3 Field Fitting

In this chapter we consider the problem of optimally fitting a piecewise functional model to a set of observations or samples taken from a field.

A survey of numerical methods for fitting functions to sets of scattered field samples is given by Schumaker (1976). He considered a variety of both global and local methods for interpolating and approximating or smoothing the sampled values. De Boor (1978) provides a good introduction to the problem of least squares fitting one dimensional splines to scattered data.

We firstly define the structure of our model. Our scope is restricted to fields approximated by linear piecewise functions of the type covered in the previous chapter. In the following section we define the error function or norm which provides us with our measure of goodness of fit. It is this function which we shall be attempting to minimise in our field approximations. We next consider how one assigns an ensemble position to each observation. A single ensemble position must be associated with each observation if the output errors, and hence the error function, are to be uniquely defined. The problem is complicated by the fact that the ensemble may not be a complete representation of the body from which the observations were made. In this case some of the observations may be from outside the region represented by the ensemble. In the next sections we consider the problem of determining optimal estimates of the ensemble field parameters to the set of field observations. Where we are fitting nongeometric fields this estimate will result in a set of linear equations in the ensemble field parameters. In the case of geometric field fits the ensemble position associated with each observation will change as the ensemble configuration is altered. We thus lose the linearity property of the linear field fits and must resort to nonlinear optimisation methods to find the ensemble configuration which minimises the error function.

3.1 Model Structure

A model consists of three components representing the kinds of knowledge we have about a process viz. the structure, the parameters, and the output. The model structure reflects the *a priori* knowledge we have of the process structure and is expressed in terms of interactions between the model components. The model parameters are those variables which are independent of the input and output while the model output is the state of the dependent variables at a given input.

We are concerned here with the estimation of parameters for models expressed as piecewise functions. It is convenient to use a piecewise function structure to model a given field over a region where the field forms a component of a model being analysed by the finite element method. In such cases there is frequently a need to define fields of independent variables for the model, such as constitutive fields, or to specify dependent fields in initial value problems.

Having defined the class of structures that we are interested in we next consider the order of the structure. The order is determined by the type and density of elements comprising the ensemble. The type of element depends upon the degree of the basis functions within the element as well as the degree of field continuity between the elements. Where the element basis functions are polynomials the degree of interpolation is identified with the order of the polynomial. The degree of continuity required between contiguous elements is determined primarily by the differentiability requirements of the differential operators acting on the field. Generally fields must be of class C^0 and sometimes C^1 (a field of class C^c is c-continuously differentiable over the region on which it is defined). The density of elements is simply the number of elements within a given region.

It should be noted that the accuracy of a field model may be improved by increasing the order of the model structure. This may be done locally or globally. Increasing the model order can be achieved by increasing the degree of interpolation within the elements or by increasing the density of the elements. In finite element applications these methods are known as *p*-version (Babuska, Szabo and Katz 1981) (Zienkiewicz and Morgan 1983) and *h*-version (Oden 1972) (Babuska and Aziz 1972) refinement schemes respectively. Both schemes result in an effective increase in the density of ensemble field parameters.

3.2 Error Function

For the estimation of ensemble parameters to model a given field we require information concerning the field on which to base our estimates. In the following we shall consider the cases where this information takes the form of sampled observations of the field values and derivatives.

Since the ensemble parameters are unable to be determined directly we must estimate them from the information we have about the field. For this we require some measure of correspondence between the field and its model. Generally this measure takes the form of a single scalar error function, **E**. The error function is defined as a function of the output errors and depends upon *a priori* knowledge of both the field and the noise component of the observations.

The error function provides a global measure of correspondence between the output and the parameters of the field and its model. We define the output error, $\mathbf{e}^{O}$, for observation O as the difference between the sampled value of the field, $\mathbf{u}^{O}$, and the value of its model, $\mathbf{U}(\mathbf{X}^{O})$, at a given ensemble position, $\mathbf{X}^{O}$:

$$\mathbf{e}^{O} = \mathbf{U}(\mathbf{X}^{O}) \; - \; \mathbf{u}^{O}$$

Any point on the ensemble, **X**, has a unique element, ε, and element position, **x**, associated with it. Letting observation O have a corresponding element identifier o(ε) we can express the observation error in terms of its associated element and element coordinates:

$$\mathbf{e}^{o(\varepsilon)} = \mathbf{u}^{(\varepsilon)}(\mathbf{x}^{o}) \; - \; \mathbf{u}^{o(\varepsilon)}$$

Since the element basis functions at any given point within the ensemble vary linearly with the ensemble field parameters the field value, and thus the output error, at that point is a linear function of the ensemble parameters.

Considerable computational advantage can be gained if we demand that the estimator be linear in the parameters. With this condition the error function must be quadratic in the parameters. Where the output errors are linear in the parameters the error function will be quadratic in the errors. The error function can thus be expressed as:

$$\mathbf{E} = \mathrm{E}_{o(\varepsilon)\, o'(\varepsilon')} \; \mathbf{e}^{o(\varepsilon)} \; \mathbf{e}^{o'(\varepsilon')}$$

where $E_{o(\varepsilon)\ o'(\varepsilon')}$ is a symmetric positive definite weighting matrix acting upon the output errors, $\mathbf{e}^{o(\varepsilon)}$. The proviso that the weighting matrix be positive definite ensures that the error function has a unique minimum.

The choice of $E_{o(\varepsilon)\ o'(\varepsilon')}$ determines the fitting scheme and depends upon the *a priori* knowledge we have about the noise component of the observations.

Where the weighting matrix is equal to the inverse of the noise covariance matrix we obtain the generalised least squares or Markov estimate. This estimate is appropriate where the observations are not statistically independent. Clearly the noise covariance matrix must be known *a priori.*

Where the observations are statistically independent the noise covariance matrix will be diagonal with entries equal to the inverse of the variance of the corresponding observation. This weighted least squares estimate is appropriate where there is *a priori* knowledge of the noise component in each observation.

In the case where one has no knowledge of the noise component in each observation it is appropriate to set $E_{o(\varepsilon)\ o'(\varepsilon')}$ equal to the identity matrix. This corresponds to the normal least squares estimate.

We shall use the error function defined above in both the linear and nonlinear field estimates covered later in this chapter.

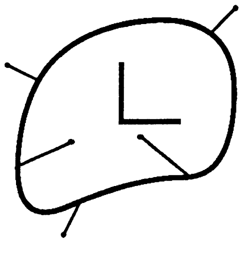
3.3 Ensemble Coordinates Of Observations

In the previous section we defined the output error, $\mathbf{e}_{o(\varepsilon)}$, as the difference between the sampled value of the field, $\mathbf{u}_{o(\varepsilon)}$, and the value of the fitted field, $\mathbf{u}_{(\varepsilon)}(\mathbf{x}_o)$, at a given element position, $\mathbf{x}_o$:

$$\mathbf{e}_{o(\varepsilon)} = \mathbf{u}_{(\varepsilon)}(\mathbf{x}_o) - \mathbf{u}_{o(\varepsilon)}$$

For the fitting problems to be well posed we require that there exists a unique ensemble position associated with each observation. This section considers the problem of how one determines the mapping between an observation and its corresponding point on the ensemble.

It should be noted that an observation need not necessarily lie within the region defined by the ensemble. Such situations may arise where the ensemble does not represent an exact geometric model of the actual region from which the observations were made. In this case we must define an equivalence class, mapping arbitrary sets of points to unique points within the ensemble. Such mappings are usually not hard to define. An obvious choice is to employ the global coordinate system using one or more of the coordinate directions to define the equivalence mapping. For example if the ensemble has been defined in polar coordinates, and there is no folding back of the ensemble in the angular direction, then the angular coordinate may be used to map arbitrary points to unique points within the ensemble. In this case the radial coordinate lines define the equivalence mapping:



While it may be possible to define an appropriate equivalence class such mappings are of limited practical use unless one can identify each equivalence class with its corresponding point on the ensemble. Since the ensemble configuration is usually defined parametrically the mapping from arbitrary points into the ensemble is an implicit operation. Unless the association of each equivalence class to its corresponding element and element coordinates can be determined analytically we must use iterative methods to find this mapping.

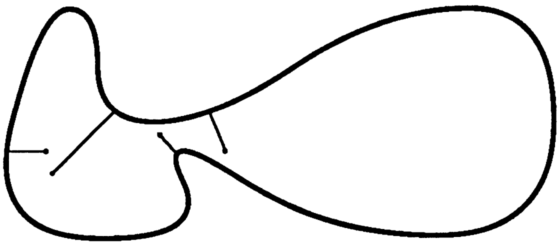
Since iterative methods must in general be used to determine the ensemble position associated with each observation it is appropriate that we should investigate a more rational method of defining our equivalence mapping. An obvious choice is the ensemble position which minimises the distance between the ensemble and the observation. As this minimisation is generally of low order (equal to the dimension of the element) and nonlinear a Newton-like method is appropriate. In the cases with which we have dealt we have used a Newton algorithm with checks to ensure that the matrix of second derivatives or Hessian is positive definite, a limit on the step length, and a quadratic line search at each iteration. This scheme has proved to be reasonably robust and efficient. Details of the algorithm used to determine the closest approach of the ensemble to an observation are given in the following subsection.

While it is relatively costly to find the ensemble position of an individual observation it should be noted that, since the ensemble position of an observation is independent of all others, the work performed in determining the closest approach for different observations may be performed in parallel.

Note that the minimum point need not be located within the element with which the observation was originally associated. If the coordinates of the minimum correspond to a point within a contiguous element the observation becomes associated with the new element and a further minimisation must be carried out with respect to the parameters of that element. If the coordinates of the minimum correspond to a point outside the ensemble the observation continues to be associated with the original element and further searching ceases.

Notice that we require the line to be a local minimum rather than a global minimum. This feature is desirable in the case of observations from distinct neighbourhoods of the actual region which are nevertheless in close physical proximity. We wish to ensure that the ensemble positions of the observations are correspondingly distinct so that their contributions are allocated appropriately.

Conversely, we may find that observations are in fact trapped by a local minimum. Trapped observations may arise where poor initial estimates of the ensemble positions are supplied. They may also occur in cases of geometric fitting where a local minimum is created between the ensemble position at the previous iteration and the desired coordinates at the current ensemble configuration. Such aberrations are difficult to detect automatically. In these cases it is best to assess the adequacy of the observation coordinates visually.



3.3.1 Closest Approach Of An Element To An Observation

Let λ be a line connecting a point at which an observation is made to a point within an element ε. The energy of λ is given by:

$e = \Sigma_n (z^{no(\varepsilon)} - z^{n(\varepsilon)})^2$

where $z^{no(\varepsilon)}$ are the rectangular cartesian coordinates of observation o and $z^{n(\varepsilon)}$ are the rectangular cartesian coordinates of a point on element ε. Now

$y^{m(\varepsilon)}(\mathbf{x}) = \psi_{\beta(\varepsilon)}(\mathbf{x})\ \upsilon^{\beta(\varepsilon)}{}_B\ Y^{mB}$

are the curvilinear coordinates of a point in element ε with element coordinates $\mathbf{x}$. The y^m are related to z^n by a coordinate coordinate transformation, $y^m = y^m(\mathbf{z})$. This transformation is invertible almost everywhere, $z^n = z^n(\mathbf{y})$.

We wish to minimise e with respect to $\mathbf{x}$. Since e is, in general, a nonlinear function of $\mathbf{x}$ we use a Newton-Raphson iteration to solve for the minimum:

i.e. $\partial^2 e/\partial x^l \partial x^{l'}\ \Delta x^l = -\partial e/\partial x^{l'}$

Now

$\partial e/\partial x^l = \partial e/\partial z^n\ \ \partial z^n/\partial y^m\ \ \partial y^m/\partial x^l$

where

$\partial y^m/\partial x^l = \partial\psi_{\beta(\varepsilon)}(\mathbf{x})/\partial x^l\ \upsilon^{\beta(\varepsilon)}{}_B\ Y^{mB}$

$\partial z^n/\partial y^m$ are derived from the coordinate transformation,

$$\begin{aligned}\partial e/\partial z^n &= \partial/\partial z^n\ \ \Sigma_{n'} (z^{n'o(\varepsilon)} - z^{n'(\varepsilon)})^2 \\ &= 2\ \Sigma_{n'} (z^{n'o(\varepsilon)} - z^{n'(\varepsilon)})\ (-\delta_{nn'}) \\ &= -2\ (z^{no(\varepsilon)} - z^{n(\varepsilon)})\end{aligned}$$

and

$$\begin{aligned}\partial^2 e/\partial x^l \partial x^{l'} &= \partial^2 e/\partial z^n \partial z^{n'}\ \ \partial z^n/\partial y^m\ \ \partial z^{n'}/\partial y^{m'}\ \ \partial y^m/\partial x^l\ \ \partial y^{m'}/\partial x^{l'} \\ &\quad + \partial e/\partial z^n\ \ \partial^2 z^n/\partial y^m \partial y^{m'}\ \ \partial y^m/\partial x^l\ \ \partial y^{m'}/\partial x^{l'} \\ &\quad + \partial e/\partial z^n\ \ \partial z^n/\partial y^m\ \ \partial^2 y^m/\partial x^l \partial x^{l'}\end{aligned}$$

where

$\partial^2 y^m/\partial x^l \partial x^{l'} = \partial^2\psi_\beta(\mathbf{x})/\partial x^l \partial x^{l'}\ \ \upsilon^{\varepsilon\beta}{}_B\ Y^{mB}$

$\partial^2 z^n/\partial y^m \partial y^{m'}$ are derived from the coordinate transformation

$\partial^2 e/\partial z^n \partial z^{n'} = 2\ \delta_{nn'}$

The resulting algorithm is given by:

```
begin;
    evaluate e at initial $\mathbf{x}$;
    Newton: for Newton_iterate in 1..Newton_iterate_max loop
        evaluate $\partial e/\partial \mathbf{x}$ and $\partial^2 e/\partial \mathbf{x}\partial \mathbf{x}$;
        if $|\det(\partial^2 e/\partial \mathbf{x}\partial \mathbf{x})| <$ delta then                  --If matrix is singular..
            $\Delta \mathbf{x} := -\partial e/\partial \mathbf{x}$;                                   --step in direction of gradient.
        else evaluate $\Delta \mathbf{x}$ from $\partial^2 e/\partial \mathbf{x}\partial \mathbf{x}\ \Delta \mathbf{x} = -\partial e/\partial \mathbf{x}$;   --Solve for Newton step.
            if $|(\partial e/\partial \mathbf{x}|\Delta \mathbf{x})| \leq$ delta $||\partial e/\partial \mathbf{x}||\ ||\Delta \mathbf{x}||$ then   --If step orthogonal to gradient..
                $\Delta \mathbf{x} := -\partial e/\partial \mathbf{x}$;                               --step in direction of gradient.
            elsif $(\partial e/\partial \mathbf{x}|\Delta \mathbf{x}) >$ delta $||\partial e/\partial \mathbf{x}||\ ||\Delta \mathbf{x}||$ then   --If Newton step is uphill..
                $\Delta \mathbf{x} := -\Delta \mathbf{x}$;                                            --reverse the step direction.
            end if;
        end if;
        $\alpha := 1.0$;                                                     --Initialise step length scale.
        if $(\Delta \mathbf{x}|\Delta \mathbf{x}) > (\Delta \mathbf{x}|\Delta \mathbf{x})_{max}$ then                               --If step length is too large..
            $\alpha := ((\Delta \mathbf{x}|\Delta \mathbf{x})_{max}/(\Delta \mathbf{x}|\Delta \mathbf{x}))^{1/2}$;                          --reset the step length scale.
        end if;
        Search: for Search_iterate in 1..Search_iterate_max loop
            evaluate e_search at $\mathbf{x} + \alpha\ \Delta \mathbf{x}$;                         --Evaluate the error function.
            exit Search when e_search < e;                            --Exit if error function reduced.
            $\alpha := (\partial e|\Delta \mathbf{x})\ (\alpha)^2\ /\ ((\partial e|\Delta \mathbf{x}) + e - e\_search)$;   --Else reset step length scale.
        end loop Search;
        exit Newton when $(\Delta \mathbf{x}|\Delta \mathbf{x})/(1.0 + (\mathbf{x}|\mathbf{x})) \leq$ delta     --Exit if small step length..
            and $(e - e\_search)/(1.0 + e) \leq$ delta                        --and small function change..
            and Search_iterate $\leq 2$;                                   --at beginning of line search.
    end loop Newton;
    if Newton_iterate = Newton_iterate_max then                       --If convergence not reached..
        raise Convergence_error;                                      --raise an exception.
    end if;
end;
```

3.4 Linear Field Estimation

In this section and subsequent subsections we consider the problem of deriving an optimal linear estimator of the ensemble field parameters based upon the knowledge that we have about the field, its derivatives and the noise covariance of the observations. We would like the parameter estimator to be unbiased, consistent, and efficient. With an unbiased estimator the expected value of the model parameters will equal the actual parameter values. When the estimator is consistent the model parameters will converge to the actual parameters as the number of observations is increased. Of all estimators in a particular class an efficient estimator is the one having the smallest mean squared error.

We have already defined our model to be linear in the ensemble parameters. This linearity condition is a consequence of the field at an element coordinate, **x**, being linear in the element field parameters which are in turn linearly related to the ensemble field parameters via the ensemble to element parameter map:

$$\mathbf{u}^{(\varepsilon)}(\mathbf{x}) = \psi_{\beta(\varepsilon)}(\mathbf{x})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}$$

Since the field is linear in the ensemble field parameters the output error will also be linear in the parameters:

$$\mathbf{e}^{o(\varepsilon)} = \mathbf{u}^{(\varepsilon)}(\mathbf{x}^{o}) - \mathbf{u}^{o(\varepsilon)}$$

Recall also that the error function was defined as being quadratic in the output errors:

$$\mathbf{E} = E_{o(\varepsilon)\,o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')}$$

With the above conditions the error function will thus be quadratic in the ensemble field parameters.

In order to estimate the ensemble field parameter values which optimise the error function we can differentiate the error function with respect to the ensemble field parameters and equate the resulting equations to zero. Note that since the error function is quadratic in the ensemble field parameters this differention results in a set of equations linear in the ensemble field parameters. Since the weighting matrix, $E_{o(\varepsilon)\,o'(\varepsilon')}$, is symmetric and positive definite the value of the error function corresponding to this extremum will be a minimum.

The linear equations resulting from this approach are in fact the normal equations produced by the Markov estimator. It can be shown that this estimator is unbiased, consistent, and efficient (Eykhoff 1974) (Strang 1986).

3.4.1 Field Information

In this chapter we have restricted our scope to deriving the best linear unbiased estimator which approximates sampled field values and derivatives using piecewise functions.

We have implicitly assumed that all of the samples are available at the time the estimate is made. If more observations become available after the fit has been performed we could add them to the original sample and reestimate the ensemble field parameters. This approach is not particularly efficient since a large number of calculations must be duplicated as the new information becomes available. A less costly alternative is to use model adjustment techniques which update the ensemble field parameters without recomputing the contributions from earlier iterations. These techniques are beyond the scope of this thesis since we deal with static models where all measurements are performed before any fitting is attempted. They do however offer interesting possibilities where the field estimation can be performed on-line while samples are being obtained. A brief introduction to model adjustment techniques may be found in Eykhoff (1974).

In this chapter we have also restricted our analysis to sampled field values and derivatives. In some cases the field information may be available as a continuous signal. The error function is thus expressed as a weighted integral of the difference between the continuous field measurement and the fitted field. In practice this integral can be evaluated numerically. Such techniques use weighted information about field values or field values and derivatives at discrete points. The linear fitting schemes described in the following subsections may thus be used with obvious minor modifications.

3.4.1.1 Sampled Values

We want to estimate the ensemble field parameters, $\mathbf{U}^{B}$,from a finite number of observations. We shall first consider the case where the observations are sampled field values. Let $\mathbf{u}^{o(\varepsilon)}$ represent the observation o in element ε at the element position $\mathbf{x}^{o}$.

The error in the observation o is given by:

$$\mathbf{e}^{o(\varepsilon)} = \mathbf{u}^{(\varepsilon)}(\mathbf{x}^{o}) - \mathbf{u}^{o(\varepsilon)}$$

$\mathbf{u}^{(\varepsilon)}(\mathbf{x}^{o})$ may be expanded in terms of the ensemble field parameters $\mathbf{U}^{B}$ and element basis functions $\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})$:

$$\mathbf{e}^{o(\varepsilon)} = \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B} - \mathbf{u}^{o(\varepsilon)}$$

The error function is defined as:

$$\begin{aligned}
\mathbf{E} &= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')} \\
&= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B} - \mathbf{u}^{o(\varepsilon)})\ \ (\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} - \mathbf{u}^{o'(\varepsilon')}) \\
&= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \ \psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} \\
&\qquad -2\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \ \mathbf{u}^{o'(\varepsilon')} \\
&\qquad +\ \mathbf{u}^{o(\varepsilon)}\ \mathbf{u}^{o'(\varepsilon')}\ )
\end{aligned}$$

Now differentiate **E** with respect to the ensemble field parameters $\mathbf{U}^{B''}$:

$$\begin{aligned}
\partial\mathbf{E}/\partial\mathbf{U}^{B''} &= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \ (\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \delta^{B}{}_{B''}\ \psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} \\
&\qquad +\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \ \psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \delta^{B'}{}_{B''} \\
&\qquad -2\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \delta^{B}{}_{B''}\ \mathbf{u}^{o'(\varepsilon')}\ ) \\
&= 2\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}(\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \ \psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} \\
&\qquad -\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \ \mathbf{u}^{o'(\varepsilon')}\ )
\end{aligned}$$

Equating this expression to zero we obtain a set of simultaneous linear equations in $\mathbf{U}^{B'}$ at which an extremum occurs:

$$\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \ \psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} = \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o})\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \ \mathbf{u}^{o'(\varepsilon')}$$

Since $\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}$ is positive definite this extremum must be a minimum.

3.4.1.2 Sampled Derivatives

We next consider the case where the observations are sampled field derivatives. Let $\partial\mathbf{u}^{o(\varepsilon)}/\partial x^{l}$ represent the derivative with respect to the element coordinate x^{l} of the observation o in element ε at the position $\mathbf{x}^{o}$.

The error in the derivative observation o is given by

$$\mathbf{e}^{o(\varepsilon)} = \partial\mathbf{u}^{(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l} - \partial\mathbf{u}^{o(\varepsilon)}/\partial x^{l}$$

$$= \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B} - \partial\mathbf{u}^{o(\varepsilon)}/\partial x^{l}$$

We define the error function as:

$$\mathbf{E} = \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')}$$

$$= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B} - \partial\mathbf{u}^{o(\varepsilon)}/\partial x^{l})(\partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} - \mathbf{u}^{o'(\varepsilon')}/\partial x^{l'})$$

$$= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'}$$

$$-2\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \partial\mathbf{u}^{o'(\varepsilon')}/\partial x^{l'}$$

$$+\ \partial\mathbf{u}^{o(\varepsilon)}/\partial x^{l}\ \partial\mathbf{u}^{o'(\varepsilon')}/\partial x^{l'}\ )$$

Now differentiate $\mathbf{E}$ with respect to the ensemble field parameters $\mathbf{U}^{B''}$:

$$\partial\mathbf{E}/\partial\mathbf{U}^{B''} = \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \delta^{B}{}_{B''}\ \partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'}$$

$$+\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}\ \ \partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \delta^{B'}{}_{B''}$$

$$-2\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B}\ \delta^{B}{}_{B''}\ \partial\mathbf{u}^{o'(\varepsilon')}/\partial x^{l'}\ )$$

$$= 2\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}(\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'}$$

$$-\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \partial\mathbf{u}^{o'(\varepsilon')}/\partial x^{l'}\ )$$

Equating this expression to zero we obtain equations in $\mathbf{U}^{B'}$ at which an extremum occurs:

$$\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \partial\psi_{\beta'(\varepsilon')}(\mathbf{x}^{o'})/\partial x^{l'}\ \upsilon^{\beta'(\varepsilon')}{}_{B'}\ \mathbf{U}^{B'} = \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \partial\psi_{\beta(\varepsilon)}(\mathbf{x}^{o})/\partial x^{l}\ \upsilon^{\beta(\varepsilon)}{}_{B''}\ \partial\mathbf{u}^{o'(\varepsilon')}/\partial x^{l}$$

Since $\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}$ is positive definite this extremum must be a minimum. Note, however, that this set of linear equations is underdetermined. At least one value observation must be included to rectify this.

3.4.2 Methods To Solve Linear Equations

In this section we provide a very brief introduction to solution methods available to solve the linear equations resulting from our value and derivative fits.

Note that since the weighting matrix is symmetric the normal equations will also be symmetric and unless the linear equations are rank deficient the system will be positive definite. We can thus use general methods for the solution of symmetric positive definite systems of linear equations. One such algorithm is the Cholesky decomposition. Wilkinson (1968) has demonstrated the stability of this algorithm for general positive definite systems. However, when the Cholesky decomposition is used to solve the normal equations the accuracy of the solution may be severely degraded.

Many methods have been proposed to reduce the effects of ill-conditioning when solving the normal equations directly. An introduction to the different approaches may be found in Golub and Van Loan (1983).

An important tool for solving the linear least squares problem when the normal equations are rank deficient is the singular value decomposition. Essentially this approach solves the usual least squares problem when the normal equations possess full rank. When the equations are rank deficient there are, by definition, an infinite number of solutions. In this case the singular value decomposition gives the unique solution possessing the smallest 2-norm. Application of the singular value algorithm to the least squares problem is discussed in Golub and Reinsch (1970). Chan (1982) proposed an algorithm for computing the singular value decomposition which reduced the complexity over that of Golub and Reinsch whenever the number of observations was greater than 5/3 times the number of ensemble field parameters.

So far we have not taken into account the structure of the system of linear equations. If the number of ensemble field parameters is large and the support of each of the parameters is small then the normal equations will be sparse. One can significantly reduce the complexity of solution by exploiting the matrix sparsity. A general introduction to the storage and solution of sparse matrix systems may be found in Jennings (1977).

Where the nonzero matrix coefficients are clustered about the diagonal many of the operations on the zero off-diagonal entries may be discarded. The simplest implementations of this approach are band methods. These methods generally store the diagonal and off-diagonal entries, up to the bandwidth of the original matrix, in a modified array. The zero coefficients beyond the bandwidth are not included in the solution so that both the number of array entries and the complexity of computation are reduced. If the nonzero matrix entries are not clustered about the diagonal so that the bandwidth is large the bandwidth can sometimes be reduced by reordering the unknowns.

Often the nonzero matrix coefficients are not clustered uniformly about the diagonal. In such cases the band schemes may possess an unacceptably large bandwidth. More efficient storage schemes exist which vary the storage bandwith through the columns or rows. Using this approach columns or rows with nonzero entries well off the diagonal will not force the storage of extra zeros in other columns or rows. Such schemes are usually refered to as envelope or variable bandwidth methods.

In both the band and variable bandwidth methods a large number of zero coefficients may still be stored if the band itself is sparse. An obvious solution to this problem is to store and operate on only the nonzero entries. To achieve this one must not only store the nonzero coefficients but also the address of each entry. The resulting solution algorithms are somewhat more complicated than the direct, band, and variable bandwidth methods but the overheads associated with storing and referencing the addresses are usually not large. Further details about this method may be found in Gustavson (1972).

A direct solution algorithm particularly well suited to the piecewise functional structure of our fitting problem is the frontal method. This algorithm appears to have its origins in the Boeing company in Seattle sometime around 1958. Irons (1970) described the frontal method and published a FORTRAN program to assemble and solve systems of linear symmetric positive definite equations for application in the finite element method. In contrast to other methods, where the factorisation is determined by the ordering of the unknown parameters, the frontal method proceeds by processing elements one at a time, assembling each element contribution into the ensemble coefficient matrix. Elimination steps are performed whenever all contributions to a particular ensemble parameter have been processed. At the completion of each elimination step the associated row (and column in the unsymmetric case) is transfered to auxiliary storage. Using this approach only the matrix coefficients corresponding to the currently active ensemble parameters need be kept in primary storage. In the frontal method it is not the ordering of the ensemble parameters, but rather the ordering of the elements, that determines the maximum number of active ensemble parameters and hence the amount of primary storage required.

We have used both the frontal method and the Cholesky decomposition method with iterative refinement to solve the normal equations generated by the value and derivative fits described in the two previous subsections. Since our field fitting problems have all been small and dense there was little advantage in using sparse matrix techniques. The iterative refinement facility included in the Harwell Library subroutine MA22AD provided estimates of the errors in the ensemble field parameters. In all the fits performed these errors were negligible, being of the order of precision of the 64 bit floating point number representation.

3.4.3 Notes On Linear Fitting

Several points should be noted about this fitting procedure:

(1) Since the weighting matrix is symmetric the normal matrix of the system of equations is also symmetric.

(2) Where the ensemble field parameters have a local support and the observations are statistically independent the normal matrix will be sparse. In such cases the system of linear equations are best solved using sparse solution algorithms such as the frontal method.

(3) The coefficients of the normal matrix are completely determined by the ensemble position (i.e. the element and element coordinate values) and the covariance of the noise contribution to the sample observations. Where the noise covariance matrix is the same for different components of a multiple component field, and where the observations from each field have been taken from identical ensemble positions, we can fit all the components at once by forming multiple right hand sides to the system of equations.

(4) If the number of observations is less than the number of ensemble parameters the problem will be underdetermined and the normal equations singular. If the number of observations is greater than or equal to the number of ensemble parameters the normal matrix will be non-singular only if we can define a not necessarily unique one to one mapping between a subset of the observations and every ensemble parameter as follows:

> For each parameter there exists a unique observation whose associated element corresponds to one of the elements in the support of the ensemble basis function associated with the ensemble field parameter.

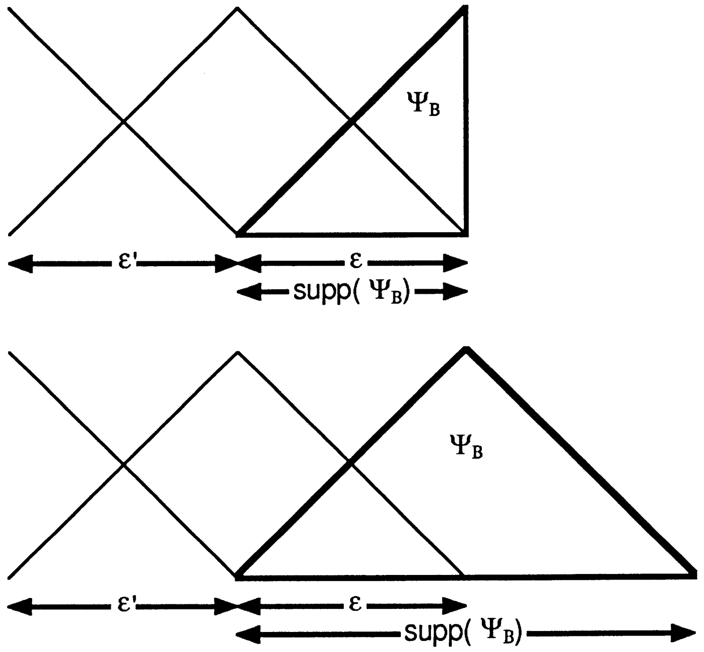
Clearly if we wish to reduce the noise contribution to the ensemble parameter estimates we will want a one to many map to exist.

(5) Where the observations are statistically independent their contributions to the normal equations will also be independent of each other. The coefficients of each observation may thus be computed in parallel on multiple data multiple instruction (MIMD) machines. Since all observations within a particular element will be associated with the same number of basis functions, although at different positions, the contribution of each observation may be computed in parallel on single instruction single data (SIMD) or vector machines.

(6) To reduce the number of ensemble field parameters some may be judiciously removed, fixed, or combined linearly with others. e.g. with bicubic Hermite basis functions one can remove the cross derivative contribution by fixing the corresponding parameters to zero. This removes the cross derivative wrinkles at each node and reduces the number of ensemble field parameters by 1/4 while still retaining C^1 continuity of the field. With tricubic Hermite basis functions by setting the cross derivative coefficients to zero the ensemble parameter count will be reduced by 3/7 while still retaining C^1 continuity of the field.

(7) Near the ensemble boundary, if it exists, the density ratio of observations to ensemble field parameters may be relatively small since the support of each ensemble parameter will usually be smaller than for corresponding parameters away from the boundary. If this is the case the normal equations governing the ensemble parameters in this region may be ill conditioned and the approximating field may vary wildly. To improve the conditioning either the number of ensemble field parameters with support at the boundary must be reduced or the number of observations near the boundary increased. Where the ensemble does not cover the entire region over which the field exists the condition of the boundary parameters may be improved by extending the support of these parameters beyond the the ensemble and taking further observations from this region. For example, consider linear Lagrange elements on a one dimensional mesh. The elements, ε and ε', near the ensemble boundary of the original mesh are given in the first diagram, together with their ensemble basis functions. The ensemble basis function, Ψ_B, located at the boundary is highlighted. In the second

diagram the support of the boundary basis function has been extended to encompass observations falling outside the original basis:



3.5 Nonlinear Field Estimation

Until now we have taken advantage of the output error being a linear function of the ensemble field parameters. This arises with the piecewise function structure because the model output at a fixed ensemble position is a linear function of the parameters. When we are fitting parameters to non-geometric fields the ensemble position corresponding to an observation will remain fixed. However, when we attempt to estimate parameters to fit the ensemble to geometric data the ensemble position assigned to each observation will usually change. The output error associated with each observation will thus be a nonlinear function of the ensemble parameters. In general this means that we can no longer find a linear estimator of the parameters to the observations so that we must resort to nonlinear optimisation techniques to find a minimum of the error function.

In the rest of this section we shall outline the nonlinear estimation method for fitting geometric fields to sampled geometric measurements. Starting from an initial ensemble configuration, for each observation find the point on the ensemble which minimises the Euclidian distance to the observation. This minimum distance defines the output error, $\mathbf{e}^{o(\varepsilon)}$, for observation o in element ε. The current error function is given, as in the linear estimator, by:

$$\mathbf{E} = E_{o(\varepsilon)\, o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')}$$

Based upon the error information from the current and previous configurations a search direction is defined which will hopefully provide a reduction in the error function. The ensemble field parameters are then updated in the search direction resulting in a new ensemble configuration. The new error function value is then computed to determine the suitability of the update. If the update gives a sufficient reduction in the value of the error function the new configuration is accepted and a new iteration begun. Otherwise further searching is performed along the search direction until a configuration giving an acceptable reduction is found. The iterations are continued until an ensemble configuration giving a local minimum in the error function is found.

Aspects of this approach are considered in more detail in the following subsections. Starting firstly with the problem of finding the derivative of the error function with respect to the ensemble field parameters we demonstrate that, because of the way we define the output error, these derivatives may be evaluated analytically. Following this a very brief introduction is given to some of the nonlinear optimisation methods suitable for minimising the error function. In the final section a few issues related to this nonlinear fitting technique are discussed.

3.5.1 Evaluation Of Derivatives

In order to facilitate the minimisation of the error function with respect to the ensemble field parameters we require as much knowledge about the nature of the error function as is possible to obtain. This knowledge may subsequently be used to estimate the values of the ensemble field parameters, based upon some assumptions about how it behaves locally, which are most likely to maximise a reduction in the error function.

Let us first take a naïve approach. Given that the ensemble position associated with each observation changes with the ensemble configuration in a somewhat obscure fashion the output error is consequently a rather complicated nonlinear function of the ensemble field parameters. One may understandably make the assumption that any analytic derivative information about the output errors, and thus the error function, is unavailable. If only the value of the error function is available the derivatives with respect to the ensemble field parameters can be approximated with finite differences. Unfortunately obtaining accurate approximations to the derivatives of multivariable fields may be a costly and difficult task. A discussion of the issues involved, with particular reference to nonlinear optimisation, may be found in Gill, Murray and Wright (1981). Generally one must make at least as many perturbed function evaluations as there are ensemble parameters in order to estimate the gradient of the error function at a given ensemble configuration. Each of these function evaluations require that the ensemble coordinates of every observation be updated in order to calculate the output errors. For our problems this is an extremely expensive task.

Fortunately the situation is not as bleak as it may appear. Since the output error is defined as the distance which locally minimises the 2-norm of the line between the ensemble and the observation, this line will be normal to the ensemble. In the next subsection we demonstrate that, as a consequence of this property, one is able to calculate first derivatives of the output error with respect to the ensemble field parameters analytically. The following subsection then gives details about how the first derivatives of the output errors, and hence the first derivatives of the error function with respect to the ensemble field parameters, may be evaluated.

3.5.1.1 Derivative Of Observation Error

Consider an observation o and its corresponding ensemble position **x**. By definition the position error or the distance d between the observation and its position on the ensemble is a local minimum and the line joining the two is perpendicular to the ensemble at that point.

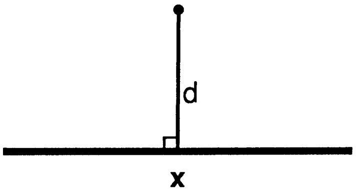
Perturbations in the ensemble configuration at the point **x** can be decomposed into three classes of movements relative to the line joining the observation and its current ensemble position:

(1) a shift of the ensemble in the direction of the line;

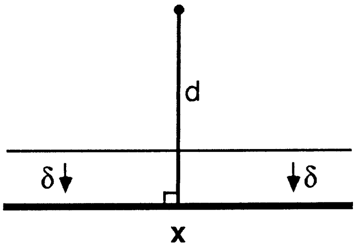
(2) a shift of the ensemble perpendicular to the direction of the line;

(3) a rotation away from normality about the current ensemble position.

Now consider the effect of these movements upon the squared distance between the observation and its new ensemble position



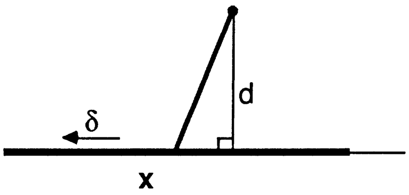
(1) In the first case where we have a shift δ in the direction of the old line the ensemble position will remain unchanged.



The derivative of the squared distance, **e**, with respect to the perturbation is given by

$$\begin{aligned}\partial e/\partial\delta &= \lim_{\delta\to 0}\ ((d+\delta)^2 - d^2)/\delta \\ &= \lim_{\delta\to 0}\ (2d\delta + \delta^2)/\delta \\ &= 2d\end{aligned}$$

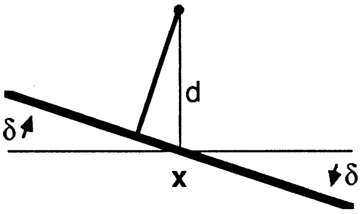
(2) In the second case where we have a shift δ perpendicular to the old geodesic the ensemble position will change.



The derivative of the squared distance with respect to the perturbation is given by

$$\begin{aligned}\partial e/\partial\delta &= \lim_{\delta\to 0}\ ((d^2 + \delta^2) - d^2)/\delta \\ &= \lim_{\delta\to 0}\ (\delta) \\ &= 0\end{aligned}$$

(3) Finally where we have a rotation δ about the current ensemble position the ensemble position associated with the observation will change.



The derivative of the squared distance with respect to the perturbation is given by

$$\begin{aligned}\partial e/\partial\delta &= \lim_{\delta\to 0}\ ((d\ \cos(\delta))^2 - d^2)/\delta \\ &= \lim_{\delta\to 0}\ (-d^2\ \sin^2(\delta))/\delta \\ &= \lim_{\delta\to 0}\ (-d^2\ \delta) \\ &= 0\end{aligned}$$

Note that the only movement resulting in a nonzero first derivative is the shift perpendicular to the ensemble. Thus the first derivative of the squared distance between an observation and its associated ensemble position with respect to perturbations in the ensemble configuration have no contributions from terms arising from changes in the ensemble position. The first derivative of the squared distance with respect to the ensemble field parameters can therefore be calculated analytically.

3.5.1.2 Derivative Of Error Function

In this section we derive the equation to evaluate the derivative of the error function **E** with respect to the ensemble field parameters $\mathbf{Y}^{m}$.

The error function is defined as:

$$\mathbf{E} = \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')}$$

where $\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}$ is a positive definite weighting matrix relating the noise covariance of observation $o(\varepsilon)$ to observation $o'(\varepsilon')$. The square of each observation error $\mathbf{e}^{o(\varepsilon)}$ is given as:

$$(\mathbf{e}^{o(\varepsilon)})^{2} = \Sigma_{n=1\ldots N}\ (z^{n}(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})^{2}$$

where $z^{o(\varepsilon)n}$ is the nth rectangular Cartesian component of observation o in element ε, $z^{n}(\mathbf{x}^{o(\varepsilon)})$ is the nth rectangular Cartesian component of the ensemble position associated with observation o in element ε and N is the dimension of the embedding space. These rectangular Cartesian components are related to the global curvilinear coordinate system, **y**, by a coordinate transformation:

$$z^{n}(\mathbf{x}^{o(\varepsilon)}) = z^{n}(\mathbf{y}(\mathbf{x}^{o(\varepsilon)}))$$

$$z^{o(\varepsilon)n} = z^{o(\varepsilon)n}(\mathbf{y}^{o(\varepsilon)})$$

The mth global curvilinear component of the ensemble position associated with observation o in element ε may be expressed as a linear combination of the ensemble field parameters:

$$y^{m}(\mathbf{x}^{o(\varepsilon)}) = \psi_{\beta}(\mathbf{x}^{o(\varepsilon)})\ \upsilon^{\beta(\varepsilon)}{}_{\mathrm{B}}\ Y^{\mathrm{B}m}$$

We now differentiate the square of the error associated with observation o in element ε with respect to the ensemble field parameters:

$$\begin{aligned}\partial(\mathbf{e}^{o(\varepsilon)})^{2}/\partial Y^{\mathrm{B}'m'} &= \partial/\partial Y^{\mathrm{B}'m'}\ \Sigma_{n}\ (z^{n}(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})^{2} \\ &= 2\ \Sigma_{n}\ (z^{n}(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})\ (\partial z^{n}/\partial y^{m}(\mathbf{x}^{o(\varepsilon)})\ \partial y^{m}(\mathbf{x}^{o(\varepsilon)})/\partial Y^{\mathrm{B}'m'})\end{aligned}$$

where

$$\partial y^{m}(\mathbf{x}^{o(\varepsilon)})/\partial Y^{\mathrm{B}'m'} = \psi_{\beta(\varepsilon)}(\mathbf{x}^{o(\varepsilon)})\ \upsilon^{\beta(\varepsilon)}{}_{\mathrm{B}'}\ \delta^{m}{}_{m'}$$

and $\partial z^{n}/\partial y^{m}(\mathbf{x}^{o(\varepsilon)})$ are derived from the coordinate transformation. From

$$\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{\mathrm{B}'m'} = (\partial(\mathbf{e}^{o(\varepsilon)})^{2}/\partial Y^{\mathrm{B}'m'})/(\mathbf{e}^{o(\varepsilon)})$$

we obtain the dervative of the error function with respect to the ensemble parameters.

$$\begin{aligned}\partial\mathbf{E}/\partial Y^{\mathrm{B}'m'} &= \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{\mathrm{B}'m'}\ \mathbf{e}^{o'(\varepsilon')} + \mathbf{e}^{o(\varepsilon)}\ \partial(\mathbf{e}^{o'(\varepsilon')})/\partial Y^{\mathrm{B}'m'}) \\ &= 2\ \mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}\ (\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{\mathrm{B}'m'}\ \mathbf{e}^{o'(\varepsilon')})\end{aligned}$$

since $\mathrm{E}_{o(\varepsilon)\ o'(\varepsilon')}$ is symmetric. Thus

$$\partial \mathbf{E}/\partial Y^{B'm'} = 4\ \Sigma_{o(\varepsilon)}\ \Sigma_{o'(\varepsilon')}\ E_{o(\varepsilon)\ o'(\varepsilon')}\ ((\Sigma_n\ (z^n(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})\ (\partial z^n/\partial y^{m'}(\mathbf{x}^{o(\varepsilon)})\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o(\varepsilon)})\ \upsilon^{\beta(\varepsilon)}{}_{B'}))$$

$$\times (\Sigma_n\ (z^n(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})2)^{1/2}\ (\Sigma_n\ (z^n(\mathbf{x}^{o'(\varepsilon')}) - z^{o'(\varepsilon')n})2)^{-1/2})$$

In the case where the observations are statistically independent, $E_{o(\varepsilon)\ o'(\varepsilon')}$ will be diagonal with entries equal to the variance, $E_{o(\varepsilon)}$, of observation o in element ε. The derivative of the error function reduces to:

$$\partial \mathbf{E}/\partial Y^{B'm'} = 4\ \Sigma_{o(\varepsilon)}\ E_{o(\varepsilon)}\ (\Sigma_n\ (z^n(\mathbf{x}^{o(\varepsilon)}) - z^{o(\varepsilon)n})\ (\partial z^n/\partial y^{m'}(\mathbf{x}^{o(\varepsilon)})\ \psi_{\beta(\varepsilon)}(\mathbf{x}^{o(\varepsilon)})\ \upsilon^{\beta(\varepsilon)}{}_{B'})$$

3.5.2 Nonlinear Optimisation

The field of nonlinear optimisation is vast. We shall attempt here to give a very brief overview of some of the algorithms currently used to solve nonlinear optimisation problems. Two special characteristics of our particular field fitting problem will be considered. Firstly, since our fits involve the minimisation of a nonlinear least squares error function, we shall look at some specialised nonlinear optimisation algorithms which exploit this structure. Reviews of the nonlinear least squares problem may be found in Dennis (1977) and Ramsin and Wedin (1977). Secondly, in those problems where the observations are independent and the ensemble field parameters have local support, the Hessian or matrix of second derivatives of the error function with respect to the ensemble field parameters will be sparse. We shall consider some methods which exploit sparsity of the Hessian.

Where we only have the error function and its first derivatives available we could update the ensemble field parameters in the direction of the gradient. Unfortunately this method of steepest descent has poor convergence properties. A better approach is to somehow estimate the Hessian, or matrix of second derivatives with respect to the ensemble field parameters, from information about the error function and its first derivatives. With second derivative information available one can then update the ensemble field parameters using Newton's method. An obvious approach to estimating the Hessian is to take finite differences of the gradient. Where the finite differences are computed accurately this discrete Newton method performs similarly to the Newton method where the Hessian is available analytically. The discrete Newton method exhibits rapid local convergence as well as an ability to avoid saddle points. In order to calculate the full matrix of second derivatives at a particular ensemble configuration at least as many finite differences as there are ensemble field parameters must be evaluated. Where there are a large number of ensemble parameters the discrete Newton method can be quite inefficient.

There exist a class of nonlinear minimisation methods which, rather than estimate the Hessian at a given ensemble configuration, build up information about the second derivatives from the observed values and first derivatives as the minimisation proceeds. Despite their relatively recent development these quasi-Newton methods have enjoyed considerable success. A review of the theory and performance of the quasi-Newton algorithms may be found in Dennis and Moré (1977). A number of distinct quasi-Newton methods have been developed, the two most popular being the Davidon-Fletcher-Powell update and its more effective dual the Broydon-Fletcher-Goldfarb-Shanno update. Both of these updates preserve positive definiteness of the Hessian. They may also be formulated in such a way that updates are made to the inverse of the Hessian so that the expensive operation of matrix inversion can be avoided.

We now consider the special structure of the error function. Recall that the error function was defined as being quadratic in the output errors:

$$\mathbf{E} = E_{o(\varepsilon)\, o'(\varepsilon')}\ \mathbf{e}^{o(\varepsilon)}\ \mathbf{e}^{o'(\varepsilon')}$$

Since $E_{o(\varepsilon)\, o'(\varepsilon')}$ is symmetric the first derivative of the error function with respect to the ensemble field parameters, Y^{Bm}, are:

$$\partial\mathbf{E}/\partial Y^{Bm} = 2\ E_{o(\varepsilon)\, o'(\varepsilon')}\ (\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{Bm}\ \mathbf{e}^{o'(\varepsilon')})$$

Differentiating once more we obtain the second derivatives:

$$\partial^2\mathbf{E}/\partial Y^{Bm}\ \partial Y^{B'm'} = 2\ E_{o(\varepsilon)\, o'(\varepsilon')}\ (\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{Bm}\ \partial(\mathbf{e}^{o'(\varepsilon')})/\partial Y^{B'm'} +\ \partial^2(\mathbf{e}^{o(\varepsilon)})/\partial Y^{Bm}\ \partial Y^{B'm'}\ \mathbf{e}^{o'(\varepsilon')})$$

Notice that the Hessian is made up of two components, one involving the outer product of the first derivatives and the other involving products of the output error and the second derivatives. If the output errors or the second derivatives are small at the minimum then the Hessian may be approximated by only the first derivative contributions:

$$\partial^2\mathbf{E}/\partial Y^{Bm}\ \partial Y^{B'm'} \approx 2\ E_{o(\varepsilon)\, o'(\varepsilon')}\ (\partial(\mathbf{e}^{o(\varepsilon)})/\partial Y^{Bm}\ \partial(\mathbf{e}^{o'(\varepsilon')})/\partial Y^{B'm'})$$

This method of approximating second derivatives from only first derivative information is called the Gauss-Newton method. When the gradient is of full column rank this method can achieve a quadratic rate of convergence. Problems do arise, however, when the gradient does not have full column rank. One method of avoiding problems associated with a singular or non positive definite Hessian was proposed independently by Levenberg (1944) and Marquardt (1963). By adding a non-negative multiple of the identity matrix to the Hessian approximation of the Gauss-Newton method the Levenberg-Marquardt Hessian can be forced to become positive definite. If a large multiple is added the update becomes small and parallel to the steepest descent direction. The choice of how large a multiple to add is discussed in Moré (1977).

For both the Gauss-Newton and Levenberg-Marquardt methods a good approximation to the Hessian is obtained only if the output errors at optimality are small. Gill and Murray (1978) proposed a quasi-Newton update to the Hessian to approximate the neglected second derivative terms in the least squares problem. It should be noted that the quasi-Newton updates for the least squares problem do not possess the property of hereditary positive definiteness.

None of the above methods have exploited the sparsity patterns that may be present in the Hessian. If the sparsity pattern is known *a priori* the finite difference or discrete Newton method can be quite attractive since the second derivatives corresponding to the positions of the zeros need never be calculated. By reordering the ensemble field parameters it is often possible to considerably reduce the number of gradient evaluations per iteration.

Sparse quasi-Newton updates have been proposed by Schubert (1970), Toint (1981), and Thapa(1981). Unfortunately these methods do not perform well. This is due in part to the loss of hereditary positive definiteness even when the solution is close to optimality. Note also that since the inverse of a sparse matrix is not in general sparse the Hessian updates must be performed on the Hessian as opposed to the factorised Hessian. A sparse system of linear equations must thus be solved at each iteration.

In the nonlinear geometric field fits that we performed the Hessian did not exhibit a large degree of sparsity. In some cases the output errors were relatively large even at the optimal solution. We thus used a modified Broydon-Fletcher-Goldfarb-Shanno quasi-Newton routine from the Harwell Library, VA13AD, to minimise the error function.

3.5.3 Notes On Nonlinear Fitting

Some points should be noted about the nonlinear fitting procedure:

(1) With the optimisation methods described above we require an initial parameter estimate. The choice of an adequate initial state is important to facilitate convergence to the desired solution as well as ensuring that this solution can be achieved with a minimum of effort. Often a good initial estimate can be obtained by using the linear method already discussed. Each observation must be assigned an initial ensemble position. If the initial ensemble configuration is smooth often some explicit mapping is able to be made between the space from which the observations are made and the ensemble.

(2) It is useful to monitor the solution at regular intervals as the iterations proceed. An effective way of achieving this is to provide a graphical output of the current state. In this way one can readily detect if the configuration is tending toward an undesirable solution. If an undesirable configuration is detected the optimisation routine may be halted, the offending parameters reassigned to more appropriate values, and the optimisation routine restarted. The above method ensures that the other parameter values, some of which may be nearly optimal, remain unaltered.

(3) Advantage can be taken of the local support of the ensemble parameters when calculating the derivatives of the error function with respect to these parameters. Only derivatives of the output error with respect to an ensemble parameter of observations lying within the support of the parameter will contribute terms to the derivative of the error function with respect to that parameter.

(4) Since the ensemble position associated with each observation is independent of all others they may be computed in parallel. This is a very useful property as these operations are usually the most expensive component of the error function evaluation.

(5) In problems with many ensemble parameters convergence to an optimal solution may be quite slow. Where some of the parameters are not orthogonal to others we may find that instabilities arise due to ill-conditioning of the problem. These effects are most pronounced when the solution is far from optimal. To counter these effects the number of degrees of freedom at the initial iterations may be reduced by constraining them to particular values or by combining them linearly with some of the remaining parameters. It should be noted that such a scheme may have the effect of increasing the support of some parameters. The constrained parameters may then be selectively freed as convergence to the desired solution is approached.

(6) In the nonlinear optimisation scheme we are not restricted to using the ensemble parameters as optimisation degrees of freedom. Other parameters defining the geometry such as the coefficients of the ensemble to element parameter map may also be used. These coefficients give direct control over the mapping of the basis functions from the element space to the ensemble space. It should be noted, however, that we may no longer be able to compute analytic derivatives with respect to these parameters. The effect of including these coefficients as optimisation degrees of freedom may be illustrated with cubic Hermite elements. In the case of cubic Hermite elements the ensemble to element parameter map corresponding to derivative parameters at a particular boundary defines a local mapping of the element coordinates, $\mathbf{x}$, to the global coordinates, $\mathbf{y}$, about that boundary. Varying the values of these coefficients alters the local density of the element coordinates with respect to the global coordinates. Where the local density is high rapid changes in the field may occur. Where the density becomes infinite the field may be discontinuous in its value or derivative.

3 Bibliography

Babuska, I., Aziz, A. K. (1972) "Survey lectures on the mathematical foundations of the finite element method" in *The Mathematical Foundations Of The Finite Element Method With Applications To Partial Differential Equations,* edited by Aziz, A. K., Academic Press, New York, pages 5-359.

Babuska, I., Szabo, B. A., Katz, I. N. (1981) "The *p*-version of the finite element method" *SIAM Journal On Numerical Analysis,* June 1981, volume 18, number 3, pages 515-545.

Bartels, R. H., Jezioranski, J. J. (1985) "Least-squares fitting using orthogonal multinomials" *ACM Transactions On Mathematical Software,* September 1985, volume 11, number 3, pages 201-217.

Chan, T. F. (1982) "An improved algorithm for computing the singular value decomposition" *ACM Transactions On Mathematical Software,* 1982, volume 8, pages 84-88.

de Boor, C. (1978) *A Practical Guide To Splines,* Springer-Verlag, New York.

Dennis, J. E., Jr. (1977) "Nonlinear least squares" in *The State Of The Art In Numerical Analysis,* edited by Jacobs, D., pages 269-312, Academic Press, London and New York.

Dennis, J. E., Jr., Moré, J. J. (1977) "Quasi-Newton methods, motivation and theory" *SIAM Review,* January 1977, volume 19, number 1, pages 46-89.

Eykhoff, P. (1974) *System Identification,* John Wiley & Sons, London.

Gill, P. E., Murray, W. (1978) "Algorithms for the solution of the nonlinear least-squares problem" *SIAM Journal On Numerical Analysis,* October 1978, volume 15, number 5, pages 977-992.

Gill, P. E., Murray, W., Wright, M. H. (1981) *Practical Optimization,* Academic Press, London.

Golub, G. H., Reinsch, C. (1970) "Singular value decomposition and least squares solutions" *Numerical Mathematics,* 1970, volume 14, pages 403-420.

Golub, G. H., Van Loan, C. F. (1983) *Matrix Computations,* North Oxford Academic, Oxford.

Gustavson, F. G. (1972) "Basic techniques for solving sparse systems of linear equations" in *Sparse Matrices And Their Applications,* edited by Rose, D. J., Willoughby, R. A., pages 41-52, Plenum Press, New York.

Irons, B. M. (1970) "A frontal solution program for finite element analysis" *International Journal For Numerical Methods In Engineering,* 1970, volume 2, pages 5-32.

Jennings, A. (1977) *Matrix Computations For Engineers And Scientists,* John Wiley & Sons, London.

Levenberg, K. (1944) "A method for the solution of certain problems in least squares" *Quarterly Of Applied Mathematics,* 1944, volume 2, pages 164-168.

Marquardt, D. (1963) "An algorithm for least-squares estimation of nonlinear parameters" *SIAM Journal On Applied Mathematics,* 1963, volume 11, pages 431-441.

Moré, J. J. (1977) "The Levenberg-Marquardt algorithm: Implementation and theory" in *Numerical Analysis,* edited by Watson, G. A., pages 105-116, Lecture Notes In Mathematics 630, Springer-Verlag, Berlin.

Oden, J. T. (1972) *Finite Elements Of Nonlinear Continua,* McGraw-Hill Book Company, New York.

Ramsin, H., Wedin, P. Å. (1977) "A comparison of some algorithms for the nonlinear least squares problem" *BIT,* 1977, volume 17, pages 72-90.

Schubert, L. K. (1970) "Modification of a quasi-Newton method for nonlinear equations with a sparse Jacobian" *Math. Comp.,* 1970, volume 24, pages 27-30.

Schumaker, L. L. (1976) "Fitting surfaces to scattered data" in *Approximation Theory II,* edited by Lorentz, G. G., Chui, C. K., Schumaker, L. L., pages 203-268, Academic Press Inc., New York.

Strang, G. (1986) *Introduction To Applied Mathematics,* Wellesley-Cambridge Press, Cambridge.

Thapa, M. N. (1983) "Optimization of unconstrained functions with sparse hessian matrices: Quasi-Newton methods" *Mathematical Programming,* 1983, volume 25, pages 158-182.

Toint, P. (1981) "A note about sparsity exploiting quasi-Newton updates" *Mathematical Programming,* 1981, volume 21, pages 172-181.

Wilkinson, J. H. (1968) "A priori error analysis of algebraic processes" in Proc. International Congress Math., pages 629-639, Izdat. Mir, Moscow.

Zienkiewicz, O. C., Morgan, K. (1982) *Finite Elements And Approximation,* Illegal Copy, Hong Kong.