1992

The Visual Diagnosis on the Numerical Calculation of PDE Problems

Yukio Umetani

Report Number: 92-082
THE VISUAL DIAGNOSIS ON THE NUMERICAL CALCULATION OF PDE PROBLEMS

Yukio Umetani

CSD-TR-92-082
October 1992
The Visual Diagnosis of Numerical Calculation for PDE Problems

Yukio Umetani
Hitachi Central Research Laboratory
Computer Science Department
Purdue University

Contents

1 Introduction  3

2 Mathematical Fundamentals  4
   2.1 Discretization  4
   2.2 Linear Mappings  5
   2.3 Eigenvectors and Eigenvalues  6

3 Methodology for Visual Diagnosis  7
   3.1 Methods for Diagnosis  9
   3.2 Methods for Presentation  11

4 Organization of the Experimental Diagnosis System  11
   4.1 Monitoring Routine  13
   4.2 Diagnostic Windows  13

5 Case Study  20

6 The Lessons of the Case Study  37

7 Concluding Remarks  49
Abstract

A systematic method is proposed and tested to diagnose by visual means the causes of symptoms found in the PDE solving process. This method relates the causes of the symptoms to the dominant eigenvectors or the column vectors of the discretization matrix. To aid the visualization of this process contour map representations of vector values are introduced. The results of applying this method to three cases show the feasibility of the method. The efficiency of the eigenvector analysis must be imposed to make this method usable for practical problems.

1 Introduction

The advent of high performance workstations of recent years enables the PDE (Partial Differential Equation) problem solver to have more interactive and human-friendly interfaces compared to the earlier batch/TSS oriented solvers. There are many efforts to establish better man-machine interfaces for pre/post processing, guidance facilities for numerical methods, or problem decomposition for parallel machines ([9], [4], and [1]), by taking advantage of the high performance and enhanced visual facilities of recent workstations. We believe, on the other hand, that the numerical computation process should also be reorganized similarly toward a better balance of intelligence and sensitivity or sensuality. The mass of numbers and mathematical formula should be replaced in order to provide a close interaction between analysts and computational objects. Efforts in this direction seem to be very rare.

Among many subjects in the computation process, we first consider the diagnosis problem in this paper. Current users of PDE solvers often encounter some numerical trouble such as unbelievable or dubious results, local hazards, oscillations, or divergence of solutions. To get reasonable solutions they are forced to do some kind of trouble shooting or verification. The reasons for this trouble are diverse, ranging from the instability of physical phenomena, inadequate mathematical models, contradictory or insufficient boundary/initial conditions, unsuitable discretizations or numerical algorithms, to a trivial input error. Human beings are intelligent but erroneous creatures. They must pay attention to trouble shooting, even though the work here can be reduced by the use of high level solvers.

Many of these troubles may be avoided by enhancing the problem formulation functionality or the guidance facility of solvers, or by reducing the amount of input through automatization. But, no solver can completely and automatically check the dynamic behavior of the problem. This is especially so when the numerical results lie on the border of true or false. Existing solvers cannot provide the means to help the user with this interpretation. If one can provide the user with the information to easily grasp the relation between results and causes, then it will greatly reduce the burden of the user. This is the purpose of this paper.

The remainder of this paper is organized as follows. In Section 2, we introduce the mathematical fundamentals related to this theme. Then the investigation into the methodology of the visual diagnosis is described in Section 3. The organization of an experimental diagnosis system embodying this method is described in Section 4, with a case study of three examples that apply this method to the actual trouble shooting in Section 5. Finally, the review of this method and future research is presented in Section 6.
2 Mathematical Fundamentals

In this chapter, the mathematical issues related to the numerical solution of PDE problems are outlined. For more details, see Section 1 of [7], or Section 5 of [8].

2.1 Discretization

The standard method to compute the numerical solution of a linear PDE boundary value problem is to approximate the problem by a large system of simultaneous linear equations and to solve this system by a numerical method. Nonlinear problems are treated by using convergent iterative schemes to linearized PDEs. The time dependent problems are treated by successively solving the stationary PDE problems for each quantamized time step.

This approximation phase is called discretization. For example, in FDMs (Finite Difference Methods) spacial differential operators are approximated by the difference formulas of values at adjacent quantized space points called grid points, mesh points, or nodes. An example is shown below. Let

\[ Lu = (aU_x)_x + (cU_y)_y + f_u = g \]  

be a PDE on a two dimensional domain. Then \((aU_x)_x\) and \((cU_y)_y\) are approximated by the FDM as

\[
\begin{align*}
(aU_x)_x &= \frac{(a_+ [U_{i+1,j} - U_{i,j}] - a_- [U_{i,j} - U_{i-1,j}])}{h^2} \\
(cU_y)_y &= \frac{(c_+ [U_{i,j+1} - U_{i,j}] - c_- [U_{i,j} - U_{i,j-1}])}{k^2}
\end{align*}
\]

where

\[
\begin{align*}
a_+ &= a(x_i + h/2, y_i), & a_- &= a(x_i - h/2, y_i) \\
c_+ &= c(x_i, y_i + k/2), & c_- &= c(x_i, y_i - k/2)
\end{align*}
\]

and \(U_{i,j}\) is the value of \(U\) at mesh point \((x_i, y_j)\), \(h\) and \(k\) are the mesh spacings in the \(x\) and \(y\) coordinate directions respectively, (this is the case for uniform grids). Then (2.1) is written in the form of finite difference equations

\[ \alpha_0 U_{i,j} + \alpha_1 U_{i+1,j} + \alpha_2 U_{i,j+1} + \alpha_3 U_{i-1,j} + \alpha_4 U_{i,j-1} = g_{i,j} \]  

(2.2)

where

\[
\begin{align*}
\alpha_0 &= f - (a_+ + a_-)/h^2 - (c_+ + c_-)/k^2 \\
\alpha_1 &= a_+ / h^2, \alpha_2 = c_+ / k^2, \alpha_3 = a_- / h^2, \alpha_4 = c_- / k^2
\end{align*}
\]

Similar equations are derived at the boundary mesh points by taking individual boundary conditions into account.

This leads to the following simultaneous equations expressed in matrix/vector form.
The $k$-th row, $l$-th column element $a_{k,l}$ of the coefficient matrix represents the rate of the contribution from the $l$-th entry of $U$ to the $k$-th entry of $g$. The important fact is that all the information of the partial differential operator $L$ in (2.1) is reflected in the coefficient matrix mixed with the information from the grid points. The coefficient matrix is a band matrix, namely the values of elements are zero except for a few diagonal lines. The information of the righthand side of (2.1) is reflected in the constant vector $\{g_{ij}\}$ of (2.3).

In the discretization by the FEM (Finite Element Method), the solution $U$ is approximated by a linear combination of basis functions from a set $\{\phi_j(x, y), \ j = 1, N\}$ as follows

$$U(x, y) = c_1\phi_1(x, y) + \ldots + c_j\phi_j(x, y) + \ldots + c_N\phi_N(x, y)$$  \hfill (2.4)

The differential operator $L$ in (2.1) is applied to each $\phi$. The coefficients $c_j$ are determined by solving the following simultaneous linear equations

$$c_1(L\phi_1, t_i) + \ldots + c_j(L\phi_j, t_i) + \ldots + c_N(L\phi_N, t_i) = (g, t_i) \quad i = 1, N$$  \hfill (2.5)

where $\{t_i, \ i = 1, N\}$ is a set of test functions, and $(f, g)$ is defined as the inner product of two functions, namely the integral of the product of two functions over the whole space domain.

In the Galerkin Finite Element Method, the basis functions are also used as test functions. Basis functions are usually selected to have a local support, namely to take zero values except on a group of adjacent nodes. In this case, the coefficient matrix of (2.5) is also a band matrix.

### 2.2 Linear Mappings

Let $x$ be an $n$-dimensional real valued vector. Then the multiplication by an $n \times n$ real valued matrix $A$ defines a linear mapping on the $n$-dimensional real space. Namely, $A(x + y) = Ax + Ay$ and $A(cs) = cA(x)$ hold. Let $y = Ax$, then $y$ is called the image of $x$ by $A$, while $x = A^{-1}y$ is called the inverse image of $y$, or the source of $A$. Solving simultaneous linear equations $Ax = y$ is the operation to obtain the inverse image of $y$, when $y$ is known.

Let the $j$-th column vector of matrix $A$ be $a_j$ and the $j$-the element of vector $x$ be $t_j$. Then $Ax$ is written from the definition of the matrix-vector multiplication as follows.
\[ Ax = a_1 x_1 + a_2 x_2 + \ldots + a_j x_j + \ldots + a_n x_n \quad (2.6) \]

That is, \( Ax \) is composed of the sum of column vectors multiplied by the corresponding vector element values. It is a simple but very important issue to conceive a vivid image of linear mapping. \( A \) is a banded matrix for the discretized form of a PDE like (2.3). Each \( a_j \) \( x_j \) defines a local mapping, namely it affects only several node values around the \( j \)-th node. Inversely, it means that the value of the \( j \)-th node of vector \( Ax \) is determined by the \( x \) values on the \( j \)-th node plus a few nodes around it, namely where the corresponding column vector of the coefficient matrix has nonzero values on the \( j \)-th element.

### 2.3 Eigenvectors and Eigenvalues

It is known that an \( n \times n \) real valued (or complex valued) matrix \( A \) has up to \( n \) independent complex valued vectors which do not change their direction when multiplied by \( A \).

\[ Ax_j = \lambda_j x_j, \quad j = 1, n \quad (2.7) \]

Here, independent means that one can not be expressed as a linear combination of the others. Each \( x_j \) is called an eigenvector of \( A \), and \( \lambda_j \), the magnification factor by this mapping, is called the eigenvalue. In general, the \( \{ \lambda_j \} \) are complex numbers.

When \( A \) is a symmetric matrix, namely when \( a_{i,j} = a_{j,i} \) for all \( i \) and \( j \), then it has real eigenvectors and eigenvalues. Furthermore, eigenvectors are also orthogonal with respect to each other, namely \( (x_i, x_j) = 0 \) if \( i \neq j \) where \( (x_i, x_j) \) is the vector inner product.

When \( A \) is not symmetric but real valued, it has the following conjugate properties. If a complex vector \( x_r + i x_i \) and a complex number \( r_r + i r_i \) are the eigenvector and its associated eigenvalue, then their conjugate counterparts \( x_r \) and \( x_i \) and \( r_r - i r_i \) are also the eigenvector of \( A \) and the associated eigenvalue. The vectors \( x_r \) and \( x_i \) are not eigenvectors, but they are related to each other by the following relations.

\[ Ax_r = r_r x_r - r_i x_i, \quad Ax_i = r_r x_r + r_i x_i \quad (2.8) \]

Therefore, when \( |r_r| \gg |r_i| \), \( x_r \) and \( x_i \) are almost eigenvectors. When \( |r_i| \ll |r_r| \), they exchange their images by the mapping defined by \( A \).

When \( A \) has \( n \) independent eigenvectors, any vector \( x \) can be expanded by a linear combination of these eigenvectors.

\[ x = c_1 x_1 + \ldots + c_k x_k + \ldots + c_1 x_1 + \ldots + c_n x_n \quad (2.9) \]

The vector of coefficient values \( \{ c_1, c_2, \ldots, c_n \} \) is called the spectrum of \( x \), and (2.9) is called the eigenvector expansion. The eigenvector expansion gives another vivid image of the linear mapping, because each \( x_j \) is an invariant of this mapping (neglecting the magnitude of the vector).

\[ Ax = c_1 Ax_1 + \ldots + c_k Ax_k + \ldots + c_1 Ax_1 + \ldots + c_n Ax_n \]

\[ = c_1 \lambda_1 x_1 + \ldots + c_k \lambda_k x_k + \ldots + c_1 \lambda_1 x_1 + \ldots + c_n \lambda_n x_n \quad (2.10) \]
The linear mapping $A$ enlarges its domain by the eigenvalue times to individual eigenvector directions. If one can obtain the spectrum of a vector, the spectrum of the image vector is easily constructed by multiplying the eigenvalue to the corresponding spectrum values. Inversely, if one gets the spectrum of the image vector, the spectrum of the source vector is easily constructed by dividing each spectrum value by the corresponding eigenvalue.

To avoid the use of complex values in the case of a nonsymmetric matrix, one can modify the eigenvector expansion using the conjugate property. For a real vector, the spectra of the conjugate eigenvector pairs in the expansion (2.9) have conjugate values that cancel their imaginary parts. Let $c_kx_k$ and $c_lx_l$ be such a conjugate pair, namely

$$c_k = c_r + i c_i, \quad x_k = x_r + i x_i$$
$$c_l = c_r - i c_i, \quad x_l = x_r - i x_i$$

Then,

$$c_kx_k + c_lx_l = (c_r + i c_i)(x_r + i x_i) + (c_r - i c_i)(x_r - i x_i)$$
$$= 2c_r x_r - 2c_i x_i$$

Therefore, in the expansion (2.9) we can use the real/imaginary vector pair $(x_r, x_i)$ instead of the conjugate pair $(x_k, x_l)$.

$$x = c_1x_1 + \ldots + c_rx_r + \ldots + c_i x_i + \ldots + c_n x_n \quad (2.11)$$

and

$$Ax = c_1Ax_1 + \ldots + c_rAx_r + \ldots + c_iAx_i + \ldots + c_nAx_n \quad (2.12)$$

We name this new expansion the quasi-eigenvector expansion. In the quasi-eigenvector expansion, all coefficients and vector elements have real values. This allows a more concrete image of the linear mapping. Of course, $x_r$ and $x_i$ are not eigenvectors, but their images are the linear combinations of only $x_r$ and $x_i$ from the relation (2.8). When $A$ is symmetric or all eigenvalues are real, the quasi-eigenvector expansion coincides with the eigenvector expansion.

3 Methodology for Visual Diagnosis

The purpose of visual diagnosis is to find an effective method in locating the causes of symptoms in an intuitively understandable way. The chart in Figure 3.1 is used to identify the parts of the problem more clearly. This chart shows the relations of the information in a PDE solving process. The information at the origin of an arrow determines the information at the destination. As explained in Section 2, the matrix data $(A)$, the source vector $(x)$, and the image vector $(y)$ are the three principle pieces of information in a PDE solving process. They are related to each other by the arrowed lines (1) $(y = Ax)$, and (2) $x = A^{-1}y$. The matrix data is determined by the mathematical model, discretization method, etc., as shown by (2.2) and (2.3). This is represented
by the arrowed line (3). Also one of these vectors, namely the source vector (the righthand side of discretized linear equations) is determined by the mathematical model and the discretization method (the arrowed line \((3)'\)).

Diagnosis means to identify the causes of peculiar symptoms seen in numerical computing processes at the image vector or the source vector, by tracing back the arrowed lines. For example, assume that the current solution, which is a typical source vector, shows certain local irregularities or oscillations. Then the causes should be sought first as a property of the matrix data or the image vector, and then from the mathematical model or the discretization method. Similarly, when divergence occurs, which one meets frequently in a computation process, it is in most cases related to the symptom of the residual vector \(r = b - Ax\) (let \(b\) be a constant vector). The residual vector is a typical example of the image vector. In this case, the causes must be sought from the matrix data or the source vector, then from mathematical models or discretization methods.

"Visual" means to present the above identification process in an intuitively understandable way. It is the technique of visualizing the information and relations in an understandable manner.

Therefore, themes in this area are categorized into two groups. The first category is the method of diagnosis, i.e., how to identify the causes of the symptoms. The second category is the method of presentation. They are related in some part. The author investigates the methods for diagnosis.
3.1 Methods for Diagnosis

The author discusses methods to trace back the four (eventually three) lines in Figure 3.1.

(a) **Image vector ← Matrix × Source vector (line 1)**.

This line is the easiest to attack. The image vector is related to the source vector by formula (2.6) or formulas (2.10), (2.12). In the former case, a matrix is represented by a set of column vectors, while in the latter case by a set of eigenvectors or quasi-eigenvectors. Formula (2.6) is easier to use for the diagnosis of the image vector, especially when the symptom is local, because of the local nature of linear mapping and the connectivity to line (3) which will be discussed later. When the symptom is global, the use of the latter formula is also effective. In that case, the symptom is at once attributed to one of the terms $c_r A x_r$ using formula (2.12), then formula (2.6) is applied to $c_r A x_r$, because $c_r A x_r$ is one kind of image vector. Given $y = A x$, to get each term of the righthand side of formula (2.6) one needs to solve the following simultaneous equations

\[(a_1, a_2, \ldots, a_j, \ldots, a_n) x = y \quad (3.1)\]

or

\[\left((a_1, a_i), (a_2, a_i), \ldots, (a_j, a_i), \ldots (a_n, a_i)\right) x = (y, a_i) \quad i = 1, n \quad (3.2)\]

In general, (3.2) is easier to solve than (3.1).

When the symptom is local, one can use the following local minimization technique. This is more economical than using (3.1), or (3.2). Let $L$ be the set of nodes which belong to the specified local region, and $A^l, a^l_j$ be the matrix or column vector consisting of rows or elements of $A$ or $a_j$ belonging to $L$. Also let $y^l$ be the projection of $y$ to $L$. And consider minimizing the norm of the residual $R^l$ on $L$,

\[R^l = (y^l - A^l x, y^l - A^l x)\]

\[= (y^l, y^l) - 2(y^l, A^l x) + (A^l x, A^l x)\]

\[= (y^l, y^l) - 2(A^{lt} y^l, x) + (A^l x, A^l x)\]

\[(A^{lt} is the transpose of A^l)\]

then

\[\partial R^l / \partial x = -2A^{lt} y^l + 2A^{lt} A^l x\]

Assuming $\partial R^l / \partial x = 0$ at the minimum point, we have

\[A^{lt} A^l x = A^{lt} y^l\]

(3.5)
is obtained. From \( A^i = (a^i_1, a^i_2, \ldots, a^i_j, \ldots, a^i_n), \) (3.5) is written as follows:

\[
((a^i_1, a^i_2), (a^i_2, a^i_1), \ldots, (a^i_j, a^i_i), \ldots, (a^i_n, a^i_i))x = (y^i, a^i_i) \quad i = 1, n
\]  

To solve this system, one needs to eliminate the rows where \( a^i_j = 0 \) as well as the corresponding \( x \) elements and column vectors. Therefore, the computation is much cheaper.

Once the righthand side of (2.6) is obtained, one can continue the diagnosis by examining the dominant terms on the righthand side. If (3.1) cannot be solved, or (3.2), (3.6) has a large residual in the original equation, this means that \( y \) is out of the range of \( Ax \). In this case, the causes must be sought in the other lines, possibly in line \((3)'\). It will be accomplished by examining the residual vector.

Relations involving formulas (2.10) or (2.12) can be treated by a similar method.

\( \text{(b) Source vector } \leftarrow \text{ Matrix}^{-1} \text{ Image vector (line (2))} \)

This route is more difficult to analyze. It is not wise to use the column vector expansion like (2.6), because the column vectors of an inverse matrix has no local properties. The only applicable way is to use the eigenvector expansion (2.9) or the quasi-eigenvector expansion (2.11), by utilizing the property that the (quasi) eigenvector is the (quasi) invariant of a linear mapping. In the case of the quasi-eigenvector expansion, the following relations hold.

\[
x = c_1x_1 + \ldots + c_rx_r + \ldots + c_iX_i + \ldots + c_nX_n
\]

\[= A^{-1}(c_1Ax_1 + \ldots + c_rAx_r + \ldots + c_iAx_i + \ldots + c_nAx_n)\]

In (3.7), \( \{c_j\} \) represents the image vector \( y \), while \( \{x_j, Ax_j\} \) represents the linear mapping and its inverse. The (quasi) eigenvector expansion of the source vector \( x \) is accomplished by the similar way as explained in (a).

Further examination follows in two directions for dominant terms of the righthand side of (3.7), terms which show similar symptoms as \( x \). Let \( c_rAx_j \) be such a term. One direction is to relate \( Ax_j \) to the column vectors of \( A \) by the methods of (a). This means investigating why this eigenvector has such a distribution in relation to the column vector. Another direction is to relate \( Ax_j \) to the symptoms in the image vector \( y \). This means seeking the cause in the image vector itself. One can continue the diagnosis by both or either of these directions.

\( \text{(c) Matrix data, Image vector } \leftarrow \text{ Mathematical model, Discretization method (line (3), (3)')} \)

The diagnosis on this route is intermixed but easy to treat because the local correspondence is maintained. Matrix column data or the constant vector element of a particular node is determined by the model and the discretization information around this node. The best diagnosis is sensitivity analysis, namely, to change source parameters around the particular nodes and to see the effect.
3.2 Methods for Presentation

(a) Presentation of data.

The author chooses the contour map as the presentation form to visualize source vectors and image vectors. Because these data are a collection of the values at each mesh point, it is desirable to present related meshing information. Among the methods of this type, the contour map is most suited for expressing local small disturbances of values which characterize the symptom of the numerical error; it is better than the color map or the dot map. The defect of the contour map is that it can not distinguish a convex or a concave area, it can be supplemented by the color map. The author chooses the same presentation for eigenvectors or quasi-eigenvectors for the same reason.

Matrix data is best presented by a set of column vectors at individual node points; this clarifies the relation to the column vector expansion of the linear transformation. Some iconic presentations or color presentations are more suited than the contour map for each column vector. Each column vector has a small support around individual node points and also need not be as precise as the vector data. One usually needs to know the rate of concentration to the central node or the combinations of arithmetic signs from this data. On the other hand, many data need to be presented simultaneously. The author temporarily adopts the contour map presentation to ease the implementation.

Mathematical models and discretization methods are very difficult to express, except for meshing information. The color map presentation of the contribution of each term of the PDE or the boundary condition to matrix data or the constant vector might be useful in diagnosing lines (3) and (3)'. Examination of this point is left to future work.

(b) Presentation of relations.

The fundamental approach is to overlap presentations. This is effective between any data which have spacial (node) correspondence, for example, between image vector and source vector through matrix data, or between eigenvector and image vector or source vector. If one normalizes the vector before presentation, the vectors with the same direction have the same contour map. The relation (3) or (3)' will be presented by the correspondence of color.

4 Organization of the Experimental Diagnosis System

An experimental diagnosis system has been developed to test the feasibility of the method described above, especially focussing on the diagnosis of line (1) and line (2). Figure 4.1 shows the organization of the diagnosis system.

This system is constructed on the PDE solver PDEQSOL [6] developed at Hitachi Ltd. and postprocessor MGRAF developed at Hitachi Dublin Laboratory. This system consists of a monitoring routine called by the PDEQSOL runtime code and four kinds of diagnostic windows. The monitoring routine gathers data from the process of computations and stores them in data files. The user performs post diagnosis using these data and diagnostic windows. Though interactive diagnosis is
Figure 4.1: Organization of the experimental diagnosis system.
more favorable for this purpose, post diagnosis was adopted from the ease of the implementation. The author explains each part in more detail as follows.

In this system, quasi-eigenvectors are used for non symmetric matrices. Therefore, 'eigenvector' in this and the following chapters also stands for 'quasi-eigenvector'.

4.1 Monitoring Routine

This routine gathers information such as matrix column vectors, constant vectors and solution vectors. It also monitors the iterative convergence process of PDEQSOL's linear equation solver which uses the preconditioned conjugate gradient (PCG) or biconjugate gradient (BPCG) methods [8], [2], and records the values of source vectors and image vectors like intermediate solutions (usually denoted as 'Xn'), residual ('Rn'), relaxation direction ('Pn'), and the image of relaxation direction ('APn') and so on for each iteration step. The image data are stored as the image by the original matrix (not preconditioned) to make the interpretation for diagnosis easy. This routine also calculates and records the eigenvalues and eigenvectors of the original matrix from matrix column data. IMSL's eigensystem analysis routine using the shifted QR algorithm [3], [5] is used for this purpose.

4.2 Diagnostic Windows

Diagnostic windows consist of four related windows, namely, Numerical Process Window (NPW), Matrix/Eigenvector Window (MEW), Spectrum Analysis Window (SAW), and Association Window (AW).

The Numerical Process Window enables users to investigate the iterative convergence process and to examine the related data in visual form. Figure 4.2 shows the whole view of the Numerical Process Window and its control bar. In the NPW, intermediate values of source or image vectors at a specified iteration count is specified by moving the slide bar at the upper right corner of the window. In this example, the values at iteration count 6 are displayed. Data types to be displayed are selected by pressing the button on the control bar. In this example, intermediate solution X (brown line with color annotation), residual R (yellow line), and the image of relaxation direction vector AP (blue line) are displayed. In this display, the silver line shows the mesh decomposition. This window is used to explore particular symptoms in a numerical process.

The Matrix/Eigenvector Window displays the matrix column vectors and/or eigenvectors with its image by the contour map. Figure 4.3 shows an example of MEW, with its control bar. The matrix column number or the eigenvector number to be displayed is specified by the slide bar on the upper right corner. The final solution and the constant vector can also be displayed in this window to make the comparison between these data and eigenvector or matrix column vector easy. In this example, the 33rd eigenvector EVR is displayed with the green line with color annotation, while the 33rd column vector MCOL is displayed with the blue line.

The Spectrum Analysis Window shows the spectrum of a specified vector in graphical form. The example in Figure 4.4 shows the spectrum of intermittent solutions at the 6th iteration count, namely, the brown contour map in Figure 4.2. This window is used to find dominant eigenvectors of the specified vector.
Figure 4.2(a) An Example of Numerical Process Window
Figure 4.3(a) An Example of Matrix/Eigenvector Window
Figure 4.4 An Example of Spectre Analysis Window
Enter Data kind and Data number \( d_{\text{kind}} \): 1:R, 5:X, 8:VR, 14:AVR
8 33
Enter base kind, 0: matrix columns, 1: eigen vectors
0
Enter NNUM
0
NODNUM
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 45 46 47 48 49 50 51 52 53 54 55
ICVEC NC= 55
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 45 46 47 48 49 50 51 52 53 54 55
IRVEC NR= 55
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 45 46 47 48 49 50 51 52 53 54 55

Node/Vector number= 31 Value=0.4985840D+00 Contribution=0.622057D+02
Node/Vector number= 33 Value=0.4968700D+00 Contribution=0.613428D+02
Node/Vector number= 30 Value=0.3076710D+00 Contribution=0.382248D+02
Node/Vector number= 45 Value=0.309880D+00 Contribution=0.380641D+02
Node/Vector number= 34 Value=0.1000000D+01 Contribution=0.259775D+02
Node/Vector number= 44 Value=0.5878350D+00 Contribution=0.143930D+02
Node/Vector number= 46 Value=0.5504420D+01 Contribution=0.579341D+01
Node/Vector number= 55 Value=0.9910910D+01 Contribution=0.274314D+01
Node/Vector number= 29 Value=0.6085590D+00 Contribution=0.220988D+01
Node/Vector number= 50 Value=0.6249950D+00 Contribution=0.205655D+01
Node/Vector number= 52 Value=0.5810470D+00 Contribution=0.170703D+01
Node/Vector number= 54 Value=0.4669100D+00 Contribution=0.155101D+01
Node/Vector number= 37 Value=0.3991560D+00 Contribution=0.118761D+01
Node/Vector number= 53 Value=0.3169010D+00 Contribution=0.112497D+01
Node/Vector number= 7 Value=0.5185480D+00 Contribution=0.111352D+01
Node/Vector number= 21 Value=0.5215600D+00 Contribution=0.106801D+01
Node/Vector number= 32 Value=0.2540490D+00 Contribution=0.932000D+00
Node/Vector number= 51 Value=0.2523140D+00 Contribution=0.794111D+00
Node/Vector number= 4 Value=0.4349130D+00 Contribution=0.695990D+00
Node/Vector number= 43 Value=0.2276330D+00 Contribution=0.686043D+00

Figure 4.5 An Example of Association Window
Finally, the Association Window has similar but more general functionality than the SAW. This window expands any source vector or image vector by a set of eigenvectors or column vectors at the specified domain, by solving the simultaneous equations (3.6). Then it displays the dominant eigenvector numbers or column vector numbers (namely the node numbers) in the order of contribution value. The contribution value is estimated by the Euclidean (L2) norm of each term in the expansion. The example in Figure 4.5 shows the results of the expansion of the 33rd eigenvector (namely the green line in Figure 4.3) by the matrix column vectors in the whole domain. It should be noted that the coefficient of the expansion denoted as ‘Value’ does not always correlate with the contribution value.

The standard usage of these windows is as follows. Assume that one finds one symptom to diagnose in some source vector or image vector by examining the Numerical Process Window. If the symptom is in the source vector or global, even in the image vector, one first finds the candidates of eigenvectors to investigate by using the Spectrum Analysis Window or Association Window. Then one examines each candidate whether it shows similar symptoms by using the Matrix/Eigenvector Window. If one can find a suspicious eigenvector, then again associate it to the dominant matrix column vectors by using the Association Window. Then examine each column vector using the Matrix/Eigenvector Window. When the symptom is associated with an image vector and local, one can directly use the Association Window to get the candidates of column vectors to inspect.

5 Case Study

The author applied this method to the diagnosis of several symptoms. The following describes the process of diagnosis conducted using an experimental diagnosis system.

Case 1: Heat distribution in a locally intensive wind velocity field.

The first case is the diagnosis of a local hazard which appears in the calculation of stationary head distribution in a locally intensive wind velocity field when one uses a relatively coarse mesh. The solution equation is

\[ \Delta U - \rho \nabla \cdot \nabla U = 0 \]  

(5.1)

where

\[ \rho = 1.0, \]

\[ V = (V_x, V_y), \quad V_x = 0.0, \]

\[ V_y = C_1 \exp(-C_2 (x - x_0)^2 - C_2(y - y_0)^2), \]

\[ C_1 = 1000.0, \quad C_2 = 100.0 \]

Thus, the \( y \) component of the velocity field has a sharp peak around the point \((x_0, y_0)\). This point is located at the center of a two dimensional domain. The domain shape is the same as in Figure 4.2. The temperature of the upper boundary is fixed at 100°C and on the right side at 0°C. The adiabatic condition is given to the other boundaries. The mesh decomposition is the same as

20
in Figure 4.2, and the Galerkin Finite Element Method with linear basis functions is used for the
discretization. The Preconditioned Biconjugate Gradient Method is used to solve the discretized
simultaneous linear equations.

The solution of the numerical computation is shown in Figure 5.1. The notable symptom is
that the solution takes the maximum and minimum values at the right and left sides of the peak
velocity point \((x_0, y_0)\). This looks strange for the given wind direction. Is this really right?

Because this is a symptom of the source vector, diagnosis begins by getting the spectrum from
the Spectrum Analysis Window. The result is the same as shown in Figure 4.4. The dominant
eigenvectors are 1 through 11, 17, 19, 25, 26. Among them, the 6th and 26th eigenvectors turn out
to show similarities to the symptom as seen in Figure 5.2. In this figure, the brown line shows the
source eigenvector, while the blue line shows its image by \(A\).

We choose the 6th quasi-eigenvector for further diagnosis. Because the constant vector in the
matrix/eigenvector window shows no resemblance to the image of the eigenvector as seen in Figure
5.3, the investigation must be made on the matrix property. For that purpose, the 6th eigenvector
is evaluated by the matrix column vectors using the Association Window. This is the same as
expanding the length image of the eigenvector in the matrix column vectors. Figure 5.4 is the
result of the evaluation. It becomes clear from the contribution value that the first 6 columns
are dominant. The profile of each matrix column is as shown in Figure 5.5. The blue line is the
contours of the matrix column vector, and the round mark is the position of the center node. One
can immediately decide from these profiles that none of these nodes are properly discretized. A
good discretization should show the peak absolute value on the center node as is typically seen in
Figure 5.6.

To investigate in more detail, let us evaluate the 6th quasi-eigenvector on the single node point
34. The eigenvector has the positive peak value on node point 34. Figure 5.7 shows the result. The
matrix columns 33 and 31 are dominant. Looking at the profiles of both vectors and the coefficient
values in Figure 5.5 (b) and (d), one can find that the eigenvector value at node point 34 ("+"
mark) is accidently kept positive as a result of the balance of peak values of both column vectors
on node 46 (double round mark).

In fact, this solution is very unstable. If one moves the point of the peak wind from \((2.5, 3.0)\)
to \((2.5, 3.1)\), the position of the positive peak value and the negative peak value is reversed as
seen in Figure 5.8. In reality, such a phenomena will not happen under the homogeneous boundary
conditions like this case. This is a false solution due to the improper discretization. To get a proper
solution, one needs to use finer meshes around the peak wind point so that the whole column vectors
have reasonable profiles.

Case 2: Material diffusion problem.

The next case is a transient material diffusion problem within a slender two dimensional region
as shown in Figure 5.9. The source of the material is placed on the ground near the inlet (left
side). The wind is blowing from the left to the right at a fixed velocity which depends only on the
height. The wind blows stronger as it gets nearer the ground. The density of the material is set
to zero at the inlet and ceiling. The Neumann condition is given at the ground and outlet. The
initial distribution is zero in all regions.
Figure 5.1 The Solution of Heat Distribution in a Locally Intensive Velocity Field
Figure 5.2(a) A Dominant Eigenvector of Solution (6th Eigenvector)
Figure 5.2(b) A Dominant Eigenvector of Solution (26th Eigenvector)
Figure 5.3 The Constant Vector for Case 1
pamela 48% dlcor
Enter Data kind, and Data number. dkind 1:r.5:x.8:vr.14:avr 8 6
Enter base kind. 0:matrix columns, 1:eigen vectors 0
Enter NNUN 0

NODNUM
1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22
23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42
43  44  45  46  47  48  49  50  51  52  53  54  55

ICVEC NC= 55
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22
23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42
43  44  45  46  47  48  49  50  51  52  53  54  55

IRVEC NR= 55
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22
23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42
43  44  45  46  47  48  49  50  51  52  53  54  55

Node/Vector number= 45 Value=-.564618D+00 Contribution=0.693306D+02
Node/Vector number= 33 Value=0.521065D+00 Contribution=0.643299D+02
Node/Vector number= 30 Value=-.497056D+00 Contribution=0.617539D+02
Node/Vector number= 31 Value=0.455216D+00 Contribution=0.567949D+02
Node/Vector number= 34 Value=0.100000D+01 Contribution=0.259775D+02
Node/Vector number= 46 Value=-.100847D+01 Contribution=0.246921D+02
Node/Vector number= 53 Value=0.356953D+01 Contribution=0.375694D+01
Node/Vector number= 50 Value=0.132318D+00 Contribution=0.102769D+01
Node/Vector number= 6 Value=-.334644D+00 Contribution=0.718755D+00
Node/Vector number= 7 Value=-.293661D+00 Contribution=0.630603D+00
Node/Vector number= 54 Value=0.166985D+00 Contribution=0.554701D+00
Node/Vector number= 32 Value=-.140537D+00 Contribution=0.515904D+00
Node/Vector number= 51 Value=-.140537D+00 Contribution=0.515904D+00
Node/Vector number= 20 Value=0.211072D+00 Contribution=0.462781D+00
Node/Vector number= 47 Value=-.912301D-01 Contribution=0.287635D+00
Node/Vector number= 43 Value=0.861528D-01 Contribution=0.259640D+00
Node/Vector number= 37 Value=-.845456D-01 Contribution=0.251549D+00
Node/Vector number= 21 Value=0.112799D+00 Contribution=0.235305D+00

Figure 5.4 Dominant Column Vectors of 6-th Eigenvector
Figure 5.5(a) Profile of 45nd Column Vector
Figure 5.5(b) Profile of 33rd Column Vector
Figure 5.5(c) Profile of 30th Column Vector
Figure 5.5(f) Profile of 44th Column Vector
Figure 5.6 Good discretization shows the peak value at the center node well.
Enter Data kind, and Data number. dkind 1: r, 5: x, 8: vr, 14: avr
8 6
Enter base kind. 0: matrix columns, 1: eigen vectors
0
Enter NNUM
1
Enter node numbers (up to 1)
34
NODNUM
34
ICVEC NC= 8
20 21 31 33 34 46 50 53
IRVEC NR= 1
34
Node/Vector number= 33 Value= 0.521065D+00 Contribution= 0.182767D+02
Node/Vector number= 31 Value= 0.455216D+00 Contribution= 0.154536D+02
Node/Vector number= 34 Value= 0.100000D+01 Contribution= 0.370450D+01
Node/Vector number= 53 Value= 0.431160D+00 Contribution= 0.472103D+00
Node/Vector number= 50 Value= 0.312318D+00 Contribution= 0.157625D+00
Node/Vector number= 20 Value= 0.211072D+00 Contribution= 0.101102D+00
Node/Vector number= 46 Value= -0.356953D-01 Contribution= 0.223194D-01
Node/Vector number= 21 Value= 0.112799D-00 Contribution= 0.143490D-01

Figure 5.7 Dominant Column Vectors of 6-th Eigenvector at Nodepoint 34
Figure 5.8 Change of nodal point from (2.5,3.0) to (2.5,3.1) reverses positions of peak positive and negative values.
Let $DD$ be the material density. The governing diffusion equation is as follows.

$$\rho(\partial DD/\partial t + V \cdot \text{grad} DD) = k \Delta DD + Q$$

(5.2)

where

$$\rho = 1.2, \quad k = 1.5,$$

$$V = (U_x, U_y), \quad U_x = VM(1 - Y^2), \quad U_y = 0.0, \quad VM = 10.0,$$

$$Q = Q0(ep^2 + (X - 1)^2 + Y^2), \quad Q0 = 10.0, \quad ep = 0.01$$

The local fine mesh near the source of material is used as seen in Figure 5.9. The discretization is the Galerkin Finite Element Method with piecewise linear basis functions.

The following semi-implicit scheme is used for the time marching to keep the discretized matrix symmetric.

$$DD = D0 + \text{dlt}(k \Delta DD + Q)/\rho - \text{dlt}V \cdot \text{grad}D0$$

(5.3)

$D0$ stands for the distribution at one time step behind, $\text{dlt} = 0.1$ is used. The Preconditioned Conjugate Gradient (PCG) method is used to solve the discretized linear equations.

The computational results are shown in the time sequence in Figure 5.9. The interesting observation is that it shows separation of the mass from the material source. It looks like quite a natural phenomena. One can not judge whether this actually occurs or not using common sense. What is the truth?

We examine the result of time step 4 which shows the first sign of separation. Because this symptom lies in the source vector again, one seeks the dominant eigenvectors by consulting the Spectral Analysis Window or the Association Window. The answer by the Association Window is shown in Figure 5.10. By comparing each of the candidates with the solution at time step 4 at the matrix/eigenvector window, one can get eigenvectors 4 and 7 as the vehicles for further investigation. Especially, the 7th eigenvector seems to be a good candidate. In Figure 5.11, the contour map of this eigenvector is displayed by the blue line in comparison with the solution denoted by the brown line. This eigenvector fits well to the small dent in the solution map which gives rise to the separation of the mass.

The next step is twofold. One is to investigate the dominant column vectors of the matrix which determine the shape of this eigenvector. In this case, one can get no information from this search because all the column vectors are reasonable in the sense explained in Case 1. Another direction is to study the relation to the constant vector. In Figure 5.11, the constant vector is also displayed by a yellow line. Obviously, the constant vector and the 7th eigenvector have some similarity. In fact, one can see that the 7th eigenvector is the most dominant one of the constant vectors by consulting the Association Window (Figure 5.12).

Then the question is why the constant vector takes this shape? A new search begins on line (3)'. Unfortunately, the current diagnostic system does not provide any help for diagnosis on this line. One can identify the reason by analyzing formula (5.3). Equation (5.3) is rewritten as follows.
The constant vector consists of the terms on the right-hand side of (5.4). Among these terms, both $D_0$ and $Q$ have monotone distributions around the material source. Therefore, only the last term is the cause of the dent in the constant vector. In fact, the last term has the distribution as shown in Figure 5.13 by the light blue line.

The cause of this symptom turns out to lie in the explicit treatment of the velocity term. To avoid this error, one needs to use a full-implicit scheme.

$$DD = D_0 + \frac{dlt(kDD + Q)}{\rho} - dlt \cdot \nabla DD$$  \hspace{1cm} (5.5)

or to use the smaller $dlt$ value in a semi-explicit schemes.

The computation results using the fully-implicit scheme at time step 5 is shown in Figure 5.14. The solution, constant vector, and 7th eigenvector at time step 4 is shown in Figure 5.15. Comparison of this figure and Figure 5.11 tells the whole story.

**Case 3: Unnatural boundary condition.**

Let us remove the wind velocity field from the model of Case 1, and change the Neumann boundary condition on both sides to the following mixed boundary condition.

$$\nabla U = C_2 U$$ \hspace{1cm} (5.6)

When $C_2$ is negative, this is a natural model of heat radiation. But if $C_2$ is positive, it is unnatural because it means the boundary of high temperatures get more heat flux than the boundary of low temperatures. One often makes this mistake by forgetting that $\nabla$ means the derivative to the outward direction from the target region.

Figure 5.16 shows the computational solution of this case. At a glance, one can see something is wrong. The dominant eigenvectors of this solution are the 1st (Figure 5.17), and the 4th (Figure 5.18). The eigenvectors are numbered in increasing order of absolute value of the corresponding eigenvalues. The constant vector is the same as Case 1. It shows no symptoms. Therefore, the matrix values must be investigated. The dominant column vectors of these eigenvectors lie on the boundary nodes. Figure 5.19 shows one example of this. It has an unsound shape. The center node does not have a peak value.

The following step is the investigation of line (3). The boundary condition must be investigated as well as the discretization method for the boundary nodes. Then the cause will be identified soon.

6 The Lessons of the Case Study

1. *Is the eigenvector analysis inevitable?*

   The most important lesson of this case study is that the eigenvector analysis takes a long time when the number of nodes exceeds two hundred, even using today's powerful workstations. It is obvious that this point becomes the biggest obstacle to the practical use. We examine whether this analysis is truely necessary or not.

   As stated in Section 3, eigenvectors are introduced as the theoretical basis for the diagnosis of the source vector. But in a real diagnosis procedure, when the constant vector shows no
Figure 5.9 Temporal Changes of Material Distribution in Case 2
<table>
<thead>
<tr>
<th>Node/Vector number</th>
<th>Value</th>
<th>Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.302103D+00</td>
<td>0.230704D+01</td>
</tr>
<tr>
<td>2</td>
<td>0.134930D+00</td>
<td>0.134930D+00</td>
</tr>
<tr>
<td>3</td>
<td>0.157907D+00</td>
<td>0.157907D+00</td>
</tr>
<tr>
<td>4</td>
<td>0.137999D+00</td>
<td>0.137999D+00</td>
</tr>
<tr>
<td>5</td>
<td>0.748850D+01</td>
<td>0.748850D+01</td>
</tr>
<tr>
<td>6</td>
<td>0.622627D+01</td>
<td>0.622627D+01</td>
</tr>
<tr>
<td>7</td>
<td>0.603745D+01</td>
<td>0.603745D+01</td>
</tr>
<tr>
<td>8</td>
<td>0.372286D+01</td>
<td>0.372286D+01</td>
</tr>
<tr>
<td>9</td>
<td>0.307403D+01</td>
<td>0.307403D+01</td>
</tr>
<tr>
<td>10</td>
<td>0.207606D+01</td>
<td>0.207606D+01</td>
</tr>
<tr>
<td>11</td>
<td>0.178075D+00</td>
<td>0.178075D+00</td>
</tr>
<tr>
<td>12</td>
<td>0.177990D+00</td>
<td>0.177990D+00</td>
</tr>
<tr>
<td>13</td>
<td>0.748680D+01</td>
<td>0.748680D+01</td>
</tr>
<tr>
<td>14</td>
<td>0.603745D+01</td>
<td>0.603745D+01</td>
</tr>
<tr>
<td>15</td>
<td>0.589099D+00</td>
<td>0.589099D+00</td>
</tr>
<tr>
<td>16</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>17</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>18</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>19</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>20</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>21</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>22</td>
<td>0.581165D+01</td>
<td>0.581165D+01</td>
</tr>
<tr>
<td>23</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>24</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>25</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>26</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>27</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>28</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>29</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>30</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>31</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>32</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>33</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>34</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>35</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>36</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>37</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>38</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>39</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>40</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>41</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>42</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>43</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>44</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
<tr>
<td>45</td>
<td>0.490852D+02</td>
<td>0.256117D+01</td>
</tr>
</tbody>
</table>

Figure 5.10 Dominant Eigenvectors of the solution at Timestep 4
Figure 5.11 Solution, Constant Vector, and 7th Eigenvector at Timestep 4
<table>
<thead>
<tr>
<th>Node/Vector number</th>
<th>Value</th>
<th>Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>0.137999D+00</td>
<td>0.29392BD-01</td>
</tr>
<tr>
<td>15</td>
<td>0.622629D-01</td>
<td>0.25390BD-01</td>
</tr>
<tr>
<td>4</td>
<td>0.157907D+00</td>
<td>0.25289BD-01</td>
</tr>
<tr>
<td>9</td>
<td>0.748849D-01</td>
<td>0.23993BD-01</td>
</tr>
<tr>
<td>11</td>
<td>0.603746D-01</td>
<td>0.18036BD-01</td>
</tr>
<tr>
<td>2</td>
<td>0.134930D+00</td>
<td>0.15440BD-01</td>
</tr>
<tr>
<td>17</td>
<td>0.307404D-01</td>
<td>0.13969BD-01</td>
</tr>
<tr>
<td>14</td>
<td>0.207606D-01</td>
<td>0.10120BD-01</td>
</tr>
<tr>
<td>18</td>
<td>0.142627D-01</td>
<td>0.80244BD-02</td>
</tr>
<tr>
<td>1</td>
<td>0.302102D+00</td>
<td>0.70953BD-02</td>
</tr>
<tr>
<td>13</td>
<td>0.161655D-01</td>
<td>0.60548BD-02</td>
</tr>
<tr>
<td>51</td>
<td>0.370300D-02</td>
<td>0.56360BD-02</td>
</tr>
<tr>
<td>34</td>
<td>0.106701D-01</td>
<td>0.53464BD-02</td>
</tr>
<tr>
<td>8</td>
<td>0.177831D-01</td>
<td>0.49504BD-02</td>
</tr>
<tr>
<td>22</td>
<td>0.105621D-01</td>
<td>0.45299BD-02</td>
</tr>
<tr>
<td>66</td>
<td>0.212762D-02</td>
<td>0.41684BD-02</td>
</tr>
<tr>
<td>104</td>
<td>0.142875D-02</td>
<td>0.40995BD-02</td>
</tr>
<tr>
<td>52</td>
<td>0.393120D-02</td>
<td>0.40533BD-02</td>
</tr>
<tr>
<td>87</td>
<td>0.219108D-02</td>
<td>0.39802BD-02</td>
</tr>
<tr>
<td>106</td>
<td>0.293683D-02</td>
<td>0.39138BD-02</td>
</tr>
</tbody>
</table>

Figure 5.12 Dominant Eigenvectors of constant Vector
Figure 5.13 The Profile of the Last Term of (5.4) (Blue Line)
Figure 5.14 Material Distribution at Timestep 5 by Full Implicit Scheme
Figure 5.15 Solution, Constant Vector, and 7th Eigenvector at Time Step 4 by Full Implicit Scheme
Figure 5.16 Solution for Case 3
Figure 5.17 A Dominant Eigenvector of Solution (1st Eigenvector)
Figure 5.18 A Dominant Eigenvector of Solution (4th Eigenvector)
Figure 5.19 An Example of Dominant Column Vectors of 1st Eigenvector
It is obvious that this point becomes the biggest obstacle to the practical use. We examine whether this analysis is truly necessary or not.

As stated in Section 3, eigenvectors are introduced as the theoretical basis for the diagnosis of the source vector. But in a real diagnosis procedure, when the constant vector shows no sign of a symptom, the search is redirected to the dominant column vectors of the dominant eigenvector. How about directly evaluating the source vector by column vectors in the specific region showing symptoms and bypassing the eigenvector expansion of the source vector?

The result for Case 1 is the order of 45, 30, 33, 31, 44, 34 by two nodes matching at 34 and 44 where the solution takes the maximum and minimum value. The comparison with Figure 5.4 has a good coincidence of results though the order is slightly different. The experiment in Case 3 also locates most of the boundary nodes. It seems to work well in these cases.

This can be interpreted as follows. Because the symptom of the source vector is the result of the compensating action to the defects in related column vectors, one can find the causes near the symptom.

Then how about the case when the cause lies in the constant vector, or on both of the column vectors and the constant vectors. In this case, the difficulty lies in how to discover the symptom there corresponding to the symptom in the source. The eigenvector plays a good role in this case because of its invariant nature to linear mapping. When the symptom of the source is local, one can count on the local property of linear mapping. One will be able to find the symptom of the image vector in the same area as the source vector. But when the symptom is global, we have no other means except for the eigenvector.

2. Economic computation of eigenvectors.

It is known from the experience that most of the dominant eigenvectors of the solution of a PDE problems or its image vector have low frequency (small absolute eigenvalue). Therefore, one need not calculate all the eigenvectors for the current purposes. To get a small number of eigenvectors, one can use more efficient methods than the $QR$ algorithm like the shifted inverse iteration with Householder transformations [3], or the Lanczos method. Then the application to the large scale problems of practical interest will be reasonable.

3. Other lessons.

The presentation of column vectors by the contour map is very useful in judging the soundness of the discretization. But, there remains the need to grasp the relation among different column vectors. For this purpose, simultaneous presentations of multiple column vectors is necessary. One needs to design a compact form of presentations. If all column vectors can be displayed at once, it will be very helpful for the diagnosis.

The function of the Association Window has proved to be useful. But, it also suffers from low response time if there is a large number of nodes. One needs to explore more efficient methods for association. Because restrictions of association domains are effective to shorten the response time, some interactive means to specify the association domain are needed.
remains to improve the efficiency as stated above and to make this method applicable to the large scale computations of practical use. For practical purposes, the appropriate method for presentation in 3D problems is also a big challenge as well as the efficiency problem. Also, further investigation is needed for the diagnosis of routes (3) and (3)', namely, for the path which relates matrix or vector properties to mathematical models and discretization methods.

The diagnostic process must be closely integrated with the computation process so that one can shift to the diagnosis mode at any time a peculiar symptom appears. This ability will be indispensable for nonlinear problems or time dependent problems where the properties of matrices change dynamically. The knowledge obtained by this method should be utilized for preventive care or the guidance facility of expert systems, or in some cases for the automated function of advanced solvers. In conclusion, this approach should be one of several different methods used to maintain the correctness of computations.

Acknowledgement

The author expresses his gratitude to Professor John R. Rice, Purdue University, and Dr. Hishashi Horikoshi, the former director of Hitachi Central Research Laboratory, for providing the author the chance to conduct this research. Also the author expresses his gratitude to Dr. Mo Mu, Purdue University, for his careful proof reading of this manuscript.

References

