)} = x_1^{(i+1)} - b \bar{x}_1^{(i+1)} \mathbf{p} \]  

Recall that \( A \mathbf{x}^{(i+1)} = \mathbf{b}(\mathbf{x}^{(i)}) \) and \( B \mathbf{x}^{(i+1)} = \mathbf{b}(\mathbf{x}^{(i)}) \) and we can rewrite (10) as

\[ x^{(i+1)} = x^{(i+1)} - A^{-1}E \mathbf{x}^{(i+1)} \]

The second correction relates to matrix \( E \) in system (4). The solution \( \mathbf{x}^{(i+1)} \) coming from equation (6) is the solution to \( B \mathbf{x} = \mathbf{b} \). Consider

\[ A \mathbf{x}^{(i+1)} = B \mathbf{x}^{(i+1)} + E \mathbf{x}^{(i+1)}. \]

(9) See [4], p.360-371).
4 Numerical Examples

Two nonlinear examples are solved by the two methods employed in the iteration process that arises. Newton-Gregory fourth order weights are used for the integration.

Example 1. Exact solution: \( y(x) = 1/(x + 1) \).
\[
y'' = 2y^2/(x + 1) - x + \int_0^1 x(t + 1)y(t)dt , y(0) = 0, y(1) = 1/2
\]

Example 2. Exact solution: \( y(x) = e^{-x} \).
\[
y'' = y + 2x - (2x + 1)e^2 + 3 + \int_0^1 (x + t)y(t)^2dt , y(0) = 1, y(1) = 1/e
\]

Algorithm: Solving a Nonlinear FIDE

Given:
- Non-symmetric coefficient matrix \( A \)
- Right hand vector expression (nonlinear in unknown vector \( y \ ))

Determine: Symmetric 5-band matrix \( B \) from (5)

Factor \( B = D_2 D_1 \),
\[
D_1 = \{ 1, d_1, 1 \}, \quad D_2 = \{ 1, d_2, 1 \}
\]

For \( D_2 \), calculate
\[
a = \frac{-d_2 \pm \sqrt{(d_2^2 - 4)}}{2}, \quad |a| > 1
\]
\[
b = a - d_2
\]

Rewrite \( D_2 = D_{21} + D_{22} \)

Select a tolerance \( \tau \) where
\[
t(\tau) = \frac{\log(2) + \log(\tau^2) + \log((b^2 - 1)) - \log(2) \log(2))}{\log(2)}
\]

For \( i = 0 \), select an initial vector \( y^{(i)} = y^{(i)} \)

Loop:
- Evaluate \( \hat{b}_i \) in parallel
- Solve in parallel
\[
D_{21,m} \hat{y}^{(i+1)}_{1,m} = \hat{b}_{i,m}
\]

and
\[
D_{21,k} \hat{y}^{(i+1)}_{1,k} = \hat{b}_{i,k}
\]

Correct \( y^{(i+1)} \) using
\[
y^{(i+1)}_1 = y^{(i+1)}_1 - bg^{(i+1)}_1 + p \\
\]
\[
\frac{1}{1 - b^2} ((bg^{(i+1)}_{1,m+1} - \hat{y}^{(i+1)}_{1,m}) (x + bq) - \hat{y}^{(i+1)}_{1,m+1} (q + br))
\]

Solve \( D_1 \hat{y}^{(i+1)} = y^{(i+1)}_1 \)

Correct \( \hat{y}^{(i+1)} \) using
\[
y^{(i+1)}_1 = \hat{y}^{(i+1)}_1 - z^{(i+1)}_1 p - z^{(i+1)}_{N-1} Q
\]

Test the Approximation
\[
\text{if } ([y^{(i+1)}_1 - y^{(i)}_1] \geq \tau), \quad i \leftarrow i + 1 \quad \hat{y}^{(i)} \leftarrow y^{(i)}
\]

else
\[
y \leftarrow y^{(i+1)}
\]

For the results in Table I, the iteration procedure was terminated using a tolerance of \( h^4 \). The size of the system was chosen to be 1500 for each example. For Example 1, 21 iterations are required for convergence within the tolerance and for Example 2, it took 20 iterations when \( p = 1 \) and 18 iterations for \( p > 1 \). This gave rise to superlinear speedup for this example and the efficiency was greater than one. The algorithm doesn’t reflect the details but \( D_1 \) was written in the form \( D_{11} + D_{12} \). Each was perturbed and factored into two bidiagonal matrices. The problems were solved sequentially and in parallel using a cluster of 9 PCs running Linux. The cluster is interconnected by standard Fast Ethernet. The TCP/IP protocol shows a high communication latency and low throughput so an experimental system for inter-process communication called GAMMA [1] was installed and tested. The algorithms are run on both message systems and the results reflect the improvement in performance achieved by GAMMA over standard TCP/IP communication protocol with the MPI message passing library.

The algorithms were coded in Fortran 77 using the g77 gnu compiler. The right hand side of system(3) involved the integration of the integral over \([0, 1]\) for each equation in the system. The resulting system of equations were mapped over the \( p \) processors using block decomposition.
Table I. Gamma vers LAM/MPI for a linear system

<table>
<thead>
<tr>
<th>Example</th>
<th>Processors</th>
<th>Time(s)</th>
<th>Speedup</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>Gamma</td>
<td>2.01472</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Gamma</td>
<td>1.03334</td>
<td>1.94972</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Gamma</td>
<td>0.74032</td>
<td>2.72142</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Gamma</td>
<td>0.54704</td>
<td>3.68293</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>1</td>
<td>2.01472</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>2</td>
<td>1.06895</td>
<td>1.884773</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>3</td>
<td>0.80038</td>
<td>2.51720</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>4</td>
<td>0.61881</td>
<td>3.25570</td>
</tr>
<tr>
<td>2</td>
<td>Gamma</td>
<td>1</td>
<td>2.37067</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Gamma</td>
<td>1.99632</td>
<td>2.16239</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Gamma</td>
<td>0.75899</td>
<td>3.12343</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>1</td>
<td>2.37067</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>2</td>
<td>1.13196</td>
<td>2.09429</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>3</td>
<td>0.80024</td>
<td>2.96246</td>
</tr>
<tr>
<td></td>
<td>LAM/MPI</td>
<td>4</td>
<td>0.64618</td>
<td>3.66876</td>
</tr>
</tbody>
</table>

References


