A NEW ITERATIVE ALGORITHM FOR IONOSPHERIC TOMOGRAPHY

Eric Sutton and Helen Na

Department of Electrical and Computer Engineering
University of Iowa
Iowa City, Iowa 52242

ABSTRACT

An ionospheric tomography system consists of a satellite and several ground stations located in a line under the path of the satellite. The data collected at the ground stations are the integrals of electron density along many paths between the ground stations and the satellite. From this data an image of electron density in the plane defined by the satellite orbit and the ground stations can be reconstructed using tomographic techniques. However, the data obtained from an ionospheric tomography system is not complete, so a priori information must be used in the reconstruction algorithm in order to obtain a useful solution. The Orthogonal Decomposition Algorithm (ODA) provides a way of incorporating a priori information into the reconstruction. However, due to problems with numerical conditioning, ODA by itself does not yield a solution. This paper presents a new algorithm for ionospheric tomography reconstruction called the Recursive Correction Method (RCM). RCM is a fast, stable, iterative algorithm that takes advantage of the special structure of the ionospheric tomography problem. This paper will also present convergence analysis of RCM, comparison between RCM and Algebraic Reconstruction Technique (ART), and an example using simulated data.

1. INTRODUCTION

Ionospheric tomography is a method of obtaining information about the ionosphere that has been studied extensively in the past few years. The data used for ionospheric tomography consist of measurements of Total Electron Content (TEC) between several ground stations and an orbiting satellite. Since TEC is the integral of electron density, computerized tomography techniques can be used to reconstruct an image of ionospheric electron density in the vertical plane between the ground stations and satellite orbit. Several recent studies have demonstrated the feasibility of this technique [1, 2].

There are several limitations associated with ionospheric tomography systems. Since all data are collected between the satellite orbit and ground, an ionospheric tomography system is approximately analogous to a limited angle tomography system. In particular, ionospheric tomography systems have problems with both limited view angle and missing data. Due to these problems, ionospheric tomography systems exhibit extremely poor vertical resolution [3, 4, 5]. Some form of a priori information must be used in the reconstruction algorithm to obtain a useful reconstruction. There exist several sources of a priori information on the vertical profile of ionospheric electron density, such as ionospheric models and ionosonde measurements.

2. DISCUSSION

2.1. Orthogonal Decomposition

The Orthogonal Decomposition Algorithm (ODA) provides a way of using a priori information in the reconstruction algorithm. In ODA the reconstruction is expressed as a sum of orthonormal basis functions [6]. A priori information is used to construct the basis functions. Thus the reconstruction is limited to a space consisting of reasonable reconstructions.

ODA reduces to the solution of a set of linear equations in the form

\[ Ax = b, \]

where \( A \) is a matrix that depends only on the geometry and the basis functions, \( b \) is a vector of data, and \( x \) is a vector of unknowns. There are generally many more equations than unknowns, so the above equation must be solved in the least squares sense. The poor vertical resolution of the ionospheric tomography system translates into very poor numerical conditioning of the \( A \) matrix. Traditional methods of inversion often either do not produce useful results or involve additional assumptions that adversely affect the reconstruction.

2.2. Separable Basis Functions

A priori information is entered into ODA by choosing the basis functions so that the vertical distribution of the solution is constrained to lie in the space spanned by the vertical distribution of a set of model ionospheres. The model ionospheres should be chosen to span the space of all reasonable vertical distributions for the particular reconstruction problem. This is accomplished by using separable basis functions, where the vertical component is calculated from the a priori information. Each separable basis function can be expressed as

\[ \phi_{ij}(\theta, r) = \xi_i(\theta)\eta_j(r) \]

where \( \phi_{ij}(\theta, r) \) is the two-dimensional basis image, \( \xi_i(\theta) \) is the horizontal basis function, and \( \eta_j(r) \) is the vertical basis function. The set \( \{\phi_{ij}(\theta, r)\} \) is orthonormal if each of the sets \( \{\xi_i(\theta)\} \) and \( \{\eta_j(r)\} \) are orthonormal.
Since all of the data rays pass vertically through the ionosphere, horizontal resolution is generally good, and a \textit{priori} information is needed primarily for the vertical direction. Therefore, basis functions that span the entire space in the horizontal direction may be used. Piece basis functions, Fourier basis functions, and Legendre polynomials are all possible choices. On the other hand, if the vertical basis functions span the entire space in the vertical direction, then the vertical resolution will be very poor. The vertical resolution is improved by choosing vertical basis functions based on a set of model ionospheres [7, 5].

2.3. Residual Correction Method

The Residual Correction Method (RCM) [8] is a numerically stable way of solving for the unknown vector $x$ in equation (1). In RCM the $A$ matrix is partitioned into well-conditioned submatrices which are then inverted separately. The well conditioned submatrices exist due to the geometry of the problem and the way in which the basis functions are constructed. RCM iterates through the submatrices, using the residual vector from the previous iteration as the new data vector. In this way, an approximate solution is first obtained then improved as the iteration continues. In addition, the submatrices can be chosen so that the rate of convergence is accelerated during the first few iterations.

Suppose there are $P$ vertical basis functions, $Q$ horizontal basis functions, and $T$ data points. Then the system matrix $A$ is $T \times PQ$ and contains a row for each data point (TEC value) and a column for each basis function. Let the columns of $A$ be ordered so that all the basis functions containing the first vertical basis function come first, followed by all the basis functions containing the second vertical basis function, etc. The matrix $A$ and vector $x$ can then be partitioned as

$$
\begin{bmatrix}
  A_1 & \cdots & A_P & \cdots & A_P
\end{bmatrix}
\begin{bmatrix}
x_1 \\
\vdots \\
x_P \\
\vdots \\
x_P
\end{bmatrix} = b. \quad (3)
$$

Since each of the submatrices of $A$ contain the columns corresponding to only one vertical basis function, when the submatrices are considered separately the problem reduces from one two-dimensional reconstruction to a set of one-dimensional reconstructions. Each one of the one-dimensional reconstructions is much easier than the complete reconstruction, because each one-dimensional reconstruction solves only for the horizontal component.

Define the generalized inverses of the submatrices as

$$A_p^g = (A_p^T A_p)^{-1} A_p^T. \quad (4)$$

Then the RCM algorithm is initialized using

$$x^{(0)} = 0, \quad (5)$$

$$b^{(0)} = b. \quad (6)$$

The vector $x^{(i)}$ is the initial solution vector, and the vector $b^{(i)}$ is the initial residual vector. The algorithm proceeds by iterating through

$$\Delta x = A_i^{-g} b^{(i-1)}, \quad (7)$$

$$x^{(i)} = x^{(i-1)} + \Delta x, \quad (8)$$

$$b^{(i)} = b^{(i-1)} - A_i \Delta x, \quad (9)$$

where the subscripts on $A$ and $x$ are interpreted modulo $P$. Equation (7) calculates a correction to the solution based on one submatrix of $A$. Equation (8) adds the correction to the solution vector, and equation (9) adjusts the residual vector to reflect the new solution. Each iteration of (7), (8) and (9) will be called a subiteration, and a set of $P$ iterations, where $P$ is the number of vertical basis functions, will be called an iteration. The magnitude of the residual vector $b^{(i)}$ is reduced at each subiteration, and the iteration is halted when the magnitude of $b^{(i)}$ is decreasing at a sufficiently slow rate. RCM converges to a least squares solution, though not necessarily the minimum norm least squares solution.

The RCM algorithm can be written in a more compact form. Equations (7) and (8) can be combined and written in terms of the vector $x$ as follows:

$$x^{(i)} = x^{(i-1)} + \begin{bmatrix} 0_{Q \times T} \\ \vdots \\ A_i^g \\ \vdots \\ 0_{Q \times T} \end{bmatrix} b^{(i-1)}, \quad (10)$$

Then define

$$B_i = \begin{bmatrix} 0_{Q \times T} \\ \vdots \\ A_i^g \\ \vdots \\ 0_{Q \times T} \end{bmatrix}, \quad (11)$$

Also, the residual vector can be written as

$$b^{(i-1)} - b = A x^{(i-1)}, \quad (12)$$

Combining (10), (11), and (12), the complete iteration can be written as

$$x^{(i)} = x^{(i-1)} + B_i (b - A x^{(i-1)}), \quad (13)$$

where the subscript on $B$ is interpreted modulo $P$. The algorithm is initialized using (5) and (6) in the same way as before.

2.4. Convergence

There are two important issues relating to iterative algorithms: (1) Does it converge to a solution? (2) To what solution does it converge? The second issue will be discussed first.
The algorithm cannot cycle through different solutions in the subiterations and arrive back at the same point at the end of an iteration, because different elements of the unknown vector \( x \) are changed in each subiteration. Therefore convergence will be considered in terms of iterations rather than subiterations. Equation (13) shows that if the algorithm converges, then it converges to a point \( x \) where

\[
B_i(b - Ax^{(i-1)}) = 0 \quad \forall 1 \leq i \leq P. 
\] (14)

Equation (14) can be written as

\[
\begin{bmatrix}
(A_i^T A_i)^{-1} & 0 \\
0 & (A_P^T A_P)^{-1}
\end{bmatrix}
\begin{bmatrix}
A_i^T (b - Ax)
\end{bmatrix} = 0. 
\] (15)

Since the block diagonal matrix on the left in (15) is obviously invertible, the algorithm converges to a point where

\[
A_i^T (b - Ax) = 0. 
\] (16)

In other words, the algorithm converges to a least squares solution, though not necessarily the minimum norm least squares solution.

It remains to be shown that RCM actually does converge. The solution to which the algorithm converges satisfies

\[
x = x + B_i(b - Ax). 
\] (17)

If (17) is subtracted from (13), then

\[
x^{(i)} - x = (I - B_i A)(x^{(i-1)} - x) 
\] (18)

Define the error at the \( i \)th subiteration as

\[
e^{(i)} = x^{(i-1)} - x, 
\] (19)

and define the error propagation matrix as

\[
M = \prod_{i=1}^{P} (I - B_i A). 
\] (20)

Then the error propagates according to the equation

\[
e^{(i+P)} = Me^{(i)}, 
\] (21)

If the error propagation matrix \( M \) satisfies the condition

\[
r_s(M) < 1, 
\] (22)

where \( r_s(\cdot) \) denotes the spectral radius, then the algorithm converges. This condition can be checked before the iteration begins.

2.5. Comparison with ART

It is instructive to compare RCM with the Algebraic Reconstruction Method (ART) [9] traditionally used for many tomographic reconstruction applications. Both ART and RCM are iterative algorithms that can be used to solve the matrix equation (1). Both ART and RCM break the problem into smaller, more easily solved pieces to obtain an approximate solution, then iterate to improve the approximation. The difference between ART and RCM is in the way in which the matrix \( A \) is broken into pieces. In the following discussion, the notation (\( \hat{\cdot} \)) will be used for vectors to emphasize the difference between scalars and vectors.

If \( A \) is divided into rows

\[
A = \begin{bmatrix}
\hat{a}_1^T \\
\vdots \\
\hat{a}_N^T
\end{bmatrix}, 
\] (23)

and \( b \) is divided into elements

\[
\bar{b} = \begin{bmatrix}
b_1 & \cdots & b_i & \cdots & b_N
\end{bmatrix}^T, 
\] (24)

then the ART algorithm is obtained. The ART algorithm can be written as

\[
\bar{x}^{(i)} = \bar{x}^{(i-1)} + \frac{(b_i - \hat{a}_i^T \bar{x}^{(i-1)})}{\hat{a}_i^T \hat{a}_i} \hat{a}_i, 
\] (25)

where the \( i \) in \( \hat{a}_i \) and \( \bar{b}_i \) is interpreted mod \( N \).

On the other hand, if \( A \) is divided into columns

\[
A = \begin{bmatrix}
\hat{a}_1 & \cdots & \hat{a}_i & \cdots & \hat{a}_M
\end{bmatrix}^T, 
\] (26)

and \( x \) is divided into elements

\[
\bar{x} = \begin{bmatrix}
x_1 & \cdots & x_i & \cdots & x_M
\end{bmatrix}^T, 
\] (27)

then the RCM algorithm is obtained. This version of RCM is one in which each submatrix of \( A \) is a single column of \( A \). The RCM algorithm can be written as

\[
x_i^{(i)} = x_i^{(i-1)} + \frac{\hat{a}_i^T (\bar{b} - A \hat{a}_i^{(i-1)})}{\hat{a}_i^T \hat{a}_i} \hat{a}_i, 
\] (28)

where the \( i \) in \( \hat{a}_i \) and \( \bar{b}_i \) is interpreted mod \( M \).

Equations 25 and 28 are very similar in form, but quite different in practice. Equation 25 is a vector equation, whereas equation 28 is a scalar equation. On the other hand, the quantity in parenthesis in 25 is a scalar quantity,
whereas the quantity in parenthesis in 28 is a vector quantity. For ART, the previous solution is projected onto a hyperplane defined by a row of $A$. For RCM, the previous residual vector is projected onto a column of $A$. When the system matrix is sparse, ART takes advantage of the sparsity of the system matrix. When the system matrix is not sparse, but contains some column oriented structure, RCM takes advantage of the structure of the system matrix.

3. EXAMPLE

Figure 1 shows a simulated ionosphere from the IRI90 ionospheric model. TEC data was calculated for the image of Figure 1 using seven ground stations and satellite positions in one degree increments. The minimum elevation for any data path was 18 degrees. A priori information was obtained from a large set of IR190 profiles at the same latitude and time, but with different values for the mean solar sunspot number. Figure 2 shows a reconstruction using RCM with 3 vertical basis functions and 11 horizontal basis functions for a total of 33 basis functions. The two peaks on the left are accurately reconstructed; however, for the peak on the right, there is insufficient data coverage to accurately resolve the vertical height of the peak electron density. Figure 3 shows the normalized RMS of the residual vector. Note that the algorithm converges very quickly during the first few iterations.

4. CONCLUSION

Ionospheric tomography systems exhibit very poor vertical resolution. A priori information must be used in the reconstruction process to obtain a useful reconstruction. Even with a priori information, the reconstruction problem is extremely numerically ill-conditioned. RCM is a fast and stable iterative method of solving the reconstruction problem.

The performance of the new algorithm has been demonstrated using simulated data from a test case generated by the IR190 ionospheric model.

Acknowledgments. This material is based upon work supported by the National Science Foundation under grant ATM-9211708 and grant ATM-9419552.

5. REFERENCES