An Efficient Implementation of a Conformal Mapping Method using the Szegö Kernel

Manfred R. Trummer

Technical Report 84-24

December 1984

An Efficient Implementation of a Conformal Mapping Method using the Szegö Kernel †

Manfred R. Trummer

University of British Columbia
Department of Computer Science
Vancouver, B.C., CANADA
V6T 1W5

and

Seminar für Angewandte Mathematik
Eidgenössische Technische Hochschule
CH-8092 Zürich, Switzerland

ABSTRACT

An implementation, based on iterative techniques, of a method to compute the Riemann mapping function is presented. The method has been recently introduced by N. Kerzman and the author; it expresses the Szegö kernel as the solution of an integral equation of the second kind. It is shown how to treat symmetric regions. The algorithm is tested on five examples. The numerical results show that the method is competitive, with respect to accuracy, stability, and efficiency.

† Research partially supported by a postdoctoral fellowship under NSERC grant A8240

1. Introduction.

The Riemann mapping function of a smooth, bounded, simply connected domain Ω in $\mathbb{C}$ can be expressed in terms of the Szegö kernel, briefly S, by a well known classical formula. For the definition and basic properties of S we refer the reader to [HE85], [GA64], or [SZ39]. Until recently, methods based on the Szegö kernel have not been popular in numerical conformal mapping. The reason for this is that S was obtained by orthonormalization of monomials, a very difficult numerical task. Methods based on the Bergman kernel suffer from similar drawbacks. Practically all effective methods for conformal mapping are based on integral equations. Surveys on the topic of conformal mapping can be found e.g. in [GA64], [HE85], and [GU85].

However, the Szegö kernel can be computed as the solution of a second kind integral equation, introduced in [KT85]. The theoretical foundations of this equation can be found in [KS78]. Its explicit, smooth, skew-hermitian kernel vanishes on the diagonal, and has a very direct geometrical interpretation.

In this paper we present an effective implementation of the method suggested in [KT85]. The resulting algorithm is a collocation method with an operations count $O(n^2)$, where n is the number of collocation points. The emphasis here is on the computational part. A few results of the original work [KT85] are included, which allows this paper to be read independently.

It is organized as follows: in § 2 we describe the method, elaborating on the solution of the resulting system of linear equations in § 3. § 4 contains some details of the implementation, and shows how to take advantage of symmetries in the region Ω. We conducted some experiments with this method, and report the numerical results in § 5. Although without theoretical justification, the method also works for regions with corners (square). In the Appendix we present a new derivation of the integral equation, which requires us to dig a little bit into the theoretical background of the method.

Acknowledgements: The author wants to thank Norberto Kerzman and Peter Henrici for their encouragement and continuing interest in this work; Gene Golub for his suggestion to use the CGW-method; Jean-Paul Berrut for many inspiring discussions; and Jim Varah for the hospitality the author enjoyed at UBC, and for useful comments on the presentation of this work.

2. Description of the method.

The problem is to compute the Riemann mapping function R of an open, bounded, simply connected domain Ω in the complex plane $\mathbb{C}$, subject to the normalization $R(a) = 0, R'(a) > 0$, where $a \in \Omega$ is an arbitrary point. We assume that $\partial\Omega$, the boundary of Ω, admits a C^2-parametrization $z(t)$, $0 \leq t \leq \beta$, with $\dot{z}(t) = \frac{dz}{dt} \neq 0$. The unit tangent to $\partial\Omega$ at the point z will be denoted by $\dot{\gamma}(z) = \frac{\dot{z}(t)}{|\dot{z}(t)|}$.

Since $\partial\Omega$ is of class C^2, $R'(z)$ has a continuous extension to the closure $\overline{\Omega}$ (Kellog's theorem, see e.g. [WA61]). The method under investigation uses the known connection, (2.1) below, between R and the Szegö kernel S of Ω (derived from the transformation rule for the Szegö kernels of different regions):

THEOREM 1 : *The Szegö kernel $S(z,a)$ is continuous as a function of z on $\overline{\Omega}$, and*

$$R'(z) = \frac{2\pi}{S(a,a)} S^2(z,a), \qquad z \in \overline{\Omega}. \tag{2.1}$$

The Riemann mapping function R can be computed on the boundary by means of

$$R(z) = \frac{1}{i} \dot{\gamma}(z) \frac{R'(z)}{|R'(z)|}, \qquad z \in \partial\Omega. \tag{2.2}$$

Notice the useful formula (2.2) which, to our knowledge, has been used the first time in [KT85].

Hence, to solve the conformal mapping problem, it is sufficient to compute the Szegö kernel. The classical approach, to compute the Szegö kernel S via orthogonal functions, is useless from a numerical point of view. However, S satisfies an integral equation of the second kind. To formulate this equation we need the following definitions:

DEFINITION 2 : The Cauchy kernel H is defined by

$$H(w,z) := \frac{1}{2\pi i}\frac{\dot{\gamma}(z)}{z-w}, \quad w \in \overline{\Omega},\ z \in \partial\Omega,\ w \neq z\ ; \tag{2.3}$$

the kernel $A : \partial\Omega \times \partial\Omega \to \mathbb{C}$ (Kerzman-Stein kernel, [KS78]) is given by

$$A(w,z) := \begin{cases} \overline{H(z,w)} - H(w,z) & w,z \in \partial\Omega,\ w \neq z \\ 0 & w = z \in \partial\Omega. \end{cases} \tag{2.4}$$

THEOREM 3 ([KT85]): *The Szegö kernel $S(z,a)$ is, as a function of z, the unique solution of the integral equation*

$$S(z,a) + \int_{w \in \partial\Omega} A(z,w)S(w,a)\,d\sigma_w = \overline{H(a,z)}, \quad z \in \partial\Omega, \tag{2.5}$$

where $d\sigma$ denotes the arc length on $\partial\Omega$.

The reader can find a few more details of the theoretical background in the Appendix of this paper which contains a new derivation of the integral equation. For a thorough discussion see [KS78, KT85].

The Kerzman-Stein kernel has two important properties: A is continuous and skew-hermitian ($A(z,w) = -\overline{A(w,z)}$). Both features are important for our numerical computations.

Using any (regular) parametrization $z(t)$ of $\partial\Omega$, $0 \le t \le \beta$, the integral equation becomes

$$\phi(t) + \int_0^\beta k(t,s)\phi(s)ds = \psi(t), \quad 0 \le t \le \beta, \tag{2.6}$$

where

$$\phi(t) := |\dot{z}(t)|^{1/2} S(z(t),a), \tag{2.7a}$$

$$\psi(t) := |\dot{z}(t)|^{1/2} \overline{H(a,z(t))}, \tag{2.7b}$$

$$k(t,s) := |\dot{z}(t)|^{1/2} A\big(z(t),z(s)\big) |\dot{z}(s)|^{1/2}. \tag{2.7c}$$

This way the skew-hermitian property of the kernel is preserved, i.e., $k(s,t) = -\overline{k(t,s)}$.

We solve the integral equation (2.6) numerically by Nyström's method ([AT76]). Taking advantage of the periodicity of all the functions in (2.6) we choose the n equidistant collocation points $t_i = \frac{i-1}{n}\beta$ and the trapezoidal rule for Nyström's method to obtain

$$\phi(t_i) + \frac{\beta}{n} \sum_{j=1}^{n} k(t_i, t_j)\phi(t_j) = \psi(t_i), \quad 1 \leq i \leq n . \tag{2.8}$$

Defining the skew-hermitian matrix B $(= -\overline{B}^T)$ by

$$B_{ij} := \frac{\beta}{n} k(t_i, t_j),$$

and

$$x_i = \phi(t_i), \quad y_i = \psi(t_i),$$

(2.8) can be rewritten as

$$(I + B)\mathbf{x} = \mathbf{y} . \tag{2.9}$$

We will elaborate on how to solve this complex system of linear equations in § 3.

After the solution ϕ has been computed at the collocation points t_i, discretization of (2.6) provides us with a *natural* interpolation formula:

$$\phi(t) = \psi(t) - \frac{\beta}{n} \sum_{j=1}^{n} k(t, t_j)x_j . \tag{2.10}$$

Finally, the boundary correspondence $\theta(t)$, defined by

$$R\big(z(t)\big) = e^{i\theta(t)}, \tag{2.11}$$

can be computed by means of

$$\theta(t) = arg\big(-i\,\phi^2(t)\dot{z}(t)\big) . \tag{2.12}$$

See [KT85] for details.

Remark: We were surprised to get very accurate approximations by using a collocation method as opposed to methods where the solution is approximated in a finite dimensional function space. Examples of the latter type are the methods of Berrut [BE76, BE84, BE85] based on expansions in Fourier series. However, since the trapezoidal rule integrates trigonometric polynomials of matching degree exactly, both methods (Nyström and Fourier) yield the same approximations at the collocation points (see [BE84a,TRU84]). But Nyström's method does not need Fourier transformations.

3. The generalized conjugate gradient method.

Our aim is to solve a system of complex linear equations of the form

$$(I + B)\,\mathbf{x} = \mathbf{y}\,, \tag{3.1}$$

where B is a skew-hermitian n by n matrix, i.e. $B^H := \overline{B}^T = -B$. We work in the (real) vector space $\mathbb{C}^n$, endowed with the scalar product

$$<\mathbf{x},\mathbf{y}> := \mathrm{Re}\left(\mathbf{x}^H \mathbf{y}\right) = \mathrm{Re}\left(\sum_{j=1}^{n} \overline{x}_j\, y_j\right), \tag{3.2}$$

This vector space has real dimension $2n$; the scalar product (3.2) corresponds to the scalar product in $\mathbb{R}^{2n}$. B^H is the adjoint of B with respect to the scalar product (3.2).

We solve the system (3.1) using an iterative method based on conjugate gradient (CGW). This algorithm has been suggested by Concus and Golub [CG75], and by Widlund [WI78], for nonsymmetric real systems. It uses a splitting of the matrix into a symmetric and a skew-symmetric part. In our case we are dealing with complex systems. On the other hand, the symmetric part is just the identity matrix, simplifying everything.

We consider an iteration of the form

$$\mathbf{x}^{(k+1)} := \mathbf{x}^{(k-1)} + \omega_{k+1}\left(\alpha_k\, \mathbf{r}^{(k)} + \mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right) \tag{3.3}$$

with

$$\mathbf{r}^{(k)} := \mathbf{y} - (I + B)\,\mathbf{x}^{(k)}\,, \tag{3.4}$$

the residual at the k-th step. It is clear that the limit of the sequence $\mathbf{x}^{(k)}$ is a solution of (3.1), provided the sequence converges and $\alpha_k\,\omega_{k+1} \neq 0$. For the CGW-method $\{\alpha_k\,,\,\omega_{k+1}\}$ are computed such that the residuals are orthogonal to each other, i.e.

$$<\mathbf{r}^{(p)},\mathbf{r}^{(q)}> = 0\,, \quad \text{for } p \neq q\,. \tag{3.5}$$

Since the $\mathbf{r}^{(p)}$ are vectors in a $2n$-dimensional space, the iteration converges in no more than $2n$ steps. However, in practice the calculated vectors $\mathbf{r}^{(p)}$ will not be orthogonal because of roundoff errors. Therefore, we cannot expect termination within a finite number of steps. We permit this loss of orthogonality, since we are more interested in the iterative aspects of the algorithm. In our examples we have convergence within less than 10 to 12 iterations, even for $n \approx 1000$. But it should be noted that we are able to supply 'good' starting vectors.

Since B is skew-hermitian, $\mathbf{z}^H B\,\mathbf{z}$ is purely imaginary for any $z \in \mathbb{C}^n$, thus

$$<\mathbf{z}\,,\,B\,\mathbf{z}> = 0\,. \tag{3.6}$$

Therefore, we can repeat the same computations as in [CG75] to obtain

$$\alpha_k \equiv 1\,, \tag{3.7}$$

and

$$\omega_1 = 1\,, \tag{3.8a}$$

$$\omega_{k+1} = \left(1 + \frac{\langle \mathbf{r}^{(k)}, \mathbf{r}^{(k)} \rangle}{\langle \mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)} \rangle} \times \frac{1}{\omega_k}\right)^{-1}. \tag{3.8b}$$

The starting vector $\mathbf{x}^{(0)}$ can be chosen arbitrarily, but, of course, a good choice insures faster convergence. Putting (3.3), (3.4), (3.7), and (3.8) together gives the algorithm used in § 4. See [CG75] and [WI78] for a detailed description of the method and its properties.

4. Symmetric regions. Implementation details.

The method described in the previous sections has been implemented on a CDC Cyber 174 machine in Pascal. In this chapter we mention a few details of the implementation, which allows one to handle up to 1000 collocation points within reasonable computing time (approximately 1000 CP-seconds), and without storage problems. It is also possible to take advantage of symmetries of the region Ω.

One of the reasons to use an iterative method for solving (2.9) was to avoid storing the matrix B. Instead, the elements of B are recomputed at each iteration step. The elements are given by

$$A(z(t_j),z(t_k)) = \frac{1}{2\pi i} \frac{\dfrac{\dot z(t_j)}{|\dot z(t_j)|} - \dfrac{\dot z(t_k)}{|\dot z(t_k)|}}{z(t_k) - z(t_j)} \tag{4.1a}$$

$$B_{jk} = \frac{\beta}{n} |\dot z(t_j)|^{1/2} A(z(t_j),z(t_k)) |\dot z(t_k)|^{1/2}, \tag{4.1b}$$

where the t_j are the collocation points in the parameter interval. We compute and store the values of

$$z(t_j), \quad \frac{\dot z(t_j)}{|\dot z(t_j)|}, \text{ and } |\dot z(t_j)|^{1/2};$$

the evaluation of these functions might be rather expensive, but it has to be done only once. The number of operations involved is $O(n)$, where n is the number of collocation points. Each element B_{jk} can now be computed very

cheaply, namely within 10 **real** multiplications and additions. Each iteration step requires essentially a matrix-vector multiplication, yielding an $O(n^2)$ operations count for the algorithm.

We start the iteration for solving (2.9) with the right hand side y, or, if available, with the solution for smaller n. Here the interpolation formula (2.10) proves to be very handy. The latter choice insures very fast convergence of the CGW-method (2 or 3 iterations suffice for large n). In fact, it is much cheaper to first compute solutions for smaller values of n in order to provide a good starting vector for the rather expensive iteration part. It should be noted that for the CGW-method no estimation of parameters is required; this is not true for Richardson extrapolation, where a reliable estimate of $||(I+B)||$ is essential for a good performance of the iteration (see [MU84]).

Symmetries of the region Ω (with respect to the origin and/or real and imaginary axis) can be used to cut the computing time by a factor of 4, or even 16. This is achieved by working with a new kernel, related to the symmetry identification function of the region:

DEFINITION 4 : Let Ω be a simply connected domain in $\mathbb{C}$, the boundary $\partial\Omega$ being a closed Jordan curve. A complex valued function λ with the property

$$z \in \partial\Omega,\ \lambda(z) = \lambda(w) \ \Rightarrow\ w \in \partial\Omega$$

is called a symmetry identification function[1] of Ω. In particular, if $\partial\Omega$ is invariant under rotations about the origin with angle $2\pi/k$, then $\lambda(z) = z^k$ is a symmetry identification function of Ω.

Examples 1, 2, and 4 in § 5 have the symmetry identification function $\lambda(z) = z^2$, example 5 (unit square) $\lambda(z) = z^4$. The Szegö kernel reflects this symmetry:

$$\lambda(z) = \lambda(w) \ \Rightarrow\ S(z,0) = S(w,0). \tag{4.2}$$

Therefore, if $\lambda(z) = z^k$, it suffices to compute the Szegö kernel at the first $\frac{n}{k}$ collocation points.

DEFINITION 5 : We denote the intersection of $\partial\Omega$ with the angular domain $0 \leq \arg z \leq \frac{2\pi}{k}$ by $(\partial\Omega)_k$:

$$(\partial\Omega)_k := \partial\Omega \cap \{z : 0 \leq \arg z \leq \frac{2\pi}{k}\}\ .$$

Equation (2.5) then becomes (with $a = 0$)

[1] A symmetry identification function is not a mapping under which Ω is invariant. It maps all points of the orbit of invariance transformations onto the same point, and all pre-images of a given point are on the orbit of one invariance transformation.

$$S(z,0) + \int_{w \in \partial\Omega} A(z,w) S(w,0) d\sigma_w = \overline{H(0,z)}, \quad z \in (\partial\Omega)_k . \tag{4.3}$$

Rewriting the integral in (4.3) in terms of an integral over $(\partial\Omega)_k$ we get

$$S(z,0) + \int_{w \in (\partial\Omega)_k} A_S(z,w) S(w,0) d\sigma_w = \overline{H(0,z)}, \tag{4.4}$$

where

$$A_S(z,w) := \sum_{\substack{\lambda(v)=\lambda(w) \\ v \in \partial\Omega}} A(z,v) . \tag{4.5}$$

This new kernel A_S is again skew-hermitian, but does not vanish on the diagonal. There is a simple relation between A_S, A, and the symmetry identification function λ of a region Ω:

DEFINITION 6 : The kernel $H_S : \overline{\Omega} \times (\partial\Omega)_k \to \mathbb{C}$ is defined (compare with (4.5)) by

$$H_S(z,w) := \sum_{\substack{\lambda(v)=\lambda(w) \\ v \in \partial\Omega}} H(z,v), \quad z \neq w , \tag{4.6a}$$

$$H_S(z,z) := \sum_{\substack{\lambda(v)=\lambda(z) \\ z \neq v \in \partial\Omega}} H(z,v), \quad z \in (\partial\Omega)_k . \tag{4.6b}$$

The kernel H_λ is defined by

$$H_\lambda(w,z) := \frac{1}{2\pi i}\frac{\lambda'(z)\dot{\gamma}(z)}{\lambda(z)-\lambda(w)}, \quad w \in \overline{\Omega},\ z \in (\partial\Omega)_k,\ \lambda(w) \neq \lambda(z); \tag{4.7}$$

and the kernel $A_\lambda : (\partial\Omega)_k \times (\partial\Omega)_k \to \mathbb{C}$ is given by

$$A_\lambda(w,z) := \begin{cases} \overline{H_\lambda(z,w)} - H_\lambda(w,z) & w,z \in (\partial\Omega)_k,\ w \neq z \\ 0 & w = z \in (\partial\Omega)_k . \end{cases} \tag{4.8}$$

H_λ and A_λ look like "pull-backs"[2] of the Cauchy kernel and the Kerzman-Stein kernel of the "region" $\lambda(\Omega)$. However, λ is not one-to-one.

It is clear from the definition that for $z,\ w \in (\partial\Omega)_k$

$$A_S(w,z) = \overline{H_S(z,w)} - H_S(w,z) . \tag{4.9}$$

With this notation we have the following result:

LEMMA 7 : *Suppose* $\lambda(z) = z^k$, $k \in \mathbb{N}$, *is a symmetry identification function of* Ω, *and the kernel* A_S *is defined by* (4.5). *Then*

$$A_S(w,z) = A_\lambda(w,z), \quad \text{for } w \neq z , \tag{4.10}$$

and

$$A_S(z,z) = (1-k)\,\frac{1}{2\pi i}\,\mathrm{Re}\,\frac{\dot{\gamma}(z)}{z} . \tag{4.11}$$

Moreover, the Szegö kernel $S(z,0)$, $z \in (\partial\Omega)_k$, *satisfies*

---

[2] But the reason (4.10) holds does not seem to be related to pull-back arguments.

$$S(z,0) + \int_{w \in (\partial\Omega)_k} A_S(z,w) S(w,0) d\sigma_w = \overline{H(0,z)} . \tag{4.12}$$

PROPOSITION 8 :

$$\sum_{j=0}^{k-1} \frac{e^{i2\pi j/k}}{e^{i2\pi j/k} w - z} = \frac{k w^{k-1}}{w^k - z^k} \tag{4.13}$$

$$\sum_{j=1}^{k-1} \frac{e^{i2\pi j/k}}{e^{i2\pi j/k} - 1} = \frac{k-1}{2} . \tag{4.14}$$

Proof: On the right hand side of (4.13) we have the rational function

$$q_z(w) := \frac{p_z'(w)}{p_z(w)} , \tag{4.15}$$

with $p_z(w) := w^k - z^k$. p_z has zeros of order 1 at $w_j = e^{-i2\pi j/k} z$, $0 \leq j < k$. The residues of q_z at each of the points w_j are equal to 1, the order of the zero w_j of p_z. Since q_z is a rational function which vanishes at ∞, it equals the sum of its principal parts (see e.g. [HE74]):

$$q_z(w) = \sum_{j=0}^{k-1} \frac{1}{w - e^{-i2\pi j/k} z} , \tag{4.16}$$

which is obviously equivalent to (4.13).

The left hand side of (4.14) can now be written as

$$\lim_{z \to 1} \left(q_z(1) - \frac{1}{1-z} \right) = \lim_{z \to 1} \left(\frac{k}{1-z^k} - \frac{1}{1-z} \right) ; \tag{4.17}$$

an elementary argument shows that the limit in (4.17) exists, and is equal to $\frac{k-1}{2}$.

□

Proof of Lemma 7: From (4.9) and (2.4) it suffices to show that for $z \neq w$

$$H_S(z,w) = H_\lambda(z,w) . \tag{4.18}$$

(4.18) follows from Proposition 8, namely:

$$\begin{aligned} 2\pi i \; H_S(z,w) &= \sum_{\lambda(v)=\lambda(w)} \frac{\dot\gamma(v)}{v-z} \\ &= \sum_{j=0}^{k-1} \frac{e^{i2\pi j/k}\dot\gamma(w)}{e^{i2\pi j/k} w - z} \qquad (4.19) \\ &= \frac{k w^{k-1}\dot\gamma(w)}{w^k - z^k} \;=\; 2\pi i \; H_\lambda(z,w) . \end{aligned}$$

In a similar way (the summation in (4.19) starts with $j=1$) we obtain by applying (4.14)

$$2\pi i \; H_S(z,z) = \frac{k-1}{2} \frac{\dot\gamma(z)}{z} , \tag{4.20}$$

which in view of of (4.9) yields (4.11).

□

So, again, by precomputing

$$\lambda(z(t_j)),\ \ \lambda'(z(t_j))\dot{\gamma}(z(t_j)),\ \text{ and }\ \ |\dot{z}(t_j)|^{1/2},$$

the new kernel A_S can be evaluated in the same way and as cheaply as the kernel A, which is exactly the point of Lemma 7.

If the region Ω is e.g. invariant under reflections at the real or imaginary axis, the number of unknowns can be further reduced. The Szegö kernel exhibiting the same symmetries as Ω satisfies

$$S(\overline{z},0) = \overline{S(z,0)},$$

or, respectively,

$$S(-\overline{z},0) = \overline{S(z,0)}.$$

When taking advantage of these symmetries we do not actually reduce the number of unknowns, but rather exploit the symmetry relations while performing the matrix vector multiplication. The iteration preserves the symmetries at each step, and we always start with a "symmetric vector".

5. Examples and numerical results.

We apply the algorithm described in this paper to five test regions, subject to the normalization $R(0) = 0$, $R'(0) > 0$.

We list the sup norm error $||\theta_n - \theta||_\infty$, where θ_n is the approximation obtained with n collocation points, and θ is the exact boundary correspondence function (see (2.11)). The sup norm is computed numerically by comparing θ_n and θ at 120 equally spaced points in the parameter interval, most of which are not collocation points. Tables 1 to 5 show the results[3].

The test regions are the following:

Example 1: Inverted Ellipse $(0<p\leq 1)$.

$$z(t) = \sqrt{1-(1-p^2)\cos^2 t}\ \ e^{it}. \tag{5.1}$$

$$\tan t = p\ \tan\theta(t). \tag{5.2}$$

This region is obtained by reflecting the exterior of an ellipse with axes 1 and $\frac{1}{p}$ in the unit circle. For $p \to 0$ the problem becomes more and more difficult.

[3] We use a different error norm in Table 2b; see the remarks for Example 2.

Example 2: Ellipse $(0 \leq \epsilon < 1$; *axis ratio* $= \frac{1+\epsilon}{1-\epsilon})$.

$$z(t) = e^{it} + \epsilon e^{-it} \quad . \tag{5.3}$$

$$\theta(t) = t + 2 \sum_{m=1}^{\infty} \frac{(-1)^m}{m} \frac{\epsilon^m}{1+\epsilon^{2m}} \sin(2mt) \, . \tag{5.4}$$

For large values of the axis ratio (ϵ close to 1) the distribution of the collocation points on the boundary becomes very poor. Therefore we carried out some computations using arc length parametrization. In addition, this allows a direct comparison with the numerical results obtained by Hayes et. al ([HKK75]).

Working with arc length requires us to evaluate elliptic integrals numerically; the arc length parameter s is given by

$$s(t) = \int_0^t | \dot{z}(\tau) | \, d\tau = \int_0^t \sqrt{1 + \epsilon^2 - 2\epsilon \cos 2\tau} \, d\tau \; ; \tag{5.5}$$

we evaluate this integral numerically by the trapezoidal rule and Romberg extrapolation. The inverse function t of s can be computed by Newton's method:

$$t_{n+1} = t_n + \frac{s - s(t_n)}{| \dot{z}(t_n) |} \quad . \tag{5.6}$$

The results for the parametrization (5.3) are listed in Table 2a, the ones for arc length parametrization in Table 2b. In Table 2b we list the average error

$||\theta_n - \theta||_1$, computed at 32 equally spaced points on the boundary (this error norm was used in [HKK75]).

The arc length parametrization seems to work better for small and moderate values of n, but if n is sufficiently large, we obtain more accurate values using the parametrization (5.3). Moreover, the computation of the inverse parameter transformation by Newton's method is very expensive, and consumes (for $n < 512$) more computing time than all other calculations.

Example 3: Epitrochoid ("Apple", $0 \leq \alpha < 1$).

$$z(t) = e^{it} + \frac{\alpha}{2} e^{2it} \ . \tag{5.7}$$

$$\theta(t) = t \ . \tag{5.8}$$

For $\alpha = 1$ the boundary curve is the cardioid with a cusp at $-1/2$. The method anticipates the upcoming singularity as α approaches the critical value 1.

Example 4: Oval of Cassini $(0 \leq \alpha < 1)$.

$$|z - \alpha| \ |z + \alpha| = 1. \tag{5.9a}$$

$$z(t) = \left(\alpha^2 \cos 2t + \sqrt{1 - \alpha^4 \sin^2 2t}\right)^{1/2} e^{it} \ . \tag{5.9b}$$

$$\theta(t) = t - \frac{1}{2} arg\left(w(t)\right) \ , \tag{5.10}$$

where

$$w(t) = \sqrt{1 - \alpha^4 \sin^2 2t} + i\ \alpha^2 \sin 2t\ . \tag{5.11}$$

This example causes no problems, unless α comes very close to the critical value 1, for which the region degenerates to a point.

Example 5: Unit square.

$$\cos\theta(t) = \mathrm{dn}\ \big(Ky\big)\ , \tag{5.12}$$

where dn denotes the Jacobian elliptic function [AS65, p 570, p 591]. Notice that this region does not satisfy the smoothness condition for the boundary. The tangent is not even defined at the corners; we define the tangent there to point into the direction of the sum of the tangents of the adjacent sides. Without further modifications we still get reasonable approximations to the solution. However, we do not feel that this is a good way to treat regions with corners. See [TR80] for conformal maps of polygons.

Finally, a few words about how the performance of our method compares to other methods. These comparisons have to be taken with a large grain of salt: There is no standard measure for errors in the literature, and, we do not have sufficient experimental data on computing time.

Our results are about as accurate as the results of Berrut [BE84,BE85], but sometimes fewer collocation points are required for our method. On the other hand, we have to use complex arithmetic, and the representation of the

solution in terms of a trigonometric polynomial might have some advantages. Both methods require approximately the same computing time. An advantage of our method is less storage requirement ($O(n)$); storing the matrix B (see (2.9)) reduces the computing time by approximately 50%.

Looking at other methods based on Symm's equation (a first kind integral equation which determines the derivative of the boundary correspondence [SY66]), like the one in [HKK72] and [HKK75] indicates that our method performs better[4]. Our results also compare favourably to the ones in [HO83]. The logarithmic singularity in Symm's equation might be an advantage when working with regions with corners.

The second method implemented in [HKK75] is based on a second kind integral equation of Lichtenstein ([LI17]). Again, our method produces more accurate values.

There are methods for computing the inverse boundary correspondence (i.e. the mapping from the disk onto the region Ω) in $O(n \log n)$ operations, where n is the discretization parameter (number of collocation points, or degree of approximating trigonometric polynomial). Among these are Gutknecht's method for solving Theodorsen's integral equation (based on function conjugation) [GU81,GU85], and Wegmann's method [WE78]. The $O(n \log n)$ operations count is accomplished by employing the Fast Fourier transform (see also [HE79]). In both methods a nonlinear integral equation has

[4] Notice that the error listed in [HKK75] is an average error, the same error measure as used in Table 2b.

to be solved, which is done by iterative procedures. However, it is not clear which value of n is the break-even point. An $O(n^2)$-method might be faster for small values of n. To compare accuracy, we have to compute the inverse boundary function. This has been implemented by Müller [MU84] for our method, following the suggested procedure in [KT85]. But we do not yet have enough experimental data available to make this comparison.

This is by no means an exhaustive list of competing methods (we refer again to [GA64], [HE85], and [GU85]). However, we hope this will help to put the proposed method in its place.

TABLE 1: Inverted Ellipse

Error norm $\| \theta - \theta_n \|_\infty$

n	p=0.8	p=0.5	p=0.2	p=0.1	p=0.05
4	4.5 (-03)	1.3 (-01)	-	-	-
8	8.7 (-05)	2.3 (-02)	1.0	-	-
16	3.0 (-05)	6.8 (-04)	2.8 (-01)	-	-
32	9.9 (-14)	4.6 (-06)	8.9 (-02)	4.9 (-01)	-
64	3.9 (-14)	2.0 (-10)	5.9 (-03)	1.4 (-01)	3.4 (-01)
128	-	5.3 (-14)	7.5 (-05)	1.0 (-02)	5.0 (-01)
256	-	-	1.3 (-07)	5.3 (-04)	1.7 (-01)
512	-	-	1.5 (-12)	2.0 (-05)	1.8 (-02)
1024	-	-	-	5.3 (-08)	1.7 (-04)

TABLE 2a: Ellipse

Error norm $|| \theta - \theta_n ||_\infty$

n	axis ratio			
	1.2	1.5	2.0	3.0
4	2.1 (-03)	2.6 (-02)	1.7 (-01)	2.0
8	1.8 (-05)	9.8 (-04)	1.6 (-02)	2.6 (-01)
16	1.2 (-09)	1.6 (-06)	1.9 (-04)	1.4 (-02)
32	2.1 (-14)	4.0 (-12)	2.9 (-08)	5.4 (-05)
64	-	3.9 (-14)	4.2 (-14)	8.2 (-10)
128	-	-	-	4.8 (-14)

n	axis ratio		
	5	10	20
32	4.2 (-02)	-	-
64	6.3 (-05)	3.4	-
128	1.5 (-10)	1.8 (-02)	-
256	5.3 (-13)	1.1 (-07)	3.1
512	3.0 (-13)	1.5 (-09)	2.9
1024	-	1.8 (-10)	7.8 (-03)
2048	-	2.0 (-10)	2.8 (-04)

TABLE 2b: Ellipse (arc length parametrization)

Error norm $|| \theta - \theta_n ||_1$

n	axis ratio				
	1.5	3.0	5.0	10.0	20.0
8	7.1 (-05)	2.4 (-02)	-	-	-
16	1.1 (-07)	3.2 (-04)	3.3 (-02)	-	-
32	1.2 (-11)	6.0 (-06)	1.7 (-04)	5.0 (-01)	-
64	1.5 (-14)	3.1 (-08)	5.9 (-06)	4.8 (-03)	-
128	-	2.7 (-12)	9.3 (-08)	2.5 (-06)	6.8 (-01)
256	-	9.4 (-14)	1.6 (-10)	2.5 (-07)	3.3 (-04)
512	-	-	1.0 (-11)	4.9 (-09)	9.6 (-07)

TABLE 3: Epitrochoid ("Apple")

Error norm $\| \theta - \theta_n \|_\infty$

n	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	
4	1.5 (-02)	7.3 (-04)	1.0 (-02)	
8	3.0 (-08)	1.7 (-05)	8.8 (-04)	
16	7.1 (-14)	2.7 (-09)	3.9 (-06)	
32	4.3 (-14)	5.7 (-14)	2.5 (-10)	
64	-	4.3 (-14)	2.8 (-14)	
n	$\alpha = 0.8$	$\alpha = 0.9$	$\alpha = 0.95$	$\alpha = 0.99$
16	1.3 (-03)	2.1 (-02)	7.4 (-02)	2.5 (-01)
32	6.1 (-06)	1.0 (-03)	1.6 (-02)	1.1 (-01)
64	1.3 (-09)	4.4 (-06)	5.2 (-04)	3.2 (-02)
128	9.9 (-14)	1.6 (-09)	2.6 (-06)	5.8 (-03)
256	7.1 (-14)	5.7 (-14)	1.2 (-09)	3.8 (-04)

TABLE 4: OVAL of CASSINI

Error norm $\| \theta - \theta_n \|_\infty$

n	$\alpha = 0.2$	$\alpha = 0.5$	$\alpha = 0.8$	$\alpha = 0.9$
4	1.0 (-05)	2.5 (-03)	5.1 (-02)	1.3 (-01)
8	3.4 (-09)	3.2 (-05)	6.0 (-03)	3.4 (-02)
16	3.5 (-14)	6.4 (-09)	7.7 (-05)	1.7 (-03)
32	-	6.3 (-14)	1.7 (-08)	5.7 (-06)
64	-	4.3 (-14)	1.1 (-10)	5.6 (-10)
128	-	-	1.6 (-13)	6.8 (-12)
256	-	-	3.6 (-14)	1.1 (-13)

n	$\alpha = 0.99$	$\alpha = 0.999$	$\alpha = 0.9999$
32	1.9 (-02)	1.6	1.0
64	5.4 (-04)	7.2 (-02)	1.0
128	6.3 (-07)	6.7 (-03)	3.3 (-01)
256	1.2 (-12)	8.6 (-05)	3.5 (-02)
512	2.6 (-13)	1.9 (-08)	2.0 (-03)

TABLE 5: Unit Square

n	$\| \theta - \theta_n \|_\infty$	*error at* $1 + i\frac{15}{16}$
32	3.9 (-03)	3.9 (-03)
64	2.5 (-03)	1.9 (-03)
128	1.1 (-03)	2.0 (-04)
256	5.7 (-04)	2.6 (-04)
512	2.2 (-04)	9.7 (-05)
1024	8.2 (-05)	3.8 (-05)
2048	3.2 (-05)	1.5 (-05)
4096	1.2 (-05)	5.8 (-06)

Appendix: Another derivation of the integral equation.

We present a derivation of the integral equation (2.5) different (and more elementary) from the one in the original paper [KT85]. The equation is derived from the corresponding operator equation (A.8), but avoids the use of Dirac δ-functions. We shall not copy theoretical results and proofs contained in [KS78] and [KT85]. However, for the reader's convenience, we recite some facts from these papers.

The space $L^2 = L^2(\partial\Omega, d\sigma)$ of complex valued square integrable functions on $\partial\Omega$ ($d\sigma =$ arc length) contains a closed subspace H^2, called the Hardy space. H^2 consists of those $u \in L^2$ which are the boundary values of some (uniquely determined) holomorphic function in Ω in the L^2-sense. The Szegö kernel is related to the orthogonal projector

$$\mathbf{S}\colon L^2 \to \mathrm{H}^2 \tag{A.1}$$

as follows:

$$\hat{u}(w) := \int_{z \in \partial\Omega} S(w,z)u(z)d\sigma_z\,, \quad w \in \Omega \tag{A.2}$$

is holomorphic in Ω and assumes $\mathbf{S}u$ as its boundary values. In the same way the Cauchy kernel leads to a bounded linear operator $\mathbf{H}\colon L^2 \to \mathrm{H}^2$, which is an oblique projector ($\mathbf{H}^2 = \mathbf{H}$, but in general $\mathbf{H}^* \neq \mathbf{H}$).

The Kerzman-Stein kernel is connected to a compact operator $\mathbf{A}\colon L^2 \to L^2$,

$$\mathbf{A}u(w) = \int_{z \in \partial\Omega} A(w,z)u(z)d\sigma_z \ . \tag{A.3}$$

$\mathbf{A}$ measures how much $\mathbf{H}$ deviates from being selfadjoint (i.e. being an orthogonal projector):

$$\mathbf{A} = \mathbf{H}^* - \mathbf{H} \ . \tag{A.4}$$

It is a nontrivial matter to prove that the operator in (A.4) is compact, and can be represented by an integral operator with continuous kernel. A cancellation of singularities in the Cauchy kernel H and its adjoint takes place.

Since both, $\mathbf{H}$ and $\mathbf{S}$, project onto the same subspace H^2, there holds

$$\mathbf{HS} = \mathbf{S} \ , \tag{A.5}$$

$$\mathbf{SH} = \mathbf{H} \ . \tag{A.6}$$

Taking adjoints in (A.6) yields

$$\mathbf{H}^*\mathbf{S} = \mathbf{H}^* \ . \tag{A.7}$$

Using (A.4) and (A.5) we finally obtain

$$\mathbf{H}^* = \mathbf{S} + \mathbf{AS} \ . \tag{A.8}$$

(A.8) is the adjoint of the corresponding operator equation in [KT85]. It is the starting point to derive the integral equation (2.5) in Theorem 3.

Let $a \in \Omega$ be fixed throughout the sequel. We apply (A.8) to the function

$$\chi(z) := \overline{H(a,z)} \ , \tag{A.9}$$

where H is the Cauchy kernel (Definition 2). Using the reproducing property of the Cauchy kernel, and $S(z,w) = \overline{S(w,z)}$ (because $\mathbf{S}$ is selfadjoint) we get immediately

$$\begin{aligned} \mathbf{S}\overline{H(a,w)} &= \int_{w \in \partial\Omega} S(z,w)\overline{H(a,w)}\, d\sigma_w \\ &= \int_{w \in \partial\Omega} \overline{S(w,z)H(a,w)}\, d\sigma_w \\ &= \overline{\int_{w \in \partial\Omega} H(a,w)S(w,z)\, d\sigma_w} \\ &= \overline{S(a,z)} = S(z,a)\,, \end{aligned} \tag{A.10}$$

because S is holomorphic in the first variable. Therefore the right hand side of (A.8) applied to χ gives

$$S(z,a) + \int_{w \in \partial\Omega} A(z,w)S(w,a)\, d\sigma_w\ . \tag{A.11}$$

To compute $\mathbf{H}^*\chi$ we have to resort to the Plemelj formula (see e.g. [HE85]):

$$\mathbf{H}\ u(z) = \frac{1}{2}u(z) + \text{P.V.} \int_{w \in \partial\Omega} H(z,w)u(w)\, d\sigma_w\,,\ z \in \partial\Omega, \tag{A.12}$$

$$\mathbf{H}^*u(z) = \frac{1}{2}u(z) + \text{P.V.} \int_{w \in \partial\Omega} \overline{H(z,w)}u(w)\, d\sigma_w\,,\ z \in \partial\Omega. \tag{A.13}$$

At first glance it seems to be very plausible that $\mathbf{H}^*\chi = \chi$; however, this has to be checked carefully, since the kernel of $\mathbf{H}^*$ has a singularity for

$z = w$.

We compute the principal value integral in (A.13):

$$\text{P.V.} \int \overline{H(w,z)H(a,w)}\, d\sigma_w = \overline{\text{P.V.} \int H(w,z)H(a,w)\, d\sigma_w} \ . \quad \text{(A.14)}$$

Let

$$D_\epsilon := \{ \mid z - w \mid < \epsilon \}, \quad \Omega_\epsilon := \Omega \backslash D_\epsilon,$$

and

$$C_\epsilon : w = z + \epsilon e^{it}, \quad \alpha(\epsilon) \leq t \leq \beta(\epsilon)$$

(see Figure 1). It is evident that

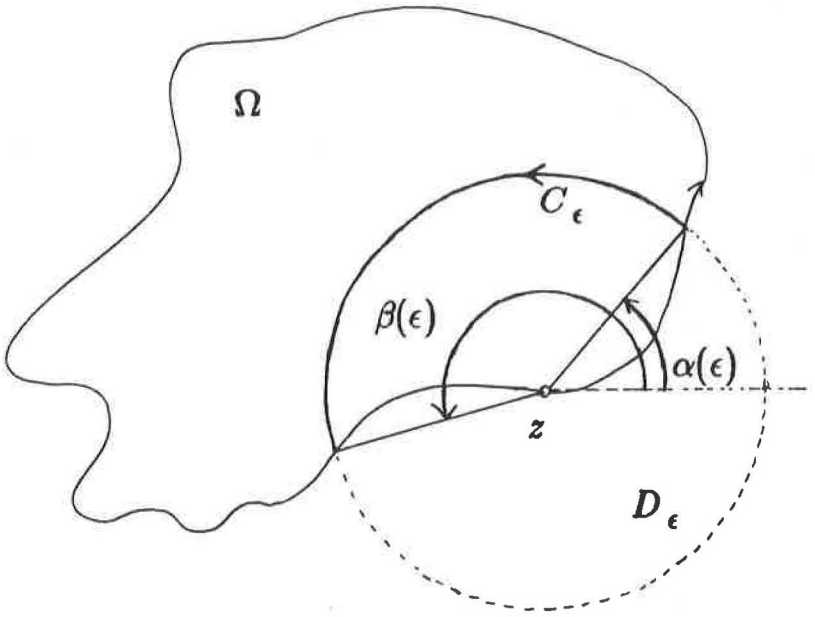

Figure 1

$$\lim_{\epsilon \to 0} [\beta(\epsilon) - \alpha(\epsilon)] = \pi \,. \tag{A.15}$$

Thus we get

$$\begin{aligned} &\text{P.V.} \int\limits_{w \in \partial\Omega} H(w,z)H(a,w)d\sigma_w \\ &= \lim_{\epsilon \to 0} \int\limits_{\partial\Omega \setminus D_\epsilon} H(w,z)H(a,w)d\sigma_w \qquad \text{(A.16)} \\ &= \lim_{\epsilon \to 0} \left(\int\limits_{\partial\Omega_\epsilon} + \int\limits_{C_\epsilon} \right) . \end{aligned}$$

The first integral is equal to $H(a,z)$, again because of the reproducing property of the Cauchy kernel H. Notice that H does not have a singularity on $\partial\Omega_\epsilon$. As for the second integral we get the following:

$$\begin{aligned} \int\limits_{C_\epsilon} &= \frac{1}{2\pi i}\frac{1}{2\pi i} \int\limits_{C_\epsilon} \frac{\dot\gamma(z)}{z-w}\frac{\dot\gamma(w)}{w-a}d\sigma_w \\ &= \frac{1}{2\pi i}\frac{1}{2\pi i} \int\limits_{C_\epsilon} \frac{\dot\gamma(z)}{z-w}\frac{dw}{w-a} \qquad \text{(A.17)} \\ &= \frac{1}{2\pi i}\frac{\dot\gamma(z)}{z-a}\,\frac{1}{2\pi i} \int\limits_{\alpha(\epsilon)}^{\beta(\epsilon)} \frac{-i\epsilon e^{it}}{\epsilon e^{it}}\frac{z-a}{z-a+\epsilon e^{it}}dt \\ &= H(a,z)\,\frac{-1}{2\pi} \int\limits_{\alpha(\epsilon)}^{\beta(\epsilon)} \frac{z-a}{z-a+\epsilon e^{it}}dt \,. \end{aligned}$$

As $\epsilon \to 0$, the integrand in the last line of (A.17) converges uniformly to 1.

Hence, from (A.15) and (A.16), the principal value integral in (A.14) equals $\frac{1}{2}\overline{H(a,z)}$. The Plemelj formula finally yields

$$\mathbf{H}^*\chi = \overline{H(a,z)} = \chi \ . \tag{A.18}$$

(A.10) and (A.18) give the integral equation (2.5), thus proving Theorem 3.

□

Remark: The integral equation (2.5) can be verified without using the operator equation (A.8). This is due to Henrici [HE85].

References.

[AS65] Abramowitz, M., and Stegun I.A., *Handbook of mathematical functions.* Dover, New York, 1965.

[AT76] Atkinson, K.E., *A survey of numerical methods for the solution of Fredholm integral equations of the second kind.* SIAM, Philadelphia, (1976)

[BE76] Berrut, J.-P., *Numerische Lösung der Symmschen Integralgleichung durch Fourier-Methoden.* Diplomarbeit, ETH Zürich, 1976.

[BE84] Berrut, J.-P., *Ueber Integralgleichungen und Fourier-Methoden zur numerischen konformen Abbildung.* Thesis, ETH Zürich, in preparation.

[BE84a] Berrut, J.-P., *Equivalence of the method of Fourier and Nyström for solving Fredholm integral equations of the second kind.* Manuscript, submitted for publication, (1984).

[BE85] Berrut, J.-P., *A Fredholm integral equation of the second kind for conformal mapping.* To appear in J.Comp.Appl.Math. , (1985).

[CG75] Concus, P., and Golub, G.H., *A generalized conjugate gradient method for non-symmetric systems of equations.* Second International Symposium on Computing Methods in Applied Sciences and Engineering, Versailles, France Dec 15-19(1975), 10p, Lecture Notes in Economical and Mathematical Systems, Vol. 134, R. Glowinski and J.L. Lions, eds., Springer Verlag, Berlin, 1976.

[GA64] Gaier, D., *Konstruktive Methoden der konformen Abbildung.* Springer Tracts, Heidelberg, (1964).

[GU81] Gutknecht, M.H., *Solving Theodorsen's integral equation for conformal maps with the fast Fourier transform and various nonlinear iterative methods.* Numer.Math. 36, pp. 405-429, (1981).

[GU83] Gutknecht, M.H., *Numerical experiments on solving Theodorsen's integral equation for conformal maps with the fast Fourier transform and various nonlinear iterative methods.* SIAM J.Sci.Stat.Comput. 4, pp. 1-30, (1983).

[GU85] Gutknecht, M.H., *Numerical conformal mapping methods based on function conjugation.* To appear in J.Comp.Appl.Math. , (1985).

[HKK72] Hayes, J.K., Kahaner, D.K., and Kellner, R.G., *An improved method for numerical conformal mapping.* Math.Comp. 26, pp. 327-334, (1972).

[HKK75] Hayes, J.K., Kahaner, D.K., and Kellner, R.G., *A numerical comparison of integral equations of the first and second kind for conformal mapping.* Math.Comp. 29, pp. 512-521, (1975).

[HE74] Henrici, P., *Applied and computational complex analysis, Vol.* 1. Pure and Appl. Math., John Wiley, New York, (1974).

[HE79] Henrici, P., *Fast Fourier methods in computational complex analysis.* SIAM Rev.21, pp. 481-527, (1979).

[HE85] Henrici, P., *Applied and computational complex analysis, Vol.* 3. Pure and Appl. Math., John Wiley, New York, to appear.

[HO83] Hoidn, H.-P., *Die Kollokationsmethode angewandt auf die Symmsche Integralgleichung.* Diss. ETH Nr. 7365, Thesis, ETH Zürich, (1983).

[KS78] Kerzman, N., and Stein, E.M., *The Cauchy kernel, the Szegö kernel, and the Riemann mapping function.* Math.Ann. 236, pp. 85-93, (1978).

[KT85] Kerzman, N., and Trummer, M.R., *Numerical conformal mapping via the Szegö kernel.* To appear in J.Comp.Appl.Math. , (1985).

[LI17] Lichtenstein, L., *Zur konformen Abbildung einfach zusammenhängender schlichter Gebiete.* Arch.Math.Phys. 25, pp. 179-180, (1917).

[MU84] Müller, P., *Das Verfahren von Kerzman und Trummer zur konformen Abbildung einfach zusammenhängender Gebiete.* Diplomarbeit, ETH Zürich, 1984.

[SY66] Symm, G.T., *An integral equation method in conformal mapping.* Numer.Math.9, pp. 250-258, (1966).

[SZ39] Szegö, G., *Orthogonal polynomials.* AMS Colloquium Publications, Vol.23, New York, 1939.

[TR80] Trefethen, L.N., *Numerical computation of the Schwarz-Christoffel transformation.* SIAM J.Sci.Stat.Comput. 1, pp. 82-102, (1980).

[TRU84] Trummer, M.R., *Nyström's method versus Fourier type methods for the numerical solution of integral equations.* Technical report 84-23, Department of Computer Science, University of British Columbia, Vancouver, B.C., Canada, V6T 1W5, 1984.

[WA61] Warschawski, S., *On the differentiability at the boundary in conformal mapping.* Proc. AMS, 12, pp. 614-620, (1961).

[WE78] Wegmann, R., *Ein Iterationsverfahren zur konformen Abbildung.* Numer.Math. 30, pp. 453-466, (1978).

[WI78] Widlund, O., *A Lanczos method for a class of nonsymmetric systems of linear equations.* SIAM J.Numer.Anal. 15, pp. 801-812, (1978).