GLOBAL FRINGE SEARCH TECHNIQUES FOR VLBI

Frederic R. Schwab

April 1982

I. Introduction

The usual means of fringe processing in VLBI is a correlator-based technique. But sizeable components of delay and fringe rate are dependent upon effects that are ascribable to individual antennas, along with their associated i.f. and l.o. systems and the atmospheric conditions overhead. For this reason one would expect that it should be worthwhile, in cases of low signal-to-noise ratio (S/N), to attempt to solve simultaneously, using observations from all baselines, for the antenna-based effects. Such a global method would obviate the requirement that there be relatively strong fringes on every baseline to be processed. An additional advantage of a global method is that the phase closure relations satisfied by the data will not be corrupted. The closure properties are disturbed when the standard technique is applied.

Following a description of the standard approach to fringe search, two global fringe search techniques are outlined below. The first global method that is described is a least-squares approach. The second, a Fourier transform (FT) method, is a straightforward generalization of the standard technique. The least-squares method requires good starting guesses but is more flexible than the FT method. In initial tests, a hybrid algorithm has been used, combining both of the global techniques. The FT method is used to generate starting guesses for the least-squares algorithm.

II. The Standard Approach

Assuming that the correlator responses have been transformed to the time and frequency domain (t,ν), the standard approach to fringe search can be described as follows. Observations $\tilde{V}_{ij}(t_k, \nu_\ell)$ on the i-j baseline $(0<i<j\leq n)$ at times t_k, $k=0,1,\ldots,n_t$, and at frequencies ν_ℓ, $\ell=0,1,\ldots,n_\nu$, are related to some "true" $V_{ij}(t,\nu)$ according to

$\tilde{V}_{ij}(t_k,\nu_\ell) = g_i(t_k,\nu_\ell)\ \bar{g}_j(t_k,\nu_\ell)\ V_{ij}(t_k,\nu_\ell)(1 + \varepsilon_{ijk\ell}) + \delta_{ijk\ell}$, where the antenna effects (clock errors, etc.) have been absorbed into complex-valued functions g of (t,ν), and where the additive errors, δ, and the components ε of the multiplicative errors are sufficiently small and well-behaved. Each g can be written as $g_k(t,\nu) = a_k(t,\nu)e^{i\psi_k(t,\nu)}$ (ψ_k is called the antenna phase of antenna k). Although the a_k are functions of (t,ν), it is assumed that they change slowly enough with time and frequency that $g_k(t,\nu) = a_k\, e^{i\psi_k(t,\nu)}$, with a_k constant over the (t,ν)-interval of the fringe search. Also, $|V_{ij}|$ is assumed to be constant over the relevant interval. Then, to first order,

$$(1)\quad \tilde{V}_{ij}(t_k,\nu_\ell) \simeq a_i a_j\, V_{ij}(t_o,\nu_o)\, e^{i[(\psi_i-\psi_j)(t_o,\nu_o)]} \cdot e^{i\left[\left.\frac{\partial(\psi_i-\psi_j+\phi_{ij})}{\partial t}\right|_{(t_o,\nu_o)}(t_k-t_o) + \left.\frac{\partial(\psi_i-\psi_j+\phi_{ij})}{\partial \nu}\right|_{(t_o,\nu_o)}(\nu_\ell-\nu_o)\right]},$$

where $\phi_{ij} \equiv \arg V_{ij}$.

The quantities $r_{ij} \equiv \left.\frac{\partial(\psi_i-\psi_j+\phi_{ij})}{\partial t}\right|_{(t_o,\nu_o)}$ and $\tau_{ij} \equiv \left.\frac{\partial(\psi_i-\psi_j+\phi_{ij})}{\partial \nu}\right|_{(t_o,\nu_o)}$, which are called, respectively, the fringe rate and the delay for baseline i-j at (t_o,ν_o), can be estimated by searching for the maximum modulus of the Fourier transform $\hat{F}_{ij}(r,\tau)$ of the distribution $F_{ij}(t,\nu)$ defined by $F_{ij}(t,\nu) = \sum_{k,\ell} \delta(t-(t_k-t_o),\ \nu-(\nu_\ell-\nu_o))\,\tilde{V}_{ij}(t_k,\nu_\ell)$, where δ is Dirac's δ-function* (i.e.,

*In practice one doesn't compute $\hat{F}_{ij}$ directly, but rather a discrete approximation to it, via the FFT algorithm, and then interpolation yields the parameter estimates.

$\hat{\delta}(r,\tau) \equiv 1$). Furthermore, $\hat{F}_{ij}(r_{ij}, \tau_{ij}) \simeq a_i a_j V_{ij}(t_o,\nu_o)\, e^{i[(\psi_i-\psi_j)(t_o,\nu_o)]}$ These quantities, r_{ij}, τ_{ij}, and $\hat{F}(r_{ij},\tau_{ij})$, then, are the product of the standard fringe search technique. They allow one to produce corrected observations

$$\frac{\tilde{V}_{ij}(t_k,\nu_\ell)}{\hat{F}_{ij}(\tau_{ij},r_{ij})}\; e^{-i[(t_k-t_o)r_{ij} + (\nu_\ell-\nu_o)\tau_{ij}]}$$

which, to the extent that the first-order model (1) is valid, and that $\frac{\partial\phi_{ij}}{\partial t}$ and $\frac{\partial\phi_{ij}}{\partial\nu}$ are negligible, can be averaged coherently over time and frequency. Note carefully, though, that the technique does not separate the $\frac{\partial\psi}{\partial t}$ and $\frac{\partial\phi}{\partial t}$ components of the rates, nor the $\frac{\partial\psi}{\partial\nu}$ and $\frac{\partial\phi}{\partial\nu}$ components of the delays, as one might wish.

Following the fringe search, the corrupting influence of the quantities $a_k^{-1}\, e^{-i\psi_k(t_o,\nu_o)} = g_k^{-1}(t_o,\nu_o)$ is removed in post-processing by the self-calibration/hybrid mapping techniques.

The major disadvantage of the standard approach is that relatively high S/N ratio (i.e., small ϵ and δ) is required on each baseline in order to obtain reliable estimates of all the r_{ij} and τ_{ij}. The obvious approach toward a fringe processing method for the low S/N regime is to try to separate the antenna-based components $\left.\frac{\partial\psi_k}{\partial t}\right|_{(t_o,\nu_o)}$ and $\left.\frac{\partial\psi_k}{\partial\nu}\right|_{(t_o,\nu_o)}$ of the fringe rates and the delays by means of some simultaneous solution over all baselines. Obviously the source phase, ϕ_{ij}, cannot be separated into antenna-based components - so a source model V_{ij}, approximating V_{ij}, is required, a la self-calibration/hybrid mapping, in the two global fringe search methods described below.

III. A Least-Squares Approach

Here, given a source model, V, approximating $\mathcal{V}$, above, we consider the problem of least-squares estimation of the antenna-based components of fringe rate and delay. Simultaneously we wish to estimate the g_k. The antenna-based parameters are the a_k, the antenna phases, $\psi_{k,o} \equiv \psi_k(t_o,\nu_o)$, the antenna rates, $r_k \equiv \left.\frac{\partial \psi_k}{\partial t}\right|_{(t_o,\nu_o)}$, and the antenna delays, $\tau_k \equiv \left.\frac{\partial \psi_k}{\partial \nu}\right|_{(t_o,\nu_o)}$. We shall fit the ensemble of observations in the (t,ν)-sample space to the model

$$\tilde{V}_{ij}(t,\nu) = a_i a_j \, V_{ij}(t,\nu) \, e^{i[(\psi_{io}-\psi_{jo}) + (r_i-r_j)(t-t_o) + (\tau_i-\tau_j)(\nu-\nu_o)]} + \text{noise}.$$

Since only pairwise differences of the ψ_{ko}, the r_k, and the τ_k appear in the model, we shall settle, in the absence of other assumptions, upon estimating 4n-3 parameters, omitting ψ_{po}, r_p, and τ_p for some choice p of "reference antenna." Thus, if we set $\psi_{po} = r_p = \tau_p = 0$, then we estimate antenna phase, rate, and delay with respect to the phase, rate, and delay at antenna p. A least-squares formulation of the problem, then, is to minimize the functional

$$S(x) = \sum_{k,\ell} \sum_{i<j} \sigma^{-2}_{ijk\ell} \left| \tilde{V}_{ij}(t_k,\nu_\ell) - a_i a_j \, V_{ij}(t_k,\nu_\ell) \, e^{i[(\psi_{io}-\psi_{jo})+(r_i-r_j)(t_k-t_o)+(\tau_i-\tau_j)(\nu_\ell-\nu_o)]} \right|^2.$$

Here, x is the (column) vector of unknown parameters, $x = \text{col}'(a_1,\ldots,a_n,\psi_{1o},\ldots,\psi_{no},r_1,\ldots,r_n,\tau_1,\ldots,\tau_n)$, the prime denoting that antenna p's parameters (except for a_p) are omitted; and $\sigma^2_{ijk\ell}$ is an estimate of the variance of $\tilde{V}_{ij}(t_k,\nu_\ell)$.

Define $E_{ijk\ell} \equiv e^{i[(\psi_{io}-\psi_{jo})+(r_i-r_j)(t_k-t_o)+(\tau_i-\tau_j)(\nu_\ell-\nu_o)]}$. Then, we might choose to minimize $S_1(x) = \sum_{k,\ell} \sum_{i<j} \frac{1}{\sigma^2} \left| \tilde{V}_{ij}(t_k,\nu_\ell)/V_{ij}(t_k,\nu_\ell) - a_i a_j \, E_{ijk\ell} \right|^2$ instead of S(x) (S_1 might be preferred if data storage were at a premium); or, if it were assumed that $a_k = 1$, $\forall k$, $S_2(x) = \sum_{k,\ell} \sum_{i<j} \frac{1}{\sigma^2} \left| \tilde{V}_{ij}(t_k,\nu_\ell)/V_{ij}(t_k,\nu_\ell) - E_{ijk\ell} \right|^2$, where the a_k have been deleted from the vector, x, of unknowns; or, if one wished to discard the

amplitude information, $S_3(x) = \sum_{k,\ell} \sum_{i<j} \frac{1}{\sigma^2} \left| e^{i[\tilde{\phi}_{ij}(t_k,\nu_\ell)-\phi_{ij}(t_k,\nu_\ell)]} - E_{ijk\ell} \right|^2$, where $\tilde{\phi}_{ij} \equiv \arg \tilde{V}_{ij}$. Thus far, in the attempt at a practical implementation of the least-squares method, only the form S_3 has been employed.

Additionally, one might choose to incorporate prior knowledge of the parameters: for example, one might tabulate the running mean, m_{r_k}, of an antenna's fringe rate, r_k, along with an empirical measure of the rms scatter σ_{r_k} about the running mean. If one were then to assume a Gaussian prior distribution $N(m_{r_k}, \sigma^2_{r_k})$ for r_k, one would add to S the "penalty" term $\sum_{\substack{k=1 \\ k\neq p}}^{n} \frac{(r_k - m_{r_k})^2}{\sigma^2_{r_k}}$. Or, if there were simply a desire to constrain the estimates of some parameter to a given interval, one could add a differentiable penalty function to S which would be small over the given interval and rise sharply at its boundaries. These techniques would serve to reduce the variance in the parameter estimates, at the expense of increased bias, over and above the bias inherent in the basic nonlinear least-squares approach. Preliminary tests of the method indicate that constraints are not required in cases of moderate S/N ratio.

S typically has many local minima, as well as other critical points. One can reasonably expect that a standard iterative minimization algorithm, starting with a reasonable initial guess, reliably and efficiently should be able to locate some local minimum. But locating a starting point for which the global minimum will be the point of attraction of the algorithm is more difficult. However, secondary minima of S ought to be spaced in delay and rate roughly like the spacing of the maxima of the cosine transforms of $f(t) = \sum \frac{1}{\sigma_k^2} \delta(t-t_k+t_o)$ and $g(\nu) = \sum \frac{1}{\sigma_\ell^2} \delta(\nu-\nu_\ell+\nu_o)$, whenever the variance estimates are separable functions of t and ν; i.e., whenever $\sigma_{ijk\ell} = \sigma_{ij}\sigma_k\sigma_\ell$. In tests with model data, this observation has improved the success rate of the algorithm

quite significantly when it has been started with quite arbitrary initial guesses. But a more fruitful approach seems to be to use a generalization of the standard method in order to generate starting points. This scheme is described in the following section.

A detailed description of an algorithm which seems effective in solving the least-squares problem as formulated above is given in the Appendix.

IV. A Generalization of the Standard Approach

As in the least-squares minimization of S_3, let us discard the amplitude information but retain the phases, $\tilde{\phi}$ and ϕ. Then, to estimate the ψ_{ko}, the r_k, and the τ_k, a global Fourier transform approach analogous to the standard approach is easily derived.

For simplicity of exposition, assume that antenna number 1 has been chosen as reference antenna; i.e., p=1. Define $D_{k\ell}(t,\nu) \equiv \delta(t-(t_k-t_o),\ \nu-(\nu_\ell-\nu_o))$. Then the location of the maximum modulus of the Fourier transform of

$$F_{12}(t,\nu) = \sum_{k,\ell} \frac{1}{\sigma^2_{12k\ell}} D_{k\ell}(t,\nu)\, e^{i[(\tilde{\phi}_{12}-\phi_{12})(t_k,\nu_\ell)]}$$

provides an estimate of r_2 and τ_2, and the argument there is an estimate of $\psi_{2,o}$. By the phase closure relations, though, $\tilde{\phi}_{12}-\phi_{12} = \tilde{\phi}_{13}-\phi_{13}-\tilde{\phi}_{23}+\phi_{23}$ + noise. In general, for $j \neq 1,2$, define

$$F_{1j2}(t,\nu) = \sum_{k,\ell} \frac{1}{\sigma^2_{1jk\ell} + \sigma^2_{2jk\ell}} D_{k\ell}(t,\nu)\, e^{i[(\tilde{\phi}_{1j}-\phi_{1j}-\tilde{\phi}_{2j}+\phi_{2j})(t_k,\nu_\ell)]}.$$

Since the differences $\tilde{\phi} - \phi$ occurring in F_{12} and in the F_{1j2} are all independent (apart from whatever scheme was used to derive the model) the Fourier transform of $F \equiv F_{12} + \sum_{j\neq 1,2} F_{1j2}$ should yield better parameter estimates than any one of the $F_{\_}$'s alone.

Paths like 1-3-4-2, with functions F_{1jk2} defined analogously to those above, provide further estimates of r_2, τ_2, and $\psi_{2,o}$. F_{1342} is not independent of F_{132} and F_{142}, although it incorporates new data from the 3-4 baseline. So it is not clear how to sum all of the $F_{\_}$'s together with properly chosen weights. Nevertheless, for an arbitrary choice, p, of reference antenna, the paths to antenna ℓ which incorporate at least partially independent phase differences can be enumerated as:

$\{(p-\ell)\}$,

$\{(p-j-\ell)\colon j=1,\ldots,n,\ j\neq p,\ell\}$,

and $\{(p-j-k-\ell)\colon\ j=1,\ldots,n-1, j\neq p,\ell;\ k=j+1,\ldots,n, k\neq p,\ell\}$.

As in the standard fringe search technique, in implementing this method one would compute a discrete approximation to $\hat{F}$, via the FFT algorithm, rather than $\hat{F}$ itself. Given prior knowledge of the rates and delays, it would be straightforward to constrain the parameter estimates by doing a restricted search for the maximum modulus of $\hat{F}$. It is not readily apparent how the method could be generalized to provide estimates of the a_k.

V. Discussion

Bill Cotton and I are implementing a hybrid global fringe search algorithm within the AIPS interactive data reduction system. The FT method is used to provide starting guesses for the least-squares algorithm, as shown in the block diagram of Figure 1. In the least-squares routine, the functional S_3 is minimized. The weights $1/\sigma^2_{ijk\ell}$, are proportional to the product, $w_i w_j$, of two "antenna weights", times the square of the modulus of the model visibility, $|V_{ijk\ell}|^2$. Thus, baselines where the source is heavily resolved are given small weight. A data editing capability based on goodness-of-fit to the model (1) has also been incorporated. A priori knowledge of the delays and rates often is good enough that the dotted line path shown in Figure 1 can be followed. Typically the FT method provides good enough parameter estimates for the least-squares algorithm to converge in 4-6 iterations. A sample result of a run of the program is shown in Figure 2. Raw phase observations on one baseline in one frequency channel, for a brief interval of time, are plotted in the upper portion of the figure. The corrected phases are shown in the lower portion.

As an alternative to the least-squares algorithm, one could use instead an ℓ_1 minimization method, in order to reduce the sensitivity to wild data points. This approach, whose application to ordinary self-calibration/hybrid mapping is described in [4], readily could be adapted to the problem at hand.

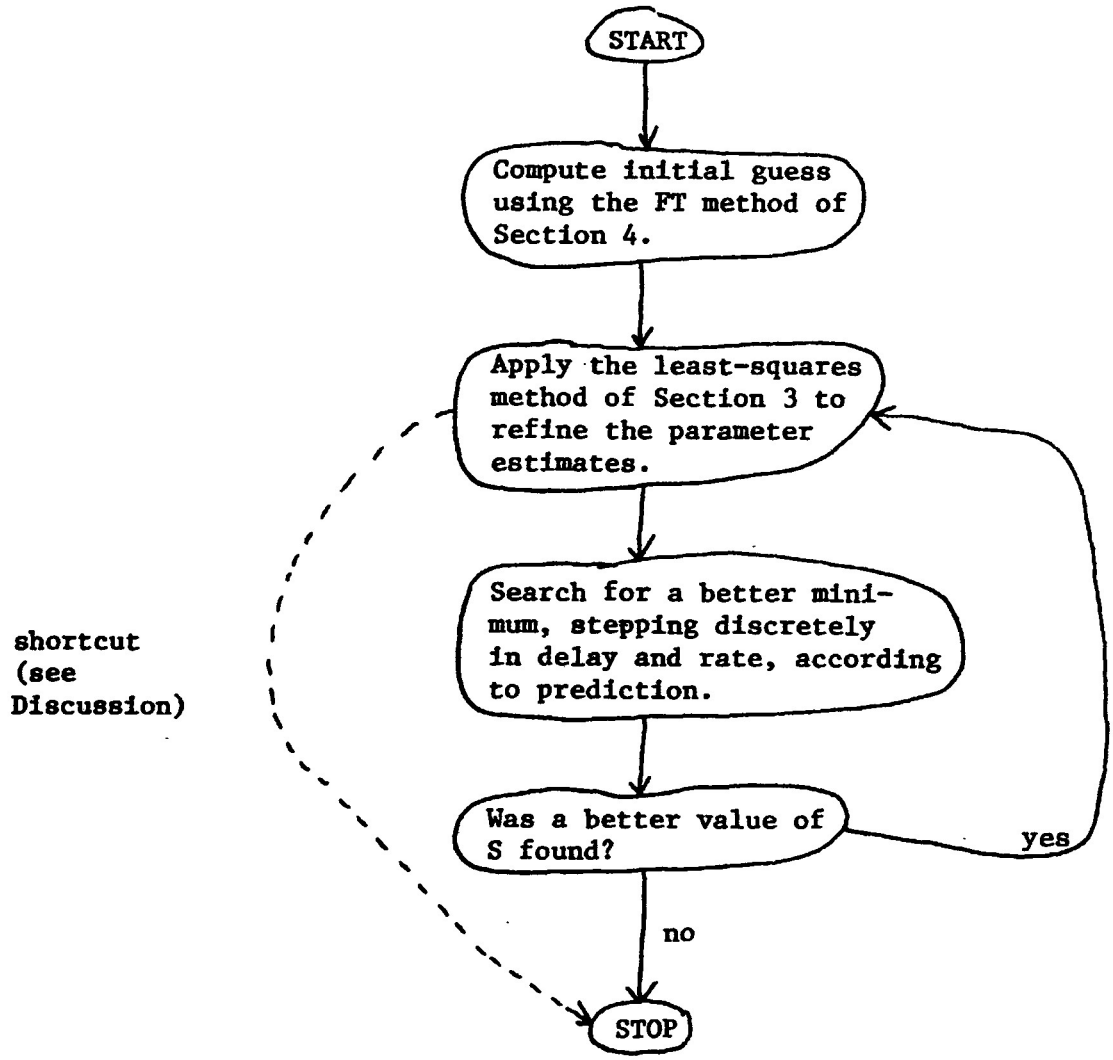

Figure 1. Hybrid method, combining both global methods.

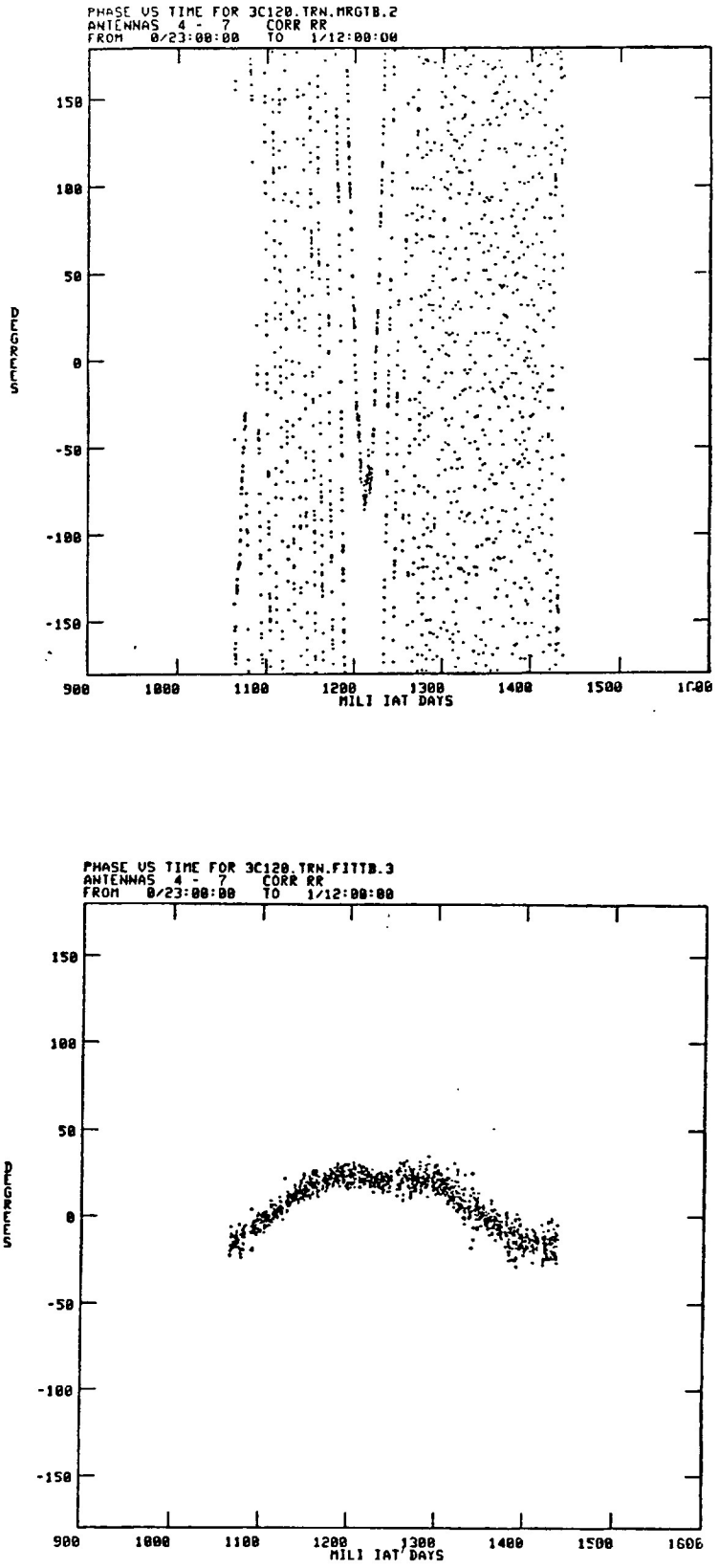

Figure 2. Phase observations on one baseline in one frequency channel before correction (top) and after (bottom).

References

1. W. Murray (ed.), Numerical Methods for Unconstrained Optimization, Acad. Pr., NY, 1975.

2. J. J. Dongarra, J. R. Bunch, C. B. Moler and G. W. Stewart, LINPACK User's Guide, SIAM, Philadelphia, 1979.

3. B. T. Smith, J. M. Boyle, J. J. Dongarra, B. S. Garbow, Y. Ikebe, V. C. Klema and C. B. Moler, Matrix Eigensystem Routines - EISPACK Guide, Lecture Notes in Computer Science 6, Springer-Verlag, NY, 1976.

4. F. R. Schwab, "Robust Solution for Antenna Gains," VLA Scientific Memo. No. 136, September 1981.

Appendix

Implementation of the Least-Squares Method

In this section, S denotes whichever functional of Section 3, S, S_1, S_2, or S_3, is to be minimized. x is the corresponding column vector of unknown parameters. With n antennas, there are N = 4n-3 parameters unless S_3 has been selected, in which case N = 3n-3. Let $\nabla S(x)$ denote the gradient of S evaluated at x; i.e., $\nabla S(x) = \text{col}\left(\frac{\partial S}{\partial x_1}, \ldots, \frac{\partial S}{\partial x_N}\right)$; and let H(x) denote the Hessian matrix of S at x; i.e., $H(x) = \left(\frac{\partial^2 S}{\partial x_i \partial x_j}\right)_{i,j=1}^{N}$. In general, S has many critical points - local minima, local maxima, and saddle points. At any critical point x^*, $\nabla S(x^*) = 0$. At a local minimum, $H(x^*)$ is positive semidefinite (i.e., all of its eigenvalues are nonnegative), and at a proper local minimum (i.e., when all parameters are well-determined) $H(x^*)$ is positive definite. At a local maximum, H is negative semidefinite (i.e., none of its eigenvalues are positive), and at a saddle point H is indefinite (i.e., it has some negative and some positive eigenvalues). Away from a local minimum, $-\nabla S(x)$ points in the direction of steepest descent in the sense of the standard Euclidean metric, $\sqrt{(x-y)^T(x-y)}$. For any positive definite matrix G, $-G\nabla S(x)$ is also a steepest descent direction - steepest in the sense of the metric $\sqrt{(x-y)^T G\,(x-y)}$ derived from the inner product $x^T G y$. Clearly, because of the multitude of local maxima and saddle points, a descent algorithm is required for the problem at hand. One would not generally choose, though, to step in the direction of steepest descent in the Euclidean metric because such a strategy can lead to arbitrarily slow convergence. But whenever H is positive definite, $-H^{-1}(x)\nabla S(x)$ is a steepest descent direction. And, by the Kantorovich theorem, steps in this search direction will converge eventually at a quadratic rate in the neighborhood of any proper local minimum of S.

Because of the special structure of our problem, a simple variant of

Newton's method for root-finding, applied to the equation $\nabla S(x^*) = 0$, is practical. A variant is required because we want the algorithm to converge on only one type of critical point - a local minimum. The reason that the structure is special and that Newton's method is practical is that, when $N = 3n-3$, each observation, since it involves only 2 antennas, and, therefore, at most 6 unknown parameters, contributes to at most 6 elements of ∇S, to at most 6 of the diagonal elements of H, and to at most 15 subdiagonal elements of the (symmetric) matrix H. Furthermore, all of these contributions are easy to compute. When $N = 4n-3$, the contributions are to at most 8, 8, and 28 elements, respectively. For many other problems Newton's method is impractical because each observation may involve all of the parameters and because analytic second order derivatives of the model may be very difficult to obtain.

The raw form of Newton's algorithm is:

Given an initial guess $x^{(0)}$, at the $k^{\underline{th}}$ iteration form the new parameter estimate $x^{(k+1)} = x^{(k)} - \alpha_k H^{-1}(x^{(k)})\nabla S(x^{(k)})$, where $\alpha_k > 0$ is chosen so that $S^{(k+1)} \leq S^{(k)}$. The difficulty is that H^{-1} must be positive definite in order for $-H^{-1}\nabla S$ to be a descent direction. Hence, any time that $H(x^{(k)})$ has nonpositive eigenvalues, we shall wish to replace it by $\tilde{H}(x^{(k)})$, where $\tilde{H}$ is positive definite, but in some sense close to H. One means of doing so, known as Greenstadt's modification[1], is to compute the eigenvector expansion of H, $H = \sum_{i=1}^{N} \lambda_i v_i v_i^T$, with v_i orthonormal, and modify it slightly so that, for some small $\varepsilon > 0$,

$$\tilde{H} = \sum_{i=1}^{N} \tilde{\lambda}_i v_i v_i^T, \text{ where } \tilde{\lambda}_i \equiv \begin{cases} \lambda_i & \text{if } \lambda_i \geq \varepsilon \\ \varepsilon & \text{if } \lambda_i < \varepsilon \end{cases}.$$

Then the search direction is given by

$$-\tilde{H}^{-1} \nabla S = -\sum_{i=1}^{N} \tilde{\lambda}_i^{-1} [v_i^T \nabla S] v_i.$$

Note that $v_i^T \nabla S$ is a scalar - the dot product of the $i^{\underline{th}}$ normalized eigenvector of H with ∇S.

A rough block diagram of my implementation of the least-squares algorithm is shown in Figure 3. The subroutine DSICO of the LINPACK package [2] is used to factor H, $H = UDU^T$, with U upper triangular and D a diagonal matrix. Using this factorization, DSIDI, also taken from LINPACK, tallies the number of positive, negative, and zero eigenvalues of H. When H is positive definite, the subroutine DSISL of LINPACK then is used to compute the search direction $d_k = -H^{-1}\nabla S(x^{(k)})$.

Whenever DSIDI reports that there are nonpositive eigenvalues, Greenstadt's modification is applied. The subroutine TRED2 of the EISPACK package [3] is used to reduce H to tridiagonal form, via similarity transforms. Then IMTQL2, also of EISPACK, is used to compute the eigensystem by the implicit QL method. The implicit QL algorithm is quite forgiving of bad parameter scaling - say, if the parameters are defined in units in which the rates are orders of magnitude larger than the antenna phases — this leads to eigenvalues of widely varying magnitude. (My program flunked one of its first tests on real data because the scaling was bad and I was using the ordinary QL method). Once the eigensystem has been computed, the search direction is given by $d_k = -\tilde{H}^{-1} \nabla S(x^{(k)})$.

Finally, a coarse search is made in the direction d_k, for an α_k satisfying $S(x^{(k)} + \alpha_k d_k) \leq S(x^{(k)})$. Close enough to a proper local minimum, the choice $\alpha_k = 1$ will do. We set $x^{(k+1)} = x^{(k)} + \alpha_k d_k$, test for convergence, and perform another iteration, if necessary. The theoretically predicted quadratic rate of convergence near a proper local minimum frequently has been observed in the initial tests of the algorithm.

Because the derivatives of the model function are entirely straightforward to compute, the expressions for ∇S and for H have been omitted from this

report.

Preliminary results have shown this method to perform tolerably well, but other algorithms than this would likely do as well or better. In particular, an alternative means of modifying $\hat{H}$ when it is not positive definite could speed up the algorithm significantly, since the eigen-decomposition is relatively expensive. The algorithm could be effectively implemented in an array processor at a moderate expenditure of human effort.

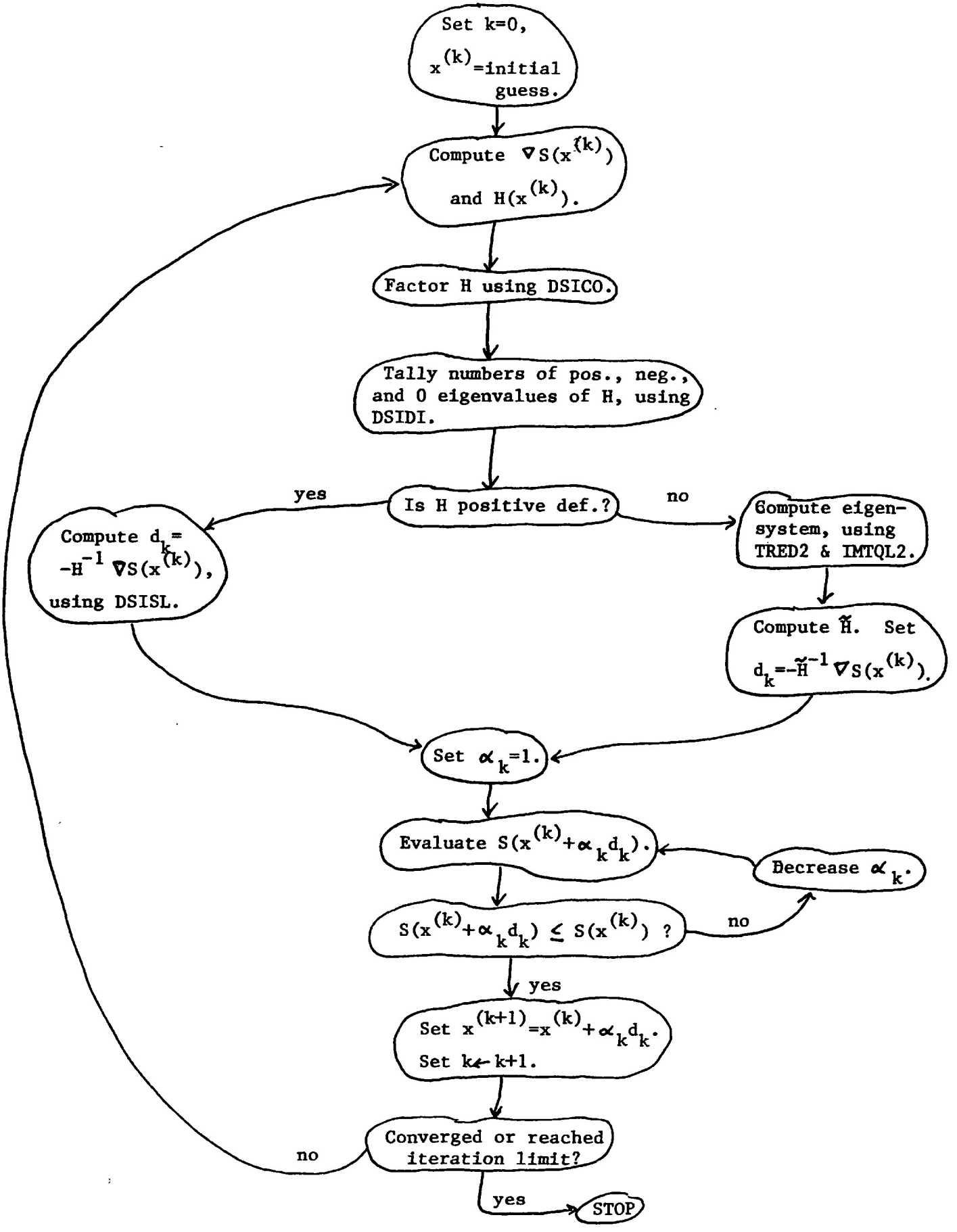

Figure 3. Rough block diagram of the least-squares algorithm.