NUMERICAL CONFORMAL MAPPING AND ANALYTIC CONTINUATION*

By

FREDERIC BISSHOPP

Brown University

Abstract: A numerical method for determination of least-square approximations of an arbitrary complex mapping function is derived here and implemented with fast Fourier transforms (FFTs). An essential feature of the method is the factoring of a discrete Hilbert transform in a pair of Fourier transforms in order to reduce the operation count of the longest computation to $O(N \log N)$. A similar factoring of the discrete Poisson integral formula allows an explicit inversion of it in $O(N \log N)$ operations instead of $O(N^3)$. The resulting scheme for analytic continuation appears to be considerably more reliable than the evaluation of polynomials. Examples are treated, and APL implementations of algorithms are provided.

1. Introduction. Chapter and verse of the book by Carrier, Krook and Pearson ([1], henceforth [CKP]) will be cited for results from the theory of functions of a complex variable. The use of integral equations for finding the analytic function that maps the interior of the unit circle onto a simply connected domain $\mathcal{D}$ is described in [CKP 4-8] and in [2 and 3]. That problem and continuation of the mapping function outside the unit circle will be addressed here by means of somewhat more direct methods.

It has been suggested lately in [4, 5, and 6] that the fast Fourier transform (FFT) can be used to provide efficient algorithms for numerical solution of the mapping problem. Fourier analysis will be used in two ways here: first, to formulate the basic iteration for solution of the mapping problem in terms of a convolution sum; then again to factor that sum and others that are used for evaluation and continuation of the solution in pairs of (fast) Fourier transforms. The arguments are not circular—the second Fourier analysis incorporates twice the number of Fourier coefficients with no significant increase of computational effort beyond what is required in the first.

The problem at hand is to find a complex analytic function $Z(\rho e^{i\theta})$ for which the boundary values at $\rho = 1$ lie on a closed curve that is specified by a complex function $X(s)$. The argument of X is a real parameter (e.g. arc length), and $X(s)$ need not be analytic in any sense. According to the Riemann mapping theorem [CKP 4-2], restated appropriately for the present purpose, there exists a monotonic function $s(\theta)$ and an analytic function $Z(z)$ such that

$$Z(e^{i\theta}) = X(s(\theta)). \tag{1.1}$$

Moreover, $s(\theta)$ is uniquely defined if in addition to (1.1) we specify

i) $\qquad\qquad\qquad\qquad Z(0) = Z_0 \quad$ in $\mathscr{D}$ bounded by X

ii) $\qquad\qquad\qquad\qquad Z(1) = Z_1 \quad$ on X.
$\hfill$ (1.2)

It may be noted (1) that X need not be a simple curve, but if it is not then conformality of the mapping is lost at points where X intersects itself; (2) given $s(\theta)$, Z is uniquely defined for $\rho < 1$ by the Cauchy integral formula [CKP 2-3], restated appropriately as

$$Z(\rho e^{i\theta}) = \frac{1}{2\pi} \int_0^{2\pi} \frac{X(s(\theta'))\, d\theta'}{1 - \rho e^{i(\theta - \theta')}} \qquad (\rho < 1). \tag{1.3}$$

In accord with current nomenclature [4] the function $s(\theta)$ will be called the *correspondence function*. The present method is based upon dropping the specification of Z_0 and Z_1 in (1.2) in favor of determination of approximations of the correspondence function that are *optimal* in the sense of least-square fitting of (1.1). In fact, specified values of Z_0 and Z_1 can be introduced, but the method then becomes more cumbersome and less accurate. It is suggested that this aspect of the mapping problem should be treated by introduction of a bilinear transformation after the optimal approximation of the correspondence function has been found.

The least-square analysis is outlined in Sec. 2, where it is found that a direct Fourier analysis gives an expression for the errors in boundary data (Eq. (1.1)) that can be identified as a rough discretization of one of the Plemelj formulae (Eq. (2.13)). The content of Sec. 3 is an iterative algorithm for determination of $s(\theta)$ that incorporates a refined discretization of the Plemelj formula. In Sec. 4 the convolution sum of Sec. 3 is factored in terms of a pair of discrete Fourier transforms in order to reduce computational effort (and storage) by the use of the fast Fourier transform, and in Sec. 5 evaluation and continuation of the solution are treated by similar techniques. Some examples are treated in Sec. 6, and some of the APL implementations of the algorithms are given in an appendix.

2. Least-square error and Fourier analysis. The mapping function Z is analytic and thus can be represented by the Taylor series

$$Z(\rho e^{i\theta}) = \sum_{k=0}^{\infty} a_k \rho^k e^{ik\theta}, \qquad \rho < 1. \tag{2.1}$$

The complex coefficients a_k and the correspondence function are now to be determined by minimization of

$$E^2 = \lim_{\rho \to 1} \frac{1}{2\pi} \int_0^{2\pi} |Z(\rho e^{i\theta}) - X(s(\theta))|^2 \, d\theta. \tag{2.2}$$

With the substitution of (2.1) in (2.2), $\partial E / \partial a_k = 0$ gives the expected result,

$$a_k = \frac{1}{2\pi} \int_0^{2\pi} e^{-ik\theta} X(s(\theta)) \, d\theta, \tag{2.3}$$

and a vanishing first variation of E^2 when $s(\theta) \leftarrow s(\theta) + \delta s(\theta)$ gives

$$\dot{X}(s(\theta)) \cdot (Z(e^{i\theta}) - X(s(\theta))) = 0. \tag{2.4}$$

The notation above is for the scalar product, $A \cdot B = \text{Re} \{AB^*\}$, and the dot above X indicates d/ds.

It will be assumed (for convenience) that $X(s)$ is twice differentiable, for then an iteration of (2.4) is easily defined by Newton's method. With a current approximation of $Z(e^{i\theta})$, $s(\theta)$ is updated according to

$$s(\theta) \leftarrow s(\theta) - \frac{\dot{X}(s(\theta)) \cdot (X(s(\theta)) - Z(e^{i\theta}))}{|\ddot{X} \cdot (X - Z) + |\dot{X}|^2|}. \tag{2.5}$$

The magnitude in the denominator insures iteration toward a *minimum* of E^2. It may be noted that θ appears as a parameter in (2.5), so a discretization of θ and s in $2N$ intervals gives $2N$ decoupled updates which implies $O(2N)$ operations per iteration.

The substitution of (2.3) in (2.1) implies

$$Z(\rho e^{i\theta}) = \frac{1}{2\pi} \sum_{0}^{\infty} \rho^k \int_{0}^{2\pi} X(s(\theta'))e^{ik(\theta - \theta')} \, d\theta', \tag{2.6}$$

and for $\rho < 1$, integration and summation (of the geometric series) can be interchanged to give

$$Z(\rho e^{i\theta}) = \frac{1}{2\pi} \int_{0}^{2\pi} \frac{X(s(\theta')) \, d\theta'}{1 - \rho e^{i(\theta - \theta')}}, \qquad \rho < 1. \tag{2.7}$$

Thus we have it that the Cauchy integral formula evaluates a least-square approximation of Z when $s(\theta)$ is not the correspondence function.

In the case where $\rho = 1$, summation and integration cannot be interchanged in (2.6), and it becomes necessary to specify a relationship between truncation of (2.1) and discretization of (2.3). Let the Taylor series be truncated to a polynomial of degree M, and let $[0, 2\pi]$ be discretized in $2N$ equal intervals. Then the Gaussian quadrature formula for (2.3) is

$$a_k \approx \frac{1}{2N} \sum_{l=0}^{2N-1} e^{-ik\theta_l} X(s_l) \tag{2.8}$$

where θ_l is $l\pi/N$ and s_l is $s(\theta_l)$. The finite sums can be interchanged, and thus the approximate form of (2.6) implies

$$Z(e^{i\theta_m}) \approx \frac{1}{2N} \sum_{l=0}^{2N-1} K(\theta_m - \theta_l) X(s_l) \tag{2.9}$$

where

$$K(\theta_j) = \sum_{k=0}^{M} e^{ik\theta_j} = \frac{1 - e^{i(M+1)\theta_j}}{1 - e^{i\theta_j}}. \tag{2.10}$$

The fact that a_k and a_{k-2N} cannot be distinguished in the discretized approximation (2.8) suggests that M should be N or $N - 1$, rather than $2N - 1$. Indeed, if M is $2N - 1$ then $K(\theta_j) = 0$, except for $K(0) = 2N$, and (2.9) implies the approximation

$$Z(e^{i\theta_m}) = X(s_m) \qquad (M = 2N - 1). \tag{2.11}$$

Here the discrete version of (2.2) is minimized with $E^2 = 0$, but no account is taken of the requirement that Z shall be analytic.

With $M = N - 1$, Eq. (2.10) implies

$$K(\theta_j) = \begin{cases} N & j = 0 \\ \dfrac{2}{1 - e^{i\theta_j}} & j = 1, 3, 5, \ldots \\ 0 & j = 2, 4, \ldots. \end{cases} \tag{2.12}$$

The factor of 2 in the second line compensates for evaluations of the integrand at half of the points, and (2.9) now turns out to be a numerical quadrature of the Plemelj formula,

$$Z(e^{i\theta}) = \tfrac{1}{2}\left(X(s(\theta)) + \frac{1}{\pi} P \int_0^{2\pi} \frac{X(s(\theta'))\, d\theta'}{1 - e^{i(\theta - \theta')}} \right), \tag{2.13}$$

where P denotes the principal value integral [CKP 8-6].

With $M = N$, Eq. (2.10) implies

$$K(\theta_j) = \begin{cases} N + 1 & j = 0 \\ i \cot \dfrac{\theta_j}{2} & j = 1, 3, 5, \ldots \\ 1 & j = 2, 4, \ldots. \end{cases} \tag{2.14}$$

In fact,

$$\frac{1}{1 - e^{i(\theta - \theta')}} = \frac{1}{2}\left(1 + i \cot \frac{\theta - \theta'}{2} \right), \tag{2.15}$$

and again (2.9) is a numerical quadrature of (2.13).

In sum, it has been established in this section:
  i) Boundary data for the least-square fit of the mapping function can be computed, either by Fourier analysis or with the Plemelj formula.
  ii) If $s(\theta)$ is not the correspondence function, then the mean-square error of the approximation can be reduced by an iterative updating of $s(\theta)$ based on (2.4).

3. The correspondence function by iteration. In this section the Euler equation (2.4) will be used to formulate an iteration for determination of the least-square approximation of the correspondence function $s(\theta)$. The Fourier analysis in Sec. 2 will be discarded in favor of the convolution integrals (2.7 and 2.13) for the evaluation of least-square approximations of the mapping function and its boundary values at $\rho = 1$.

Before a discretization is introduced it is convenient to remove the singularity at $\theta = \theta'$ that appears in (2.7) in the limit where $\rho \to 1$, or, equivalently, to convert the principal value integral in (2.13) to an ordinary integral of a bounded function. Following others here [2], we note that '1' is an analytic function; thus

$$1 = \frac{1}{2\pi} \int_0^{2\pi} \frac{d\theta'}{1 - \rho e^{i(\theta - \theta')}} \qquad (\rho < 1) \tag{3.1}$$

and

$$1 = \frac{1}{2}\left(1 + \frac{1}{\pi} P \int_0^{2\pi} \frac{d\theta'}{1 - e^{i(\theta - \theta')}} \right). \tag{3.2}$$

Either of the above equations can be multiplied by $X(s(\theta))$ and used with the correspond-

ing one of (2.7 and 2.13) to obtain (in the limit where $\rho \to 1$ or directly)

$$
\begin{aligned}
Z(e^{i\theta}) - X(s(\theta)) &= \frac{1}{2\pi} \int_0^{2\pi} \frac{X(s(\theta')) - X(s(\theta))}{1 - e^{i(\theta - \theta')}} \, d\theta' \\
&= \frac{1}{4\pi} \int_0^{2\pi} \left(1 - i \cot \frac{\theta' - \theta}{2} \right) (X' - X) \, d\theta'.
\end{aligned}
\tag{3.3}
$$

Provided X (to be assumed a C^2 function of s for convenience) is C^1, contributions to quadrature formulae at $\theta = \theta'$ can be evaluated by l'Hospital's rule and the rough quadrature formulae of Sec. 2 can be replaced by a refined one in which the integrand is evaluated at every point of the discretization.

In order to define the iteration for the Nth approximation of the correspondence function and the boundary values of the mapping function, the range $[0, 2\pi]$ of the independent variable θ is discretized in $2N$ equal intervals bounded by

$$
\theta_k = k\pi/N
\tag{3.4}
$$

and the current approximation of $s(\theta)$, evaluated at the multiples of π/N, is denoted by

$$
s_k = s(\theta_k), \qquad s_{k+2N} = s_k + S.
\tag{3.5}
$$

Given a current approximation of s_k, the error is evaluated from

$$
Z_l - X_l = \tfrac{1}{2}(\bar{X} - X_l) - \frac{1}{4\pi}(X_{l+1} - X_{l-1}) - \frac{i}{4N} \sum_{k=1}^{N-1} (X_{l+k} - X_{l-k}) \cot \frac{\theta_k}{2}
\tag{3.6}
$$

where X_l is $X(s_l)$, Z_l is the current least-square approximation of $Z(e^{i\theta_l})$, and

$$
\bar{X} = \frac{1}{2N} \sum_{k=0}^{2N-1} X_k.
\tag{3.7}
$$

The special contribution to (3.3) at $\theta = \theta'$ has been approximated by substitution of a central difference for $(dX/d\theta)_l$. Since X is periodic, the subscripts on $X_{l\pm k}$ are to be reckoned mod $2N$.

Given the current error (3.6), s_k is updated by application of Newton's method to the $2N$ conditions

$$
\dot{X}_k \cdot (Z_k - X_k) = 0
\tag{3.8}
$$

where (again) $A \cdot B \equiv \text{Re } \{A^* B\}$ and $\dot{X}$ denotes dX/ds. In each iteration Z_k is held fixed and s_k is updated with one pass of Newton's method as

$$
s_k \leftarrow s_k + \delta s_k \leftarrow \frac{\dot{X}_k \cdot (Z_k - X_k)}{||\dot{X}_k|^2 - \ddot{X}_k \cdot (Z_k - X_k)|}.
\tag{3.9}
$$

Here, depending on the magnitudes $|\delta s_k|$, the iteration is either terminated or repeated starting at Eqs. (3.6), and (3.7).

The computational effort (per iteration) for the algorithm given above is $O(N)$ operations for the updates (3.9) and $O(N^2)$ operations for the convolution sums (3.6). In the next section it will be shown how the convolution sums can be done in $O(N \log N)$ operations with FFTs. Before that, a few incidental comments can be appended here:

1) In practice it was found to be necessary to limit the displacements (one-third of the widths of the neighboring intervals was used) in order to preserve monotonicity of $s(\theta)$ in early stages of the iteration.

2) Convergence rates (number of iterations) are not particularly impressive, but they can be improved by replacement of Newton's method by a conjugate gradient method [7]. Interpolation of a lower-order approximation to provide starting values for a higher approximation is less helpful here than in some other problems because $s(\theta)$, though monotonic, can have a large curvature in some regions (Fig. 6).

3) If, after all, it is insisted that the map of $\rho = 0$ shall be specified as $Z_0 \in \mathcal{D}$ bounded by X, that can be incorporated (with loss of accuracy) by replacement of Eq. (3.7) by

$$\bar{X} = Z_0. \tag{3.7a}$$

Again, it is my opinion that it is preferable to find the optimal mapping function $Z(z)$ that maps $|z| = 1$ onto X and the inverse image z_0 such that $Z(z_0) = Z_0$. Then $Z(z(\rho e^{i\theta}))$ maps $\rho = 0$ to Z_0 and $\rho = 1$ onto X when z is the bilinear transformation

$$z = \frac{z_0 + \rho e^{i\theta}}{1 + z_0^* \rho e^{i\theta}}. \tag{3.10}$$

4. Fourier analysis of the Hilbert integral formula. Because of its similarity to the equations that define a Hilbert transform pair [5 and CKP 8-6] and for want of a better name, Eq. (3.3, 2nd line) will be called the Hilbert integral formula. It appears here in an unfamiliar way, however, since neither the real parts nor the imaginary parts of the boundary values of the mapping function are known until the correspondence function has been found. What is called for is merely an evaluation of the error in a least-square approximation of the mapping function that is given in terms of real and imaginary parts of X that are specified, but not consistently. This can be done in $O(N \log N)$ operations, as follows.

The identity used in factoring the Hilbert integral formula is the same as is used in the derivation of the Poisson kernel [CKP 2-5]:

$$\frac{1}{1 - \rho e^{i\theta}} + \frac{1}{1 - \frac{1}{\rho} e^{i\theta}} = 1 + \frac{2i\rho \sin \theta}{1 + \rho^2 - 2\rho \cos \theta}. \tag{4.1}$$

This has the limit, where $\rho \to 1$,

$$1 + i \cot \frac{\theta}{2} = \lim_{\rho \to 1} \left(\frac{1}{1 - \rho e^{i\theta}} + \frac{1}{1 - \frac{1}{\rho} e^{i\theta}} \right) \quad (\theta \neq 0). \tag{4.2}$$

The substitution of (4.2) in the discretization of (3.3) in $2N$ equal intervals gives a rearrangement of (3.6) as

$$Z_l - X_l = -\frac{i}{4\pi} (X_{l+1} - X_{l-1}) + \sum_{k=0}^{2N-1} H_{lk}(X_k - X_l), \tag{4.3}$$

where $H_{ll} = 0$, and otherwise

$$H_{lk} = \frac{1}{4N} \lim_{\rho \to 1} \left(\frac{1}{1 - \rho e^{i(\theta_l - \theta_k)}} + \frac{1}{1 - \frac{1}{\rho} e^{i(\theta_l - \theta_k)}} \right) \tag{4.4a}$$

$$= \frac{1}{4N} \lim_{\rho \to 1} \left(\frac{\sum_{j=0}^{2N-1} \rho^j e^{ij(\theta_l - \theta_k)}}{1 - \rho^{2N}} + \frac{\sum_{j=0}^{2N-1} \rho^{-j} e^{ij(\theta_l - \theta_k)}}{1 - \rho^{-2N}} \right) \tag{4.4b}$$

$$= \frac{1}{4N} \sum_{j=0}^{2N-1} \left(1 - \frac{j}{N} \right) e^{ij(\theta_l - \theta_k)}. \tag{4.4c}$$

Since the coefficient of H_{ll} is zero anyway, it can be included, as given in (4.4c), to give

$$Z_l - X_l = -\frac{i}{4\pi} (X_{l+1} - X_{l-1}) + \frac{1}{4N} \sum_{j=0}^{2N-1} \left(1 - \frac{j}{N} \right) e^{ij\theta_l} \sum_{k=0}^{2N-1} (X_k - X_l) e^{-ij\theta_k}. \tag{4.5}$$

From the properties of finite Fourier transforms (Eq. (2.10) with $M = 2N - 1$) it follows that

$$(X_k - X_l) \sum_{j=0}^{2N-1} e^{ij(\theta_l - \theta_k)} = 0 \tag{4.6}$$

and

$$jX_l \sum_{k=0}^{2N-1} e^{-ij\theta_k} = jX_l \sum_{k=0}^{2N-1} e^{-ik\theta_j} = 0. \tag{4.7}$$

Thus the final, factored version of the discrete Hilbert formula is

$$Z_l - X_l = \frac{-i}{4\pi} (X_{l+1} - X_{l-1}) - \frac{1}{4N^2} \sum_{j=1}^{2N-1} je^{ij\theta_l} \sum_{k=0}^{2N-1} X_k e^{-ij\theta_k}. \tag{4.8}$$

If $s(\theta)$ is the correspondence function, then, in the limit where $N \to \infty$, (4.8) converges to the polar form of the Cauchy-Riemann equations,

$$\lim_{\rho \to 1} \left(i\rho \frac{\partial Z}{\partial \rho} \right) = \frac{dX}{d\theta}. \tag{4.9}$$

For another check on algebra, the algorithm for determination of $s(\theta)$ was implemented in both ways ((3.6) and (4.8)); identical results were obtained. Even when $2N$ is a relatively low power of 4, the algorithm that incorporates (4.8) and uses FFTs is significantly more efficient.

5. Fourier analysis of the Poisson integral formula. Once a satisfactory approximation of the correspondence function has been found, there comes the problem of evaluation of the mapping function for $\rho \leq 1$ and (sometimes) continuation, for $\rho > 1$. Since it is a byproduct of the iteration for $s(\theta)$, the discrete Fourier transform of X will be used in both cases; i.e. we take as given

$$a_k = \frac{1}{\sqrt{2N}} \sum_0^{2N-1} X_l e^{-ik\theta_l}, \qquad k = 0 \text{ to } 2N - 1, \tag{5.1}$$

where $X_l = X(s(\theta_l))$ and $\theta_l = l\pi/N$.

For the evaluation of the mapping function at $\rho \leq 1$, there is little to be gained by the
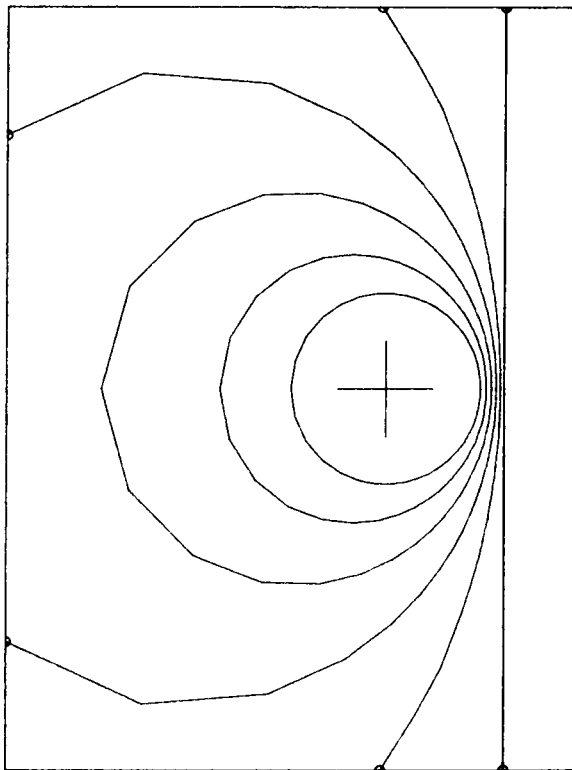
FIG 1. Zero at $-.5$, pole at -2. $r = 1, 1.2, 1.4, 1.6, 1.8, 2$.

use of anything but the polynomial

$$Z(\rho e^{i\theta}) = \frac{1}{\sqrt{2N}} \sum_{k=0}^{2N-1} a_k \rho^k e^{ik\theta}, \qquad (5.2)$$

and that has been done in all cases. A map of a level curve (ρ is constant) at the points $Z_k(\rho, \delta) = Z(\rho e^{i(\theta_k + \delta)})$ can be evaluated from (5.2) in $O(N \log N)$ operations. When that is done, it is found that the maps of level lines near the boundary are poorly represented. What happens is that $s(\theta)$, though monotonic, can still vary quite rapidly; and, if s is arc length on X, large and small values of $s'(\theta)$ correspond to small and large densities of mapped points along $Z_k(1, \delta)$. The phenomenon is not necessarily an indication of an inadequate approximation; indeed, an evaluation of the error in (5.2) at $\rho = 1$ for a few interpolated angles where $s'(\theta)$ is large serves well as a check on whether or not enough Fourier coefficients have been included in (5.1).

When the polynomial (5.2) is evaluated at θ_l for $\rho > 1$, reliable results can be expected at moderate values of ρ, provided the mapping function has a relatively distant nearest singularity. For larger values of ρ, (5.2) produces numerical trash, thus indicating, in part, the encounter with the nearest singularity of Z and, in part, the truncation error of the discrete Fourier transforms. In this section the Poisson integral formula [CKP 2-5] will used to derive a different continuation scheme that appears (in simple examples) to be more reliable than the polynomial.
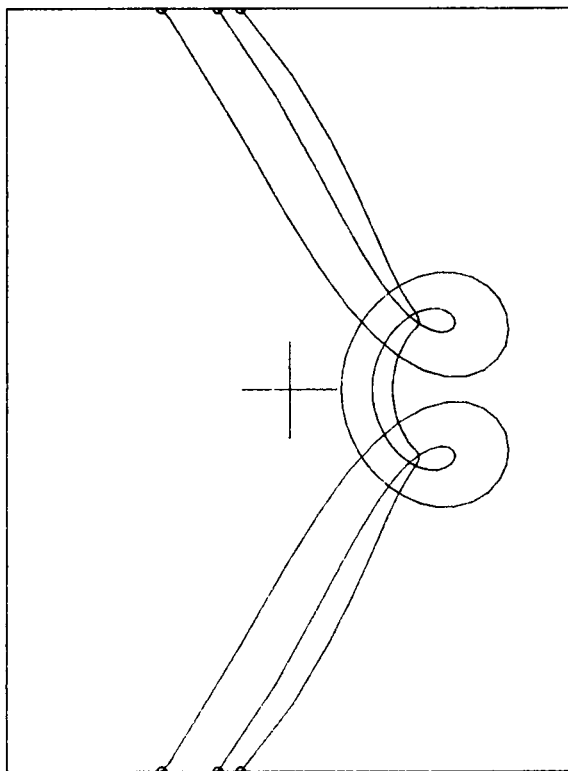
FIG. 2. Zero at $-.5$, pole at -2. Polynomial at 1.78, 1.81, 1.83.

Given a mapping function Z that is analytic in a circle of radius ρ', the Poisson integral formula is

$$Z(\rho e^{i\theta}) = \frac{1 - \left(\dfrac{\rho}{\rho'}\right)^2}{2\pi} \int_0^{2\pi} \frac{Z(\rho' e^{i\theta'})\, d\theta'}{1 + \left(\dfrac{\rho}{\rho'}\right)^2 - 2\dfrac{\rho}{\rho'}\cos(\theta - \theta')} \tag{5.3}$$

where $\rho < \rho'$. The formula is often derived from the identity

$$\frac{1}{1 - \dfrac{\rho}{\rho'}e^{i(\theta - \theta')}} - \frac{1}{1 - \dfrac{\rho'}{\rho}e^{i(\theta - \theta')}} = \frac{1 - \left(\dfrac{\rho}{\rho'}\right)^2}{1 + \left(\dfrac{\rho}{\rho'}\right)^2 - 2\dfrac{\rho}{\rho'}\cos(\theta - \theta')} \tag{5.4}$$

and this will be used, as in Sec. 4.

Again let $\theta_k = k\pi/N$ and let $Z_k = Z(\rho e^{i\theta_k})$ and $Z_k' = Z(\rho' e^{i\theta_k})$: then the discretization of (5.3, 5.4) is
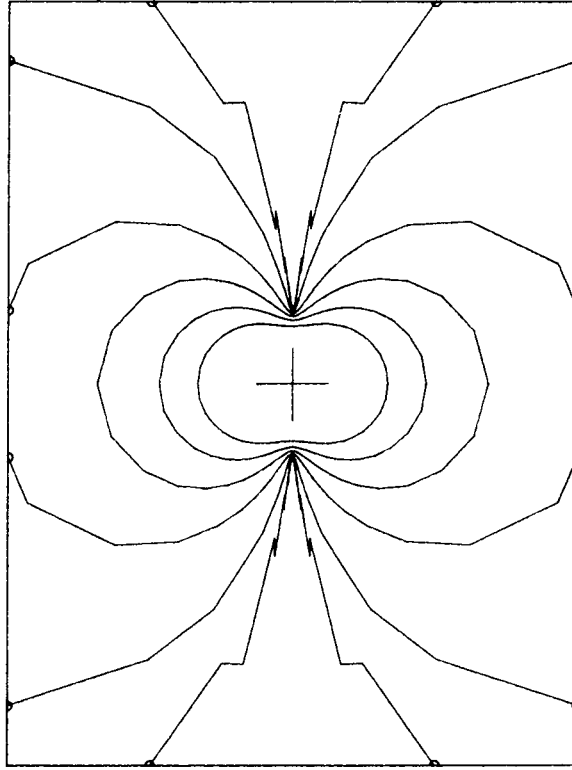
$$Z_l = \sum_{k=0}^{2N-1} P_{lk} Z_k', \tag{5.5}$$

FIG. 3. Zero at 0, poles at 2 and -2. $r = 1, 1.2, 1.4, 1.6, 1.8, 1.9$.

where

$$P_{lk} = \frac{1}{2N} \left[\frac{\displaystyle\sum_{j=0}^{2N-1} \left(\frac{\rho}{\rho'}\right)^{j} e^{ij(\theta_l - \theta_k)}}{1 - \left(\dfrac{\rho}{\rho'}\right)^{2N}} - \frac{\displaystyle\sum_{j=0}^{2N-1} \left(\frac{\rho'}{\rho}\right)^{j} e^{ij(\theta_l - \theta_k)}}{1 - \left(\dfrac{\rho'}{\rho}\right)^{2N}} \right] \tag{5.6a}$$

$$= \frac{1}{2N} \frac{\displaystyle\sum_{j=0}^{2N-1} (\rho^j \rho'^{2N-j} + \rho'^j \rho^{2N-j}) e^{ij(\theta_l - \theta_k)}}{\rho'^{2N} - \rho^{2N}}. \tag{5.6b}$$

The substitution of (5.6b) in (5.5) gives a pair of discrete Fourier transforms that can be inverted explicitly (and exactly) to give

$$Z'_l = \frac{(\rho'^{2N} - \rho^{2N})}{2N} \sum_{j=0}^{2N-1} \frac{e^{ij\theta_l}}{\rho^j \rho'^{2N-j} + \rho'^j \rho^{2N-j}} \sum_{k=0}^{2N-1} Z_k e^{-ij\theta_k}. \tag{5.7}$$

Now let $\rho = 1$ and $Z_k = X_k$, and then drop the prime on ρ' to obtain the formal result,

$$Z'_l = \frac{\rho^N - \rho^{-N}}{\sqrt{2N}} \sum_{k=0}^{2N-1} \frac{a_k}{\rho^{N-k} + \rho^{k-N}} e^{ik\theta_l} \tag{5.8}$$
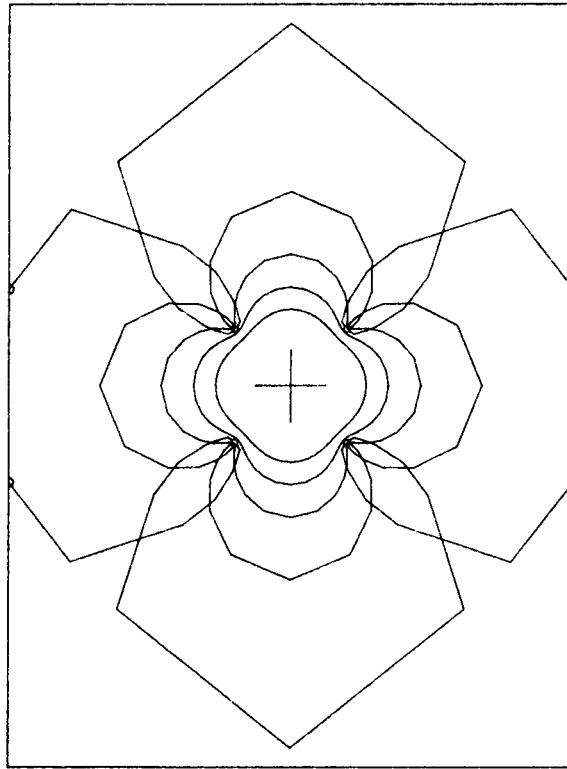
FIG 4. Zero at 0, poles at 2, 2i, -2, $-2i$. $r = 1, 1.2, 1.4, 1.6, 1.8$.

for ρ (formerly ρ') greater than 1. The formula (5.8) is blatantly wrong at $\rho = 1$, and that is undoubtedly due to the neglect of the singularity in (5.3) at $\rho = \rho'$ and $\theta = \theta'$. In lieu of a more detailed analysis of (5.3), it is proposed now that (5.8) simply be replaced by

$$Z_l = \frac{\rho^N + \rho^{-N}}{\sqrt{2N}} \sum_{k=0}^{2N-1} \frac{a_k}{\rho^{N-k} + \rho^{k-N}} e^{ik\theta_l} \tag{5.9}$$

for "continued" values of $Z(\rho e^{i\theta_l})$ when $\rho \geq 1$. In the sense of boundary layer theory, (5.9) is a *uniformly valid approximation*, i.e.:

i) Boundary data is recovered at $\rho = 1$,
ii) (5.8) is recovered outside a boundary layer of width $O(1/N)$ at $\rho = 1$,
iii) (5.2) is recovered, term by term, for fixed k and fixed $\rho > 1$, when $N \to \infty$.

As will be seen presently, the filtering of high harmonics that has been obtained in (5.9) has a soothing effect in some cases where it is certain that a numerical analytic continuation should be possible.

6. Examples. Numerical analytic continuation of maps defined by rational functions will be considered first. The APL functions $\nabla G \leftarrow Y$ *POLS* $F\nabla$ and $\nabla G \leftarrow Y$ *ZERS* $F\nabla$ were used to generate boundary data at $\rho = 1$. Thus, for example,

$$Z \leftarrow {}^{-}.5 \quad 0 \quad ZERS \quad {}^{-}2 \quad 0 \quad POLS \quad 130 \quad \rho 1 \quad 0$$
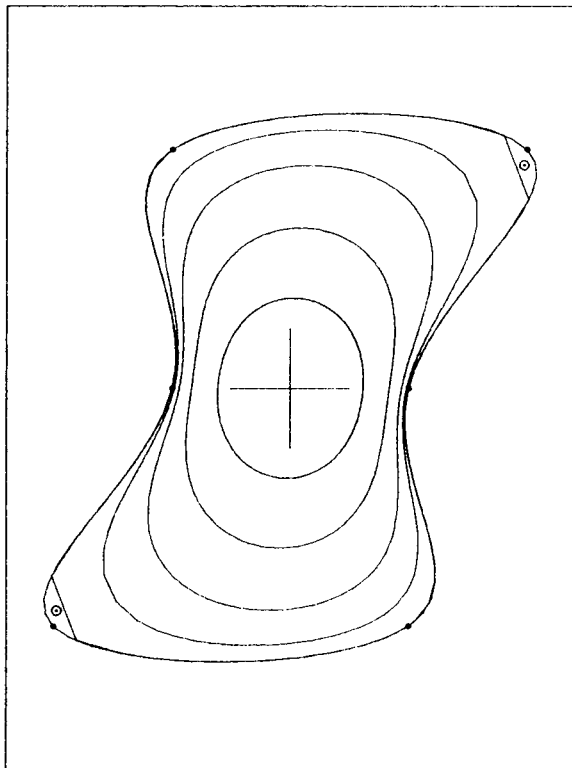
FIG. 5. Knots, data, maps at .5, .75, .9, .97, 1, interpolated at 37–38 and 165–166.

assigns sixty-five coordinate pairs X, Y (last pair equals the first to close the curve) on $Z = (0.5 + e^{i\theta})/1 + 0.5 \times e^{i\theta}$ at the 64th roots of unity. Fourier coefficients were evaluated by $\nabla F \leftarrow FFT4\ Z\nabla$ and then evaluations and continuations were given by $\nabla Z \leftarrow R$ $FIFT4\ F\nabla$. Thus with Z assigned as above

$$F \leftarrow FFT4\quad Z$$

assigns the Fourier coefficients in a 2 by 64 array, and, for example,

$$G \leftarrow 0.5\quad 1\quad 1.5\quad FIFT4\quad F$$

assigns the points on the level lines in a 3 by 130 array. The decision to limit the implementations of FFT's to cases where the number of coefficients is a power of *four* helped to make them reasonably efficient: an execution of what has been mentioned so far in this section was clocked at 565 milliseconds CPU time on the IBM 4341 at Brown University.

Some maps of continued level lines at 64 roots of ρ are shown in Figs. 1–4. All the mapping functions had one or more poles at $\rho = 2$ and, to be fair, it should be mentioned that the Poisson continuations, while qualitatively correct, tended to become rather noisy at values of ρ between 1.9 and 2. Fig. 1 is a bilinear transformation (Eq. (3.10) with $z_0 = 0.5$) that maps $\rho = 0$ to $z = 0.5$ and $\rho = 2$ to a circle of infinite radius. Scale and orientation are indicated by the crosshair, which is centered at $z = 0$ and has a unit diameter. In Fig. 2, level lines of a polynomial approximation of the same map are shown for values of ρ at which the 64-term polynomial begins to show qualitative discrepancies.

FIG. 6. Correspondence function for Fig. 5.

For Fig. 2, which indicates that the antipole is as troublesome for the polynomial as the pole, it was necessary to doctor the transform of the boundary data in order to control round-off errors. The transform of the symmetric figure had round-off errors of $O(10^{-15})$ in its imaginary part, and before these were reassigned exact zeros, the polynomial gave results that were similar to Fig. 2, but not symmetric. Fig. 3 shows continuations by the Poisson formula for a map that has one zero and two poles, and Fig. 4 does the same for a map that has one zero and four poles.

For better or worse, the algorithm for conformal mapping and the correspondence function was set up to take as input the coefficients of a B-spline representation of the boundary curve. This was well suited to the use of Newton's method in the iteration and made construction of complicated boundaries easy, but gave results that cannot be continued much beyond $\rho = 1$ because of the singularities at the knots. The APL function $\nabla B \leftarrow PFIT\ X \nabla$ returns coefficients of B-splines defined on $[-2, 2]$ and centered at any number of knots, as specified by X. The function $\nabla F \leftarrow N\ BDFT\ B \nabla$ then performs the iteration of Sec. 3. Thus, for example,

$$B \leftarrow 64 \quad BDFT \quad A \leftarrow PFIT \quad 1 \quad 0 \quad 0 \quad 1 \quad ^-1 \quad 0 \quad 0 \quad ^-1$$

assigns to A a 2 by 4 array of coefficients for a spline with period 4 that is near a unit circle and to B a 3 by 64 array of Fourier coefficients and the correspondence function.

A more complicated boundary was constructed and approximations were carried to 256 roots of one to generate the example in Fig. 5. As can be seen from the marked
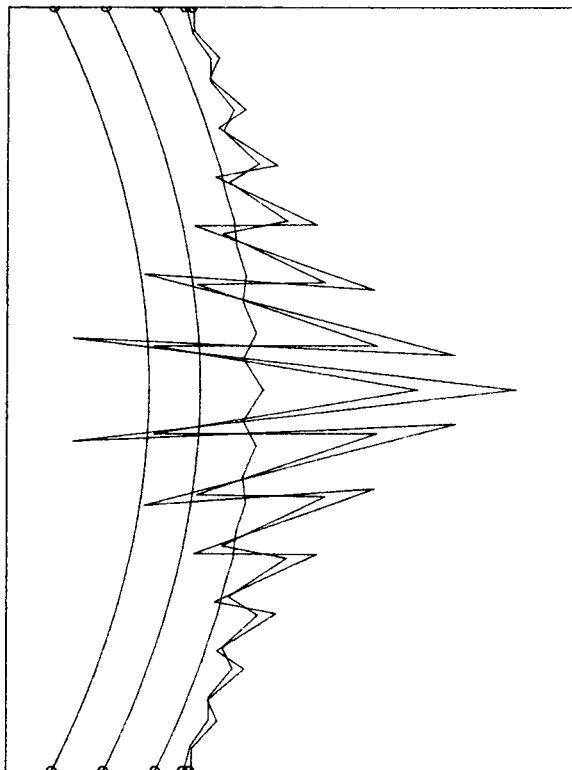
FIG. 7. $r = 1, 1.05, 1.1, 1.125, 1.13$, near a knot.

evaluations at a few 512th roots of one, the approximation is still off by a bit near the ends of two rays. Fig. 6 is the correspondence function for the same map, evaluated at 256th roots of one. With tolerance on δs set at 10^{-6}, this computation ran through 200 iterations and took 3 minutes CPU time—just about the limit of terminal boredom for interactive users.

Finally, the approximation for the nearly circular boundary was carried to 256th roots of one and a few 'continuations' were plotted on an expanded scale to show the rather abrupt breakdown of the Poisson continuation formula near a discontinuity in the third tangential derivative of boundary data.

It can be said now, on the basis of some experience with these algorithms, that they probably should not be exercised much more strenuously than they have been here. Even with the filtering that comes with the use of the Poisson continuation formula, the attempt to improve accuracy by increasing N fails when $\varepsilon\rho^N \approx 1$. In APL's double precision calculations $\varepsilon \approx 10^{-15}$, and for $\rho \approx 2$ this gives $N \approx 50$. When $2N$ was increased from 64, as in Figs. 1–4, to 256 the results deteriorated rapidly with increasing ρ, and only a moderate improvement was achieved by the assignment of exact zeros to small coefficients that was incorporated in the last version of $\nabla F \leftarrow FFT4\,Z\nabla$.

REFERENCES

[1] G. F. Carrier, M. Krook, and C. E. Pearson, *Functions of a complex variable*, McGraw-Hill, 1966

[2] Chr. Anderson et al., *Conformal mapping* (Chap. III) *Selected numerical methods*, ed. Christian Gram, Regnecentralen, Copenhagen, 1962

[3] D. Gaier, *Konstructive Methoden der konformen Abbilding*, Springer, 1964

[4] P. Henrici, *Fast Fourier methods in computational complex analysis*, SIAM Review 21, 481–527 (1979)

[5] D. I. Meiron and S. A. Orszag, *Applications of numerical conformal mapping*, J. Comp. Phys. 40, 345–359 (1981)

[6] B. Fornberg, *A numerical method for conformal mappings*, SIAM J. Sci. Stat. Comp. 1, 386–400 (1980)

[7] W. Murray (ed.), *Numerical methods for unconstrained optimization*, Academic Press (1972)

Appendix: APL implementations.

```
F←FFT4 Z;□IO;N;L;C;S;C4;S4;I;J;CJ;SJ;U;V
ค    FAST FOURIER TRF OF Z[ı2;ıN=4*L], INVERSE IS FIFT4
ค    RESHAPE ALTERNATE INPUT Z[ı2N+2]
→(2=ρρZ)ρB1
Z←⍉((⁻1+0.5×ρZ),2)ρZ
B1:→(L≠⌊L+4●N←(ρZ)[1+□IO←0])ρERR
C←(N÷4)⌽S←0,S,1,(⌽S),0,(-S),⁻1,-⌽S←1○(1+ıN÷4)×○2÷N
S4←(0 1 0 ⁻1)[S4], 4 0 ρC4←(1 0 ⁻1 0)[S4+4|(ı4)○.×ı4]
U←Z[0;],0ρV←Z[I+1;]
→B2,F←4,J←N÷4
ERR:→S∆FFT4←ERR
B2:U←,⍉(C4+.×Z←FρU)+S4+.×V←FρV
→(J=1)ρE,V←,⍉(C4+.×V)-S4+.×Z
U←((CJ←C[CJ])×Z+U)+V×SJ←S[CJ+N|,(ıJ)○.×(I+I×4)ρI×ı4]
→B2,(J←J÷4),V←(CJ×V)-SJ×Z
E:F[((□CT>|F)/ıρF←(÷N*÷2)×(,⍉LρU),,⍉(L+Lρ4)ρV]←0
F←(2,N)ρF
```



```
Z←R FIFT4 F;□IO;N;L;C;S;C4;S4;G;P;U;V;W;I;J;K;SJ;CJ
ค    FILTERED INVERSE (FAST) FOURIER TRANSFORM
ค    F[ı2;ıN=4*L] IS FOURIER TRF OF MAP OF UNIT CIRCLE
ค    Z[ıρR;ı2N+2] IS X,Y...X,Y ON RHO=R AT N ROOTS OF ONE
ค    EVALUATE POLYNOMIAL OF DEGREE N-1 FOR R≤1
ค    INVERSE POISSON INTEGRAL FOR R>1
ค    CHECK N=4*L AND SETUP XERS FOR FFT'S
→(L≠⌊L+4●N←(ρF)[1+□IO←0])ρERR
C←(N÷4)⌽S←0,S,1,(⌽S),0,(-S),⁻1,-⌽S←1○(1+ıN÷4)×○2÷N
S4←(ρC4← 4 4 ρ(1 0 ⁻1 0)[S4])ρ(0 1 0 ⁻1)[S4+4|,(ı4)○.×ı4]
→B1,(,Z←(0,2×N)ρR←,R),(F←F[0;]),(G←F[1;]),(L←Lρ4),K←4,N÷4
ERR:→S∆FIFT4←ERR
ค    CHECK R≤1 OR R>1   (R<0 → POWER SERIES)
B1:→(1<P←R[0])ρB2
→B3,P←1,×\(N-1)ρP
B2:P←P,1+⌽1+P←P[0]÷P←⌽2,P+÷P←×\(N÷2)ρP
ค    INVERSE FFT FOR EVALUATION OR CONTINUATION
B3:→0ρ(J←N÷4),(I←1),(U←F×P),V←G×P
B4:→(J=1)ρE4,(V←,⍉(C4+.×V)+S4+.×W),U←,⍉(C4+.×W←KρU)-S4+.×V←KρV
SJ←S[CJ←N|,(ıJ)○.×(I+4×I)ρI×ı4]
→B4,(J←J÷4),(V←(CJ×V)+SJ×W),U←((CJ←C[CJ])×W+U)-SJ×V
E4:Z←Z,[0],(÷N*÷2)×(,⍉LρU),[0.5],⍉LρV
→(0<ρR←1+R)ρB1
→(1<1↑ρZ←Z,Z[;ı2])ρ0
Z←,Z
```

```
G←Y POLS F;⎕IO;Z;X;N
 ⍝  F[⍳2N+2] IS (F(Z) AT N ROOTS OF ONE),F(1)
 ⍝  G[⍳2N+2] IS F÷(×/(Z-ZL))×÷/1-Z÷ZG FOR |ZL|<1 AND |ZG|>1
 ⍝  Y[⍳2M] IS M POLES,  ZL OR ZG
Z← 2 1 ⍉.○(⍳N)×○2÷N÷(ρF÷�((‾1+÷2÷ρF),2)ρF)[1+⎕IO←0]
B1:→(1<G←+/X×X÷2ρY)ρB2
→B3,X←(Z[0;]-X[0]),Z[1;]-X[1]
B2:X←(1-(Z[0;]×X[0])+Z[1;]×X[1]),(Z[0;]×X[1])-Z[1;]×(X÷X÷G)[0]
B3:F←(2,N)ρ((2×N)ρ+↑/⍉X×X)×(+/⍝F×X),-/⍝(X÷(2,N)ρX)×⍉F
→(0<ρY÷2÷Y)ρB1
G←G,2↑G←,⍝F



G←Y ZERS F;⎕IO;Z;X;N
 ⍝  F[⍳2N+2] IS (F(Z) AT N ROOTS OF ONE),F(1)
 ⍝  G[⍳2N+2] IS F×(×/(Z-ZL))×÷/1-Z÷ZG FOR |ZL|<1 AND |ZG|>1
 ⍝  Y[⍳2M] IS M ZEROS, ZL OR ZG
Z← 2 1 ⍉.○(⍳N)×○2÷N÷(ρF÷⍝((‾1+÷2÷ρF),2)ρF)[1+⎕IO←0]
B1:→(1<G←+/X×X÷2ρY)ρB2
→B3,X←(Z[0;]-X[0]),Z[1;]-X[1]
B2:X←(1-(Z[0;]×X[0])+Z[1;]×X[1]),(Z[0;]×X[1])-Z[1;]×(X÷X÷G)[0]
B3:F←(2,N)ρ(-/⍝F×X),+/⍝F×⍉X÷(2,N)ρX
→(0<ρY÷2÷Y)ρB1
G←G,2↑G←,⍝F



Z←T IIFT F;S;C;N;⎕IO
 ⍝  INTERPOLATE INVERSE (SLOW) FOURIER TRF
 ⍝  F[⍳2;⍳N] IS FOURIER TRF OF MAP OF RHO=1
 ⍝  Z[⍳2T] IS X,Y,..,X,Y ON RHO=1 AT T×○2÷N
C←2○S←(N|T○.×⍳N)×○2÷N←(ρF)[1+⎕IO←0]
Z←(C+.×F[0;])-(S+1○S)+.×F[1;]
Z←,⍝(2,ρZ)ρ(÷N↑÷2)×Z,(C+.×F[1;])+S+.×F[0;]



B←PFIT Z
 ⍝  Z[⍳K;⍳N] IS N POINTS ON CURVE IN K DIMS
 ⍝  B[⍳K;⍳N] IS B-SPLINE COEFFS, PERIOD N, FOR CLOSED CURVE
 ⍝  RESHAPE ALT INPUT Z[⍳2N] FOR 2-D CURVE
→(1≠ρρZ)ρB1
Z←⍝((0.5×ρZ),2)ρZ
B1:B←⍝(⍝Z)⌹(2-⍳B)⌽(2ρB)ρ(3ρ↑ 24 6),(‾3+B←‾1↑ρZ)ρ0
```

```
F←N BDFT B;⎕IO;P;L;C;S;C4;S4;M;X;I;J;K;U;V;W;CJ;SJ;D
 ⍝  B[⍳2;⍳M] IS B-SPLINE COEFFS FOR X(P), PERIOD M
 ⍝  Z[⍳2;(T)], PERIOD ○2, IS MAP OF THE UNIT CIRCLE
 ⍝  RETURNS FOURIER TRF OF Z, P(T) AT N ROOTS OF ONE
 ⍝  INITIAL P(T):   LINEAR IF 0=ρρN, ELSE P←N
→(ρρN)ρB1+⎕IO←F←0
→B2,P←(⍳N)×((ρB)[1])÷N
B1:N←ρP←N
 ⍝  CHECK N=4*L AND SETUP XERS FOR FFT'S
B2:→(L≠⌊L+4●N)/ERR
C←(K←4)⌽S←0,S,1,(⌽S),0,(-S),⁻1,-⌽S←1○(1+⍳N÷4)×○2÷N
S4←(ρC4← 4 4 ρ(1 0 ⁻1 0)[S4])ρ(0 1 0 ⁻1)[S4←4|,(⍳4)∘.×⍳4]
 ⍝  APPEND COEFFS FOR PERIODIC SPLINE EVALS
→B3,,B←D[;⁻1+M←(ρB)[1]],B,B[;⍳2]
ERR:→S∆BDFT←ERR
 ⍝  X''(P),X'(P) AND X(P) FROM SPLINE COEFFS, BEGIN ITERATION
B3:X←B+.×(0.25×K)-J←0⌈⁻1+K←0⌈2-|I←(⁻1,⍳M+2)∘.-M|P
X←X,[⁻0.5] B+.×(0.5×I)×(0.25×W←K×K)-V←J×J
X←X,[0](⌽6)×B+.×(0.25×W×K)-V×J
 ⍝  FFT OF BOUNDARY DATA (×N*÷2)
K←2ρ4,(J←N÷4),(V←X[2;I←1;]),U←X[2;0;]
B4:→(J=1)ρE4,(V←,⍉(C4+.×V)-S4+.×W),U←,⍉(C4+.×W←KρU)+S4+.×V←KρV
SJ←S[CJ←N|,(⍳J)∘.×(I←I×4)ρI×⍳4]
→B4,(J←J÷4),(V←(CJ×V)-SJ×W),U←((CJ←C[CJ])×W←U)+SJ×V
E4:→(F=1)ρE3,(U←,⍉KρU),V←,⍉(K←Lρ4)ρV
 ⍝  FFT FOR REGULAR PART OF Z(T)-X(P(T))
K←2ρ4,(J←N÷4),(I←1),(U←U×D),V←V×D←(⍳N)×2÷N×N
B5:→(J=1)ρE5,(V←,⍉(C4+.×V)+S4+.×W),U←,⍉(C4+.×W←KρU)-S4+.×V←KρV
SJ←S[CJ←N|,(⍳J)∘.×(I←I×4)ρI×⍳4]
→B5,(J←J÷4),(V←(CJ×V)+SJ×W),U←((CJ←C[CJ])×W←U)-SJ×V
E5:D←(,⍉KρU),V←,⍉(K←Lρ4)ρV
 ⍝  ADD SINGULAR PART OF D←Z-X
D←(2,N)ρ((÷○2)×(1⌽F-⁻2⌽F+X[2;1;]),⁻1⌽F-2⌽F+X[2;0;])-D
 ⍝  D=∆P BY NEWTON'S METHOD (ONE PASS)
D←(+/⍉D×X[1;;])+|(+/⍉X[1;;]×X[1;;])-+/⍉D×X[0;;]
 ⍝  LIMIT D FOR MONOTONICITY
P←P+D←(-⁻1⌽F)⌈D⌊F←(÷3)×(1+P,P[0]+M)-P
→B3,F←∧/(10*-⎕PP)>|D,0ρ⍈←'+'
E3:F←(3,N)ρ((÷N*÷2)×U,V),P
```