A modified iterative refinement algorithm for efficient solution of parameter-dependent sets of linear equations

Space Syst./Loral, Palo Alto, CA, USA ;

This paper appears in: Antennas and Propagation Society International Symposium, 1998. IEEE
Publication Date: 21-26 June 1998
Volume: 3
On page(s): 1510 - 1513 vol.3
Number of Pages: 4 vol. viii+2354
Meeting Date: 06/21/1998 - 06/26/1998
Location: Atlanta, GA

Abstract
A modified iterative refinement (MIR) algorithm is presented that reduces the time required to solve systems of linear equations that arise when using the method of moments to analyze electromagnetic problems over a band of frequencies. In such an analysis one typically factors the system matrix and solves the linear system at each analysis frequency. With MIR, the LU factorization of the system matrix obtained at one frequency is used to compute solutions at several additional nearby frequencies, often resulting in a major savings in computational effort. Since MIR is not an interpolation technique, answers obtained are numerically exact to any desired degree of precision

Index Terms
Available to subscribers and IEEE members.

References
Available to subscribers and IEEE members.

Citing Documents
Available to subscribers and IEEE members.
A Modified Iterative Refinement Algorithm for Efficient Solution of Parameter-Dependent Sets of Linear Equations

Peter S. Simon*, Charles S. Kenney**, Kim McInturff†, Robert W. Jobsky†, and Thomas A. Bryan††

*Space Systems/Loral, Palo Alto, CA
**ECE Dept., University of California, Santa Barbara
†Raytheon E-Systems, Goleta, CA
††The Mathworks, Natick, MA

Abstract

A modified iterative refinement (MIR) algorithm is presented that reduces the time required to solve systems of linear equations that arise when using the method of moments to analyze electromagnetic problems over a band of frequencies. In such an analysis one typically factors the system matrix and solves the linear system at each analysis frequency. With MIR, the LU factorization of the system matrix obtained at one frequency is used to compute solutions at several additional nearby frequencies, often resulting in a major savings in computational effort. Since MIR is not an interpolation technique, answers obtained are numerically exact to any desired degree of precision.

Introduction

Several interpolation techniques have been proposed in the literature [1], [2] to reduce the time needed to repetitively analyze a structure over a band of frequencies using the method of moments. These methods are quite useful but they both require some means of estimating the frequency sampling rate to determine the interpolation knots. A finite amount of interpolation error is incurred with either of these techniques, the precise amount of which is not always easy to quantify. We present here an alternative numerical method termed “modified iterative refinement” or “MIR” which is well suited for large problems where solution time is dominant. It reduces the execution time by re-using the factorization obtained at a “start” frequency, as described below.

MIR Algorithm Description

Suppose that we wish to solve the frequency-dependent system of equations

$$Ax = b$$

where the matrix $A$ and vector $b$ are obtained at each frequency of interest from the method of moments, finite element method, or some other frequency domain formalism. Assume also that we have available an approximation $M$ which is “nearly equal” to $A^{-1}$. We now perform a series of iterative refinements (pp. 126-127 of [3]), but using $M$ rather than $A^{-1}$ to obtain the iterates. The residuals are computed as usual in double precision using $A$.

$$x^{(1)} = Mb, \quad r^{(1)} = b - Ax^{(1)}$$
$$x^{(2)} = x^{(1)} + Mr^{(1)}, \quad r^{(2)} = b - Ax^{(2)}$$
$$x^{(3)} = x^{(2)} + Mr^{(2)}, \quad r^{(3)} = b - Ax^{(3)}$$
$$\vdots$$

1
It can be shown that $x^{(i)}$ converges to the solution $x$ of $Ax = b$, independently of the choice of the initial vector $x^{(0)}$, if and only if $M$ is chosen “sufficiently close” to the inverse of $A$. More precisely, convergence occurs when the spectral radius of the matrix $I - AM$ is less than unity, where $I$ is the identity matrix. Because of space limitations, the straightforward proof of this statement is omitted.

The basic idea of MIR is thus very similar to standard iterative refinement, where one presumes that $A^{-1}$ (single precision) is close to $A^{-1}$ (double precision) in the same sense described above. We note that factoring a matrix $A$ of order $m$ requires on the order of $m^3$ operations, but once the factors are known, solving the system of equations requires only order $m^2$ operations. There is thus the potential for execution time savings if the same $M$ can be reused for multiple analysis frequencies. An obvious choice for $M$—that used in this study—is the matrix $A^{-1}$ evaluated at a nearby previously analyzed frequency.

We now show how MIR is implemented as an actual computer algorithm. A frequency-dependent set of linear equations is to be solved at the frequencies $f_1 < f_2 < \cdots < f_{N_f}$. The equations are written at the $n$th frequency as $A_n x_n = b_n$, where $A_n \equiv A(f_n) \in \mathbb{C}^{m \times m}$, $x_n \equiv x(f_n) \in \mathbb{C}^m$, and $b_n \equiv b(f_n) \in \mathbb{C}^m$. The standard procedure [3] is to obtain via Gaussian elimination the lower and upper triangular factors at each frequency (algorithmically denoted as $LU := \text{decomp}(A_n)$), and then to solve the system of linear equations by means of forward and back substitution ($x_n := \text{solve}(LU, b_n)$).

In Figure 1 we present a pseudo code listing for the MIR algorithm. The idea is to use the LU factorization of $A_n$ to solve the hopefully nearby problems $A_{n+1} x_{n+1} = y_{n+1}$, etc., until the residuals no longer appear to be converging to zero, at which time a new LU factorization is performed. This eliminates the requirement for factoring at many analysis frequencies. As shown below, for large problems the time saved by avoiding factoring far outweighs the extra effort expended in non-convergent iterations.

**Numerical Example**

As a demonstration we applied our algorithm to the PATCH code [4], which analyzes scattering from arbitrary metallic objects. The results from the original version of the code are labeled “standard.” We then created a version of the code which implements the MIR algorithm of Figure 1. The results from this version are labeled “MIR.”

The problem we selected consists of calculating normal incidence radar cross section (RCS) of an ideal square metallic plate of dimension $a = 1$ meter. The RCS was calculated at twenty equally spaced frequencies with $a/\lambda = 0.05, 0.10, 0.15, \ldots, 1.00$, using discretizations of 176, 736, and 1408 unknowns, where $\lambda$ is the free-space wavelength. All computations were performed on a dedicated Sun SPARCstation 2. For this test we used $tol = 5 \times 10^{-5}$ and $N_t = 46$, values which had been empirically determined on a different, larger problem. We did not attempt to find optimum values of these parameters.

We display timing histograms for the 1408-unknown discretization in parts (a) and (b), respectively, of Figure 2. Examining part (a) of the figure, we see that matrix fill time and matrix equation solution time do not vary with frequency using the standard algorithm, and that solution time dominates for this problem size. Timing results for the MIR algorithm are shown in part (b) of the figure, where it is seen that the timing details for solution of the matrix equation using MIR differ markedly from the standard algorithm. Only six (instead of twenty) LU decompositions were required. We also note that as the analysis frequency increases, the spacing between frequencies at which a new LU decomposition is required also increases, an observation which is consistent with the comments in [1].
$LUflag := true$

for $n := 1$ to $N_f$ do
    $NewLU$:
    if $LUflag$
        $LU := \text{decomp}(A_n)$
        $LUflag := false$
    endif
    $y := \text{solve}(LU, b_n)$
    $\varepsilon_{\text{old}} := +\infty$
    for $i := 1$ to $N_t$ do
        $r := b_n - A_n y$
        $\varepsilon := ||r||$
        $y := y + \text{solve}(LU, r)$
        if $\varepsilon / ||b_n|| < \text{tol}$
            $x_n := y$
            goto $NewF$
        elseif $\varepsilon > \varepsilon_{\text{old}}$
            $LUflag := true$
            goto $NewLU$
        endif
        $\varepsilon_{\text{old}} := \varepsilon$
    enddo
    $LUflag := true$
    goto $NewLU$
enddo

End of frequency loop

Begin iteration loop

Compute residual (double precision)

Iterative improvement step

$n$th solution has been obtained

Skip to next frequency

Solution is diverging

End of iteration loop

No convergence within $N_t$ iterations

End of frequency loop

Figure 1: Modified iterative refinement algorithm description.

![CPU Time (sec) vs Frequency Index](a)

![CPU Time (sec) vs Frequency Index](b)

Figure 2: Execution time histograms for the 1408 unknown discretization using the (a) standard algorithm and (b) MIR algorithm.
regarding “logarithmic spacing.” Examination also reveals that the amount of CPU time used in iterative refinements increases with separation from the previous LU frequency, since the spectral radius of $I - AA_0^{-1}$ increases and requires more iterations to achieve a given accuracy.

Because of the additional computations needed to perform the iterative refinement, the time savings relative to the standard algorithm are negligible for 176 unknowns. However, as the number of unknowns increases, the savings become dramatic, as shown in Figure 2 and summarized in Table 1. We see that the total execution time is reduced by roughly 38% and 51% for the cases of 736 and 1408 unknowns, respectively.

<table>
<thead>
<tr>
<th></th>
<th>176 Unknowns</th>
<th></th>
<th>736 Unknowns</th>
<th></th>
<th>1408 Unknowns</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Standard</td>
<td>MIR</td>
<td>Standard</td>
<td>MIR</td>
<td>Standard</td>
</tr>
<tr>
<td>Fill</td>
<td>124</td>
<td>123</td>
<td>1758</td>
<td>1741</td>
<td>5931</td>
</tr>
<tr>
<td>Solve</td>
<td>63</td>
<td>59</td>
<td>4155</td>
<td>1941</td>
<td>28895</td>
</tr>
<tr>
<td>Total</td>
<td>189</td>
<td>183</td>
<td>5929</td>
<td>3698</td>
<td>34893</td>
</tr>
</tbody>
</table>

Table 1: Execution time comparison for analyzing the square plate RCS at twenty frequencies. Times are reported in CPU seconds.

**Conclusions**

Use of MIR allows one to forgo factoring the coefficient matrix at many frequencies when solving a system of linear equations over a band of closely spaced frequencies. The method consists of replacing the frequency-dependent matrix inverse used in the standard iterative refinement algorithm with an approximation that is cheaper to obtain for each analysis frequency. Using this simple algorithm the execution time savings are dramatic for moderately large (more than 1000 unknowns) problems. The algorithm is fully automatic, requiring no user intervention or tuning from problem to problem. Numerical solutions produced by the present algorithm are often more accurate than those of the standard single precision algorithm by virtue of the iterative refinement steps. The present algorithm requires more storage than the standard algorithm since both the previous LU decomposition and the current coefficient matrix must be held in memory simultaneously.

**References**


