

## A NONOVERLAPPING DOMAIN DECOMPOSITION METHOD FOR SYMM'S EQUATION FOR CONFORMAL MAPPING\*

TOBIN A. DRISCOLL<sup>†</sup>

**Abstract.** Symm's equation is a first-kind integral equation for computing conformal maps of simply connected regions. The package CONFPACK solves Symm's equation by an indirect boundary element method using an accurate corner representation. This solution technique is extended here to include nonoverlapping domain decomposition. Degrees of freedom are introduced on one or more interfaces and different unknowns are used, leading to a system of second-kind equations. The corresponding linear system can be expressed in Schur complement form. The accurate treatment of corners is preserved in the new formulation. The results of serial and parallel MATLAB implementations of the new algorithm show significant speedups as the number of unknowns grows.

**Key words.** Symm's equation, conformal mapping, boundary elements, domain decomposition

**AMS subject classifications.** 30C30, 65N55, 65N38, 31A10

**PII.** S0036142997324162

**Introduction.** Symm's equation [15, 25] is one of the best-known integral equations underlying numerical methods for conformal mapping. Suppose  $\Omega$  is a bounded, simply connected region with a piecewise analytic boundary. Symm's method deals with the Riemann map  $g$  from  $\Omega$  (or  $\Omega^c$ ) to the unit disk. The nonlinear problem of finding  $g$  is reduced to the linear problem of finding a harmonic function satisfying certain Dirichlet boundary conditions, using a Fredholm integral equation of the first kind. Symm's method has been efficiently and elegantly implemented as the public-domain package CONFPACK [13]. An important feature of CONFPACK is an accurate treatment of the singularities introduced by corners in the boundary.

There has been little study of domain decomposition in conformal mapping [6, 9, 20, 21] and apparently none relating to integral equations for conformal mapping. This is true despite the fact that Symm's method is essentially an indirect boundary element method for Laplace's equation, and domain decomposition is common in boundary element methods generally [4, 16]. Any region with a long, narrow channel, or with structures on different scales, is a candidate for decomposition. The basic idea is to introduce a few extra unknowns on the interfaces between subregions in order to introduce sparse structure into the system matrix, corresponding to a localization of the boundary integrals. The resulting linear system is in Schur complement form and is ideal for parallelization. An additional benefit is that "deadwater" regions can be treated just once, whereas regions of greater activity can be resolved adaptively with a minimum of subdomain interactions. All this can be done while retaining the highly accurate treatment of corners used by CONFPACK.

In section 1 we introduce Symm's equation for the interior map and briefly summarize existing numerical methods, including CONFPACK. In section 2 we show how to incorporate domain decomposition into the integral equations. In section 3 we discuss the representation of unknowns on the boundary in a way that preserves the accurate treatment of corners. In sections 4 and 5 we fully describe the numerical method and

---

\*Received by the editors July 9, 1997; accepted for publication (in revised form) July 27, 1998; published electronically May 5, 1999.

<http://www.siam.org/journals/sinum/36-3/32416.html>

<sup>†</sup>Department of Applied Mathematics, University of Colorado, Boulder, CO 80309 (tad@colorado.edu).

compare its efficiency to the single-domain method. In section 6 we present the results of experiments with serial and parallel MATLAB implementations of the domain decomposition method for polygons, using both fixed and adaptive strategies. Finally, in section 7 we remark on a potential link to the crowding phenomenon.

**1. Symm's equation.** We begin with a summary of the mathematics behind Symm's method. This material can be found in more detail in, e.g., section 16.6 of [11].

Let  $\Omega$  be a simply connected open region in  $\mathbf{C}$  such that  $\Gamma = \partial\Omega$  is a finite collection of analytic curves. We do not allow  $\Gamma$  to have interior or exterior cusps. Let  $z_0 \in \Omega$  and let  $g$  be a conformal map of  $\Omega$  to  $D$ , the unit disk, that maps  $z_0$  to the origin. (The choice of  $g$  has one rotational degree of freedom.)

Because  $g'(z_0) \neq 0$ , the function  $g(z)/(z - z_0)$  is nonzero throughout  $\Omega$ . Thus  $g(z)/(z - z_0)$  has a logarithm in  $\Omega$ , so we can write

$$(1.1) \quad g(z) = (z - z_0) \exp(u(z) + iv(z)),$$

where  $u$  and  $v$  are real harmonic conjugates. Furthermore, since  $|g(z)| = 1$  for  $z \in \Gamma$ , the function  $u$  is a solution of the boundary-value problem

$$(1.2a) \quad \Delta u(z) = 0, \quad z \in \Omega,$$

$$(1.2b) \quad u(z) = -\log|z - z_0|, \quad z \in \Gamma.$$

We can express  $u$  in terms of a single-layer potential [4]:

$$(1.3) \quad u(z) = -\frac{1}{2\pi} \int_{\Gamma} \sigma(\zeta) \log|z - \zeta| d\Gamma.$$

By evaluating this expression on  $\Gamma$ , we arrive at *Symm's equation* for interior mapping:

$$(1.4) \quad \log|z - z_0| = \frac{1}{2\pi} \int_{\Gamma} \sigma(\zeta) \log|z - \zeta| d\Gamma, \quad z \in \Gamma.$$

This is a Fredholm integral equation of the first kind.

Symm's equation can always be solved uniquely for the potential  $\sigma$ , provided the capacity of  $\Gamma$  is not equal to 1 [8]. In fact,  $\sigma$  has a special meaning—it is closely related to the *boundary correspondence function*, which is defined as follows. Suppose that  $\zeta = \zeta(s)$ ,  $0 \leq s \leq L$ , is a piecewise analytic parameterization of the boundary  $\Gamma$ . The boundary correspondence function is defined to be

$$(1.5) \quad \theta(s) = \arg(g(\zeta(s)))$$

for any choice of the argument that makes  $\theta(s)$  continuous. The potential  $\sigma$ , regarded as a function of  $s$ , is

$$(1.6) \quad \sigma(s) = |\zeta'(s)|\theta'(s),$$

where  $\zeta'(s)$  can be taken as an arbitrary finite number at a corner of  $\Gamma$ . If arc length is used to parameterize the boundary,  $\sigma$  and  $\theta'$  are identical. Equation (1.6) is useful in the construction of accurate numerical methods, as we shall see. Once  $\sigma$  is known, the conjugate pair  $u$  and  $v$  can be found (up to an additive constant in  $v$ ) by replacing  $\log|z - \zeta|$  by  $\log(z - \zeta)$  in (1.3), taking care to make the logarithm continuous in  $\Omega$ . In light of (1.1), this determines  $g$  (up to a rotation).

There are many other integral equation formulations for interior conformal mapping [7, 11], including those of Berrut [3], Warschawski [28], Henrici [11], and those based on the Szegő kernel [17, 19]. There are also methods for the inverse mapping, including conjugate function approaches [10] such as Theodorsen's method [26] and Wegmann's method [29].

To approximate the solution of Symm's equation (1.4), the typical procedure is to discretize the boundary, choose a representation for the approximation to  $\sigma$ , and replace the integral equation by a linear system of algebraic equations. This is the approach originally taken by Symm [25], who chose a piecewise constant representation for  $\sigma$ . To improve accuracy, Hough [12] and Hough and Papamichael [14] use information about the behavior of  $\theta'(s)$  at the corners of  $\Gamma$  to demonstrate that Jacobi polynomials and weights give the correct singular behavior of  $\sigma$  at the corners. This is the approach implemented in CONFPACK [13], and it allows high accuracy of the solution even near corners.

These numerical techniques can be viewed as "indirect" boundary element methods [4]. In some contexts, indirect methods are not as popular as direct methods because the potential  $\sigma$  obtained has no direct physical meaning. However, in this context, the use of  $\sigma$  allows the representation of corner singularities by the Jacobi weight method.

Alternative numerical approaches are possible. Berrut [2] and Reichel [24] use FFT-based iterative methods to solve (1.4). Amano [1] uses a charge-simulation method in which point charges outside the domain are used to approximate the solution to (1.2).

**2. Domain decomposition.** If two portions of  $\Omega$  are separated by a long, thin channel, the influence of data at one end of the channel is exponentially weak at the other end. We seek to improve efficiency by not having to compute all such weak interactions explicitly, as is required in (1.4). Our mechanism will be to introduce new quantities on an artificial interface in the channel, creating a nonoverlapping domain decomposition. All interaction between subdomains will be localized to the interface. In terms of the linear system resulting from discretization, the new unknowns along the interface give the system matrix a block sparsity structure. In fact, the structure we shall arrive at is the Schur complement form [5], which yields computational savings in both assembly and solution.

Accordingly, we now let  $\Omega$  be divided into two nonoverlapping regions  $\Omega_1$  and  $\Omega_2$  and let  $\Gamma_1$  and  $\Gamma_2$  be the positively oriented boundaries of these subregions. The interface  $\Gamma_{12}$  is equal to  $\Gamma_1 \cap \Gamma_2$  and is taken to be oriented positively with respect to  $\Omega_1$ . See Figure 2.1.

A natural first idea would be to introduce potentials on the interface as we did on the boundary. A potential on each "side" of  $\Gamma_{12}$  would be mathematically necessary to keep the system square. Thus, for  $k = 1, 2$  we could define  $\sigma_k$  on all of  $\Gamma_k$ , define a function  $u_k$  on  $\Omega_k$  by (1.3), and supplement (1.4) with equations requiring the  $u_k$  and their normal derivatives to match along  $\Gamma_{12}$ .

This is quite easy to do with, say, piecewise constant boundary elements, and the resulting linear system is indeed of the desired form. But the  $\sigma_k$  have no obvious relationship to the global potential  $\sigma$ . One can still use the individual potentials to compute  $g$  in the subregions, but the boundary correspondence function will not be known. More seriously, the new potentials  $\sigma_k$  do not retain the same behavior at the corners as  $\sigma$ , and therefore the specialized treatment of corners used by CONFPACK cannot be applied. This makes calculations of high accuracy difficult. We now derive

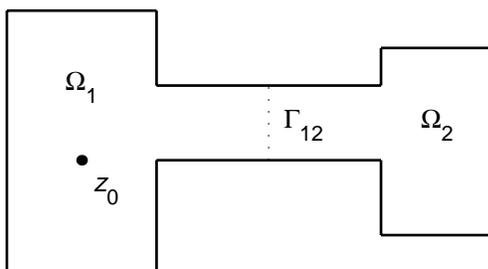


FIG. 2.1. Typical nonoverlapping domain decomposition.

an alternative formulation that avoids this problem. In what follows, we adopt the notation

$$(2.1) \quad G(z, \zeta) = \frac{1}{2\pi} \log |z - \zeta|$$

for the fundamental solution for the Laplace operator.

Let

$$(2.2) \quad \tilde{u}(z) = u(z) + \log |z - z_0|.$$

(Recall that  $z_0$  is the conformal center of  $\Omega$ .) From (1.2) we see that

$$(2.3a) \quad \Delta \tilde{u}(z) = 2\pi \delta(z - z_0), \quad z \in \Omega,$$

$$(2.3b) \quad \tilde{u}(z) = 0, \quad z \in \Gamma.$$

Hence  $\tilde{u}(z)$  is a multiple of the Green's function for  $\Omega$  with pole at  $z_0$ . We shall now apply a direct integral equation to solve for  $\tilde{u}$  [16]. Let  $z \in \Omega$ . By Green's second identity,

$$(2.4) \quad \int_{\Omega} \varphi \Delta \psi \, d\Omega - \int_{\Omega} \psi \Delta \varphi \, d\Omega = \int_{\Gamma} \varphi \frac{\partial \psi}{\partial n} \, d\Gamma - \int_{\Gamma} \psi \frac{\partial \varphi}{\partial n} \, d\Gamma,$$

with  $\psi = \tilde{u}(\zeta)$  and  $\varphi = G(z, \zeta)$ , we have

$$\log |z - z_0| - \tilde{u}(z) = \int_{\Gamma} G(z, \zeta) \frac{\partial \tilde{u}}{\partial n}(\zeta) \, d\Gamma.$$

Now suppose  $z$  approaches a point on  $\Gamma$ . The  $\tilde{u}(z)$  term on the left will need to be multiplied by a constant depending on the angle subtended by  $\Gamma$  at the limit point, but in any case  $\tilde{u}(z) \rightarrow 0$ . Thus we have

$$(2.5) \quad \log |z - z_0| = \int_{\Gamma} G(z, \zeta) \frac{\partial \tilde{u}}{\partial n}(\zeta) \, d\Gamma.$$

By comparison with (1.4), we see that  $\partial \tilde{u} / \partial n$  is identical to the potential  $\sigma$ .

Assume without loss of generality that  $z_0 \in \Omega_1$ . By applying Green's identity (2.4) to the subregions  $\Omega_1$  and  $\Omega_2$ , inserting (2.3), and letting  $z \in \Gamma$ , we arrive at the system

$$(2.6a) \quad c_1(z) \tilde{u}_1(z) - \log |z - z_0| = \int_{\Gamma_1} \tilde{u}_1(\zeta) \frac{\partial G}{\partial n}(z, \zeta) \, d\Gamma_1 - \int_{\Gamma_1} G(z, \zeta) \frac{\partial \tilde{u}_1}{\partial n}(\zeta) \, d\Gamma_1,$$

$$(2.6b) \quad c_2(z) \tilde{u}_2(z) = \int_{\Gamma_2} \tilde{u}_2(\zeta) \frac{\partial G}{\partial n}(z, \zeta) \, d\Gamma_2 - \int_{\Gamma_2} G(z, \zeta) \frac{\partial \tilde{u}_2}{\partial n}(\zeta) \, d\Gamma_2,$$

where  $\tilde{u}_{1,2}$  represent the values of  $\tilde{u}$  on the two subregions and  $2\pi c_k(z)$  is the interior angle subtended by  $\Gamma_k$  about  $z$ . (For  $z$  on a smooth portion of  $\Gamma_k$ ,  $c_k(z) = \frac{1}{2}$ .) Of course,  $\tilde{u}_1$  and  $\tilde{u}_2$  are not independent; they are related by

$$(2.7a) \quad \tilde{u}_1 = \tilde{u}_2,$$

$$(2.7b) \quad \frac{\partial \tilde{u}_1}{\partial n} = -\frac{\partial \tilde{u}_2}{\partial n} \quad \text{on } \Gamma_{12}.$$

We have replaced the Fredholm equation of the first kind (1.4) with a system of Fredholm equations of the second kind. The kernel  $\partial G/\partial n$  is equal to the *Neumann kernel* [11] given by

$$\frac{\partial G}{\partial n}(z, \zeta(s)) = \frac{1}{2\pi} \operatorname{Im} \left( \frac{\zeta'(s)}{|\zeta'(s)|(\zeta(s) - z)} \right)$$

for any parameterization  $\zeta(s)$  of a boundary. This kernel is nominally more singular than the logarithmic one arising from the fundamental solution. However,  $\tilde{u}_i(\zeta) = 0$  for  $\zeta \in \Gamma_i \setminus \Gamma_{12}$ ; furthermore, if we assume that the interface is a straight line segment, then the kernel will be zero on  $\Gamma_{12}$  when  $\zeta, z \in \Gamma_{12}$ . Hence no singular integrals need be computed. Note that this essentially returns (2.6) to a system of equations of the first kind.

**3. Representation of solutions.** We now recall and extend the representation of unknowns in Symm's equation described by Hough [12, 13]. Let  $\zeta(s)$  be a local parameterization of a typical analytic arc of  $\Gamma$  with endpoints  $\zeta(-1) = z_-$  and  $\zeta(1) = z_+$  such that  $|\zeta'(s)|$  is bounded and never zero. Let  $\Gamma$  have interior angles  $\pi(1 + \beta_-)$  and  $\pi(1 + \beta_+)$  at the endpoints, respectively. We represent  $\partial \tilde{u}/\partial n$  on this arc by

$$(3.1) \quad \frac{\partial \tilde{u}}{\partial n}(\zeta(s)) = \phi(s)w(s) = \phi(s)(1+s)^{\alpha_-}(1-s)^{\alpha_+},$$

where the *Jacobi indices*  $\alpha_{\pm}$  are given by

$$\alpha_{\pm} = -1 + \frac{1}{1 + \beta_{\pm}}.$$

Hough [12] has shown that while  $\partial \tilde{u}/\partial n$  is only in  $L^2[-1, 1]$  and is infinite at a reentrant corner,  $\phi$  is Hölder continuous on  $[-1, 1]$  with index greater than  $\frac{1}{2}$ .

We approximate  $\phi$  by

$$(3.2) \quad \phi(s) \approx \sum_{j=0}^{m-1} q_j P_j(s),$$

where the  $P_j$ 's are Jacobi polynomials associated with the weight function  $w(s)$ . To match the  $m$  unknown coefficients, we choose  $m$  collocation points  $z_j$ . These are  $\zeta(s_j)$ ,  $j = 1, \dots, m$ , where  $s_j$  is the  $j$ th zero of  $P_m$ .

We also need a representation of  $\tilde{u}$  on the interface  $\Gamma_{12}$ . Let  $\zeta(s)$  now be a parameterization of the interface. It seems natural to suppose that the appropriate weight function will be as in (3.1) with the Jacobi indices increased by 1. Hence we define

$$(3.3) \quad \tilde{u}(\zeta(s)) = \psi(s)(1+s)^{1+\alpha_-}(1-s)^{1+\alpha_+}.$$

We now prove that  $\psi$  is as smooth as  $\phi$ . This theorem is the analog of Proposition 2.1 in [12].

**THEOREM 3.1.** *With the notation of this section, the quotient function  $\psi(s)$  is in  $H^\nu[-1, 1]$ , where  $\nu$  is given by*

$$\nu = \begin{cases} \min((1 + \beta_-)^{-1}, (1 + \beta_+)^{-1}) & \text{if } \max(\beta_-, \beta_+) > 0, \\ 1 & \text{if } \max(\beta_-, \beta_+) < 0, \\ 1 - \epsilon & \text{for any } \epsilon > 0, \text{ otherwise.} \end{cases}$$

*Proof.* Let  $\pi\gamma$  be the interior angle at the left endpoint  $z_-$ ; i.e.,  $\gamma = 1 + \beta_-$ . From [22] we have two expansions of  $g(z)$  near  $z_-$ :

(i) If  $\gamma = p/q$ , with  $p$  and  $q$  relatively prime, then

$$(3.4a) \quad g(z) - g(z_-) = \sum_{j,k,l} B_{jkl} (z - z_-)^{j+k/\gamma} (\log(z - z_-))^l,$$

where  $j, k$ , and  $l$  run over integers such that  $j \geq 0, 1 \leq k \leq p$ , and  $0 \leq l \leq q$  and where  $B_{010} \neq 0$ .

(ii) If  $\gamma$  is irrational, then

$$(3.4b) \quad g(z) - g(z_-) = \sum_{j,k} B_{jk} (z - z_-)^{j+k/\gamma},$$

where  $j \geq 0, k \geq 1$ , and  $B_{01} \neq 0$ .

Note from (1.1) and (2.2) that  $\tilde{u}(z) = \log |g(z)|$ . Hence for  $z$  near  $z_-$ ,

$$\tilde{u}(z) = \log |g(z_-) + S(z)| \leq |S(z)| + O(|S(z)|^2),$$

where  $S(z)$  represents the sum on the right-hand side of the appropriate member of (3.4). We set  $z = \zeta(s)$  and observe that  $|\zeta(s) - \zeta(-1)|$  and  $(s + 1)$  have the same asymptotic behavior. We wish to factor out  $(s + 1)^{1/\gamma}$  from  $S(\zeta(s))$  and show that the quotient is appropriately well behaved for  $s$  near  $-1$ ; i.e., it is in  $H^\nu[-1, -1 + \delta]$  for some  $\delta > 0$ . We consider three cases.

(a) Suppose  $1 < \gamma < 2$ . If  $\gamma$  is irrational, then clearly  $(s + 1)^{1/\gamma}$  is the most singular term after the factorization. But the same is true if  $\gamma = p/q$ , since necessarily  $p > 1$  and the  $B_{020}$  term appears in (3.4a). We observe that  $(s + 1)^{1/\gamma}$  is in  $H^{1/\gamma}[-1, -1 + \delta]$  for any  $\delta > 0$ .

(b) Suppose  $\gamma = 1$ . Then from (3.4a) we see that the most singular term in the quotient is  $(s + 1) \log(s + 1)$ , which is in  $H^{1-\epsilon}$  for any  $\epsilon > 0$ .

(c) Suppose  $0 < \gamma < 1$ . If  $\gamma$  is irrational, again it is clear that  $(1 + s)^{1/\gamma}$  is the most singular term. If  $\gamma = p/q$ , then  $(1 + s)$  is the most singular term, because  $q > 1$  and the logarithmic terms in (3.4a) cannot appear until  $j \geq 2$ . In either case, the quotient is in  $H^1$ .

Similar reasoning applies near  $z_+$ . □

The natural representation of  $\psi$  is analogous to (3.2), with the Jacobi polynomials being taken with respect to the new weight function in (3.3), that is, with Jacobi indices increased by 1.

We have seen that on each analytic arc of  $\Gamma_k$ , we have unknown coefficients from the Jacobi expansion (3.2) defining  $\partial\tilde{u}/\partial n$  on that arc. These coefficients are globally

collected into a vector  $q^{(k)}$ . Let us suppose the elements of  $q^{(k)}$  are ordered such that the coefficients relating to values on  $\Gamma_{12}$  are last, so that

$$q^{(k)} = \begin{bmatrix} q_p^{(k)} \\ q_s^{(k)} \end{bmatrix}.$$

(The subscripts are meant to convey “private” and “shared.”) Let the length of  $q_p^{(k)}$  be  $N_{kp}$ , the length of  $q_s^{(k)}$  be  $N_{ks}$ , and  $N_k = N_{kp} + N_{ks}$ . We also have the  $N_{ks}$ -vector  $v_s^{(k)}$  defining  $\tilde{u}_k$  on  $\Gamma_{12}$ . (For symmetry of notation, we can define  $v_p^{(k)}$ , all of whose entries are zero.) Similarly, we define  $z^{(k)}$  as a vector of  $N_{kp}$  collocation points on  $\Gamma_k \setminus \Gamma_{12}$  and  $N_{ks}$  collocation points on  $\Gamma_{12}$ . We require that  $N_{1s} = N_{2s} = N_s$  and that corresponding entries of the  $s$ -vectors have the same meaning globally on  $\Gamma_{12}$ .

**4. Numerical method.** We now present the discretized form of (2.6). Each integral on the right-hand side of the equations is replaced by a matrix operating on  $q^{(k)}$  or  $v^{(k)}$ . For example, the second integral in (2.6a) involves the logarithmic kernel  $G(z, \zeta)$  against the normal derivative of  $\tilde{u}_1$  around  $\Gamma_1$ . The integration is performed for each basis function in the representation (3.2) and for  $z$  ranging over the collocation points in  $z^{(1)}$ . The result is a matrix  $B^{(1)}$  that operates on  $q^{(1)}$  to produce a vector of integral evaluations for the discrete representation at the collocation points. We have a matrix  $B^{(2)}$  playing the same role on  $\Gamma_2$ , and we have  $A^{(1)}$  and  $A^{(2)}$  operating on  $v^{(1)}$  and  $v^{(2)}$  to perform integrations of the discrete  $\tilde{u}_1$  and  $\tilde{u}_2$ . We also need approximations  $\tilde{I}^{(k)}$  of the identity operating on  $v^{(k)}$  for the left-hand sides of (2.6). Each column of  $\tilde{I}^{(k)}$  is the evaluation of one basis function in the representation of  $v^{(k)}$  at the collocation points  $z^{(k)}$ .

All of these matrices can be written in block form according to the private/shared partitioning. The result for  $k = 1, 2$  is

$$\begin{bmatrix} B_{pp}^{(k)} & B_{ps}^{(k)} \\ B_{sp}^{(k)} & B_{ss}^{(k)} \end{bmatrix} \begin{bmatrix} q_p^{(k)} \\ q_s^{(k)} \end{bmatrix} - \begin{bmatrix} A_{pp}^{(k)} - \frac{1}{2}\tilde{I}_{pp}^{(k)} & A_{ps}^{(k)} \\ A_{sp}^{(k)} & A_{ss}^{(k)} - \frac{1}{2}\tilde{I}_{ss}^{(k)} \end{bmatrix} \begin{bmatrix} 0 \\ v_s^{(k)} \end{bmatrix} = \begin{bmatrix} b_p^{(k)} \\ b_s^{(k)} \end{bmatrix},$$

where  $b^{(1)} = \log |z^{(1)} - z_0|$  and  $b^{(2)} = 0$ . The only singular integrals that arise are those needed in the  $B$  matrices. These involve boundary integrals with the kernel  $\log |z - \zeta|$  and are performed as in CONFACK, by splitting out the singularity by subtraction and making calls to QUADPACK [23]. Nonsingular integrals are calculated by compound Gauss–Jacobi integration. The integrals for the  $A$  matrices need be computed only over the interface  $\Gamma_{12}$ . As was pointed out above, no singular integrals are necessary as long as  $\Gamma_{12}$  is a line segment, since in that case  $A_{ss}^{(k)} = 0$ .

We now combine the subdomain problems into a global system. Relations (2.7) imply

$$\begin{aligned} v_s^{(1)} &= v_s^{(2)}, \\ q_s^{(1)} &= -q_s^{(2)}. \end{aligned}$$

Altogether we have

$$(4.1) \quad \begin{bmatrix} B_{pp}^{(1)} & 0 & B_{ps}^{(1)} & -A_{ps}^{(1)} \\ 0 & B_{pp}^{(2)} & -B_{ps}^{(2)} & -A_{ps}^{(2)} \\ B_{sp}^{(1)} & 0 & B_{ss}^{(1)} & \frac{1}{2}\tilde{I}_{ss}^{(1)} \\ 0 & B_{sp}^{(2)} & -B_{ss}^{(2)} & \frac{1}{2}\tilde{I}_{ss}^{(2)} \end{bmatrix} \begin{bmatrix} q_p^{(1)} \\ q_p^{(2)} \\ q_s^{(1)} \\ v_s^{(1)} \end{bmatrix} = \begin{bmatrix} b_p^{(1)} \\ 0 \\ b_s^{(1)} \\ 0 \end{bmatrix}.$$

This matrix is in Schur complement form. The advantage of this form is significant when  $N_s$  is much smaller than  $N_{1p}$  and  $N_{2p}$ . In this case, the system (4.1) is solved by first solving for  $q_p^{(1)}$  and  $q_p^{(2)}$  from the first and second block rows, respectively, and then substituting into the equations for the interface variables:

$$\begin{aligned} & \left( \begin{bmatrix} B_{ss}^{(1)} & \frac{1}{2} \tilde{I}_{ss}^{(1)} \\ -B_{ss}^{(2)} & \frac{1}{2} \tilde{I}_{ss}^{(2)} \end{bmatrix} - \begin{bmatrix} B_{sp}^{(1)} \\ 0 \end{bmatrix} [B_{pp}^{(1)}]^{-1} \begin{bmatrix} B_{ps}^{(1)} & -A_{ps}^{(1)} \\ -B_{ps}^{(2)} & -A_{ps}^{(2)} \end{bmatrix} \right) \begin{bmatrix} q_s^{(1)} \\ v_s^{(1)} \end{bmatrix} \\ & = \begin{bmatrix} b_s^{(1)} - B_{sp}^{(1)} [B_{pp}^{(1)}]^{-1} b_p^{(1)} \\ 0 \end{bmatrix}. \end{aligned}$$

Once  $q_s^{(1)}$  and  $v_s^{(1)}$  have been found, they can be substituted back to solve for  $q_p^{(1)}$  and  $q_p^{(2)}$ .

Approximations to  $\tilde{u}_1(z)$  and  $\tilde{u}_2(z)$  for  $z \in \Omega$  could now be obtained from (2.6), with  $c_i(z) = 1$ , by integration around  $\Gamma_1$  or  $\Gamma_2$ . However, we also need the harmonic conjugate  $v(z)$  of  $u(z)$  in order to compute  $g$ . It is therefore convenient to augment the integration kernels in (2.6) by their harmonic conjugates:

$$(4.2) \quad u_k(z) + iv_k(z) = l_k(z) + \frac{1}{2\pi} \int_{\Gamma_k} \left[ \frac{-it_k(\zeta)}{\zeta - z} \tilde{u}_k(\zeta) - \log(z - \zeta) \frac{\partial \tilde{u}_k}{\partial n}(\zeta) \right] d\Gamma_k,$$

where  $t_k(\zeta)$  is the unit tangent to  $\Gamma_k$  at  $\zeta$ ,  $l_1(z) = 0$ , and  $l_2(z) = -\log(z - z_0)$ . If  $z$  is pathologically close to  $\Gamma$  or  $\Gamma_{12}$ , we can proceed as in CONFPACK by analytically continuing the boundary parameterization and the representation of unknowns [13].

The domain decomposition procedure is straightforwardly generalized to regions decomposed into three or more subdomains. One forms Schur complements for the subregions adjacent to each interface, solves for the interface quantities simultaneously, and substitutes back to find the primary quantities.

We do not see an application of domain decomposition to inverting the map  $g$ . CONFPACK treats the inverse map analogously to the forward map by means of an inverse boundary correspondence function on the unit circle. Naturally, one could simply use the computed  $\partial \tilde{u} / \partial n$  in the same fashion. Alternatively, one could attempt to use a nonlinear iteration on the forward map.

**5. Computational efficiency.** The domain decomposition procedure requires  $O(N_{1p}^2 + N_{2p}^2 + N_s(N_{1p} + N_{2p}))$  flops to form a single linear system and  $O(N_{1p}^3 + N_{2p}^3 + N_s(N_{1p}^2 + N_{2p}^2 + N_s^2))$  to solve the system by direct Schur complementation. The undecomposed version requires  $O((N_{1p} + N_{2p})^2)$  flops to form and  $O((N_{1p} + N_{2p})^3)$  to solve one system. Asymptotically, the domain decomposition method takes a constant fraction of the time needed by the direct solution, assuming a uniform growth in  $N_{1p}$ ,  $N_{2p}$ , and  $N_s$ . In practice, forming the system is usually the most expensive step.

A practical method for solving Symm's equation must be adaptive. CONFPACK, for example, initially chooses a uniform distribution of unknowns along the boundary arcs, solves the discretization of Symm's equation, and adjusts the number of unknowns on each arc based on the size of the coefficients in the orthogonal expansion (3.2). The system is updated and solved again, leading to an iterative process. During

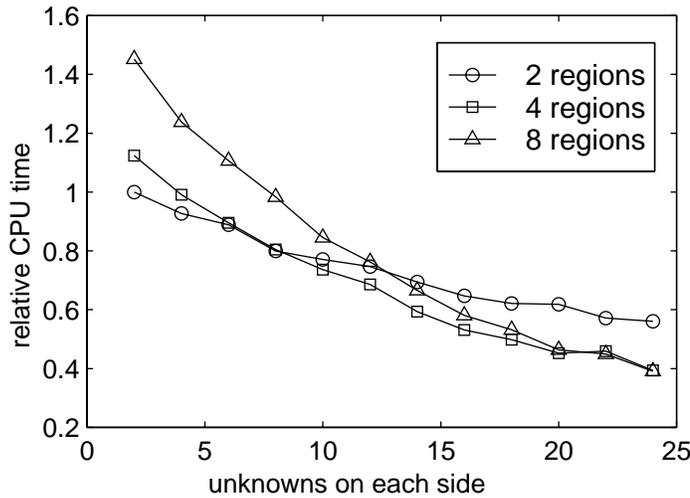


FIG. 6.1. Relative CPU time for forming and solving the decomposed Symm's equation for a rectangle. The timings for decompositions into 2, 4, and 8 subdomains are shown as a fraction of the time needed by the undecomposed version.

an update, only those columns and rows corresponding to new unknowns and collocation points need be computed, but the entire linear system must be solved from scratch.

Domain decomposition allows adaptation to proceed more efficiently. Consider the polygon in Figure 2.1, with the channel made more narrow. CONFPAK quickly detects that relatively few unknowns are needed on the arcs of  $\Gamma_2$ , but it must repeatedly solve for a linear system including those unknowns as it adds resolution to  $\Gamma_1$ . If a domain decomposition method is used, the block  $B_{pp}^{(2)}$  need be formed and  $LU$ -factored only once. Hence a good choice of decomposition can reduce the amount of computational overhead required by deadwater regions in the adaptive solution.

Parallelization is trivially achieved in both the system formation and private block inversion stages that consume most of the computational effort. The load balancing depends on the evenness of the distribution of boundary unknowns among the subregions. This can be a delicate matter for an adaptive strategy, in which unknowns tend to concentrate near the conformal center.

Evaluation of the map at points in  $\Omega_k$  requires  $O(N_{kp} + N_s)$  flops, as opposed to  $O(N_{1p} + N_{2p})$ . Over many evaluations, this can be a substantial savings, especially within deadwater regions where  $N_{kp}$  is ideally rather small.

**6. Numerical experiments.** The domain decomposition version of Symm's method described above has been implemented in MATLAB, except for singular integration against the logarithmic kernel, for which QUADPACK is used. For simplicity of coding, the implementation so far accepts only polygonal domains.

We begin with nonadaptive computations performed serially on a SPARC-10 workstation. Our first example is a  $16 \times 1$  rectangle with 34 vertices distributed evenly around the boundary. We consider decompositions into  $d = 2$ ,  $d = 4$ , and  $d = 8$  equally sized subregions by splitting in the long direction. Figure 6.1 shows the CPU time required in each case, expressed as a fraction of the time needed for

a single domain solution, as a function of the number of unknown Jacobi polynomial coefficients per side or interface. Even without parallelization, we see the advantage of forming and solving several smaller systems rather than a single large one. The  $d = 4$  and  $d = 8$  versions are slower than the others for small systems, because of additional overhead, but overtake the other versions as the number of unknowns grows.

In Figure 6.2 we show results for an early stage in the construction of the Koch snowflake. The region is decomposed into seven parts, one of which has exclusively interface unknowns. Again the relative timing improves as more unknowns are added. (These calculations make no use of symmetry, which could render the domain decomposition superfluous in this case.)

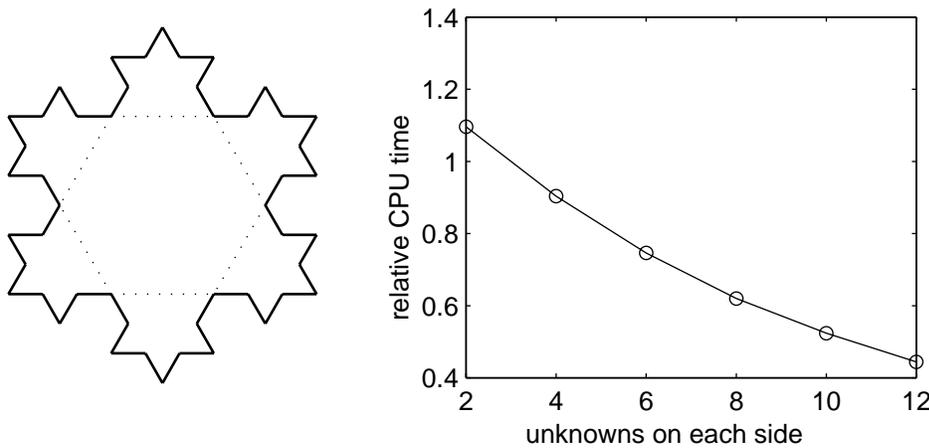


FIG. 6.2. Domain decomposition and relative CPU time for the decomposed algorithm on a partially constructed Koch curve.

We now consider adaptive computations, performed in parallel on an IBM SP2.<sup>1</sup> In the adaptive strategy, the solution is initially obtained by a uniform distribution of unknowns per side. Since the Jacobi polynomials are orthogonal with respect to the weight functions used, the size of the coefficients on a side is assumed to be a fair indicator of the error on that side. The number of unknowns is updated based on an exponential fit of the coefficients, and the system or systems are updated and solved. The individual subdomain matrix computations, inversions, and Schur complementation are performed in parallel, while the interface solution, subdomain back-substitutions, and adaptive refinement decisions are performed by a master process, since these steps take negligible time. We measure the total elapsed time required for a solution at each requested accuracy and report speedup and parallel efficiency. Figures 6.1 and 6.2 suggest that a parallel efficiency greater than 100% is possible because the domain decomposition alone lends a speedup independent of the parallel processing.

We consider the  $16 \times 1$  rectangle (depicted in Figure 6.3) with the conformal center one unit from the left edge and a quarter unit from the bottom edge. We put a single vertical interface two units from the left edge to achieve much better load balancing than that achieved by placing the interface in the middle of the rectangle. In

<sup>1</sup>The parallel MATLAB computations were done in the MultiMATLAB environment [27].

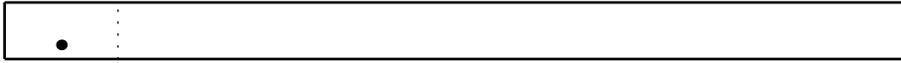


FIG. 6.3. Rectangular region used in computational tests. The aspect ratio of the rectangle is 16, the conformal center is one unit to the right and a quarter unit above the lower left corner, and the interface is two units from the left edge.

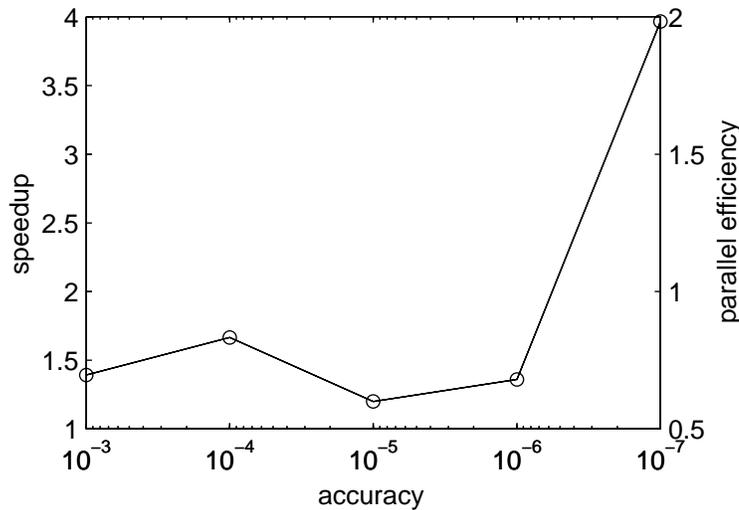


FIG. 6.4. Speedup and efficiency for parallel adaptive Symm computations on the rectangle in Figure 6.3. The parallel efficiency exceeds 100% because the serial computations for the decomposed region can be faster than for the undecomposed case.

Figure 6.4 we display the speedup and efficiency for several target accuracies. Because the rectangle map is known exactly, we can verify that the requested accuracy is met; in fact, it is exceeded in all of these examples.

In Figure 6.5 we display results for a spiral region with 36 vertices. A decomposition into 5 subdomains is shown; a decomposition into 3 subdomains is obtained by eliminating the interfaces closest to the conformal center. Initially the subdomain containing the conformal center has relatively few unknowns, but the adaptive refinement quickly pushes the distribution into more of a balance. Parallel efficiency for 3 domains rises and then falls, presumably due to poor balancing. The efficiency for 5 domains starts at about 25% and increases to 75%.

**7. On crowding.** A well-known phenomenon in conformal mapping goes by the name of *crowding* [18]. When mapping regions have long, thin channels, such as a rectangle, the images of the vertices are separated by a distance that is exponentially small in the aspect ratio of the channel.

In terms of Symm's equation,  $g'$  and  $\theta'$  can become exceedingly small in a crowded region. For computing  $g$ , this normally is not regarded as a problem because  $\theta'$  can still be found relatively accurately in a *global* sense, i.e., compared to  $O(1)$ . However, the inverse map  $g^{-1}$  is frequently important in applications. Unless  $g$  is resolved in a *locally* accurate sense,  $g^{-1}$  cannot be computed accurately in a crowded area.

The domain decomposition procedure should allow us to circumvent this problem.

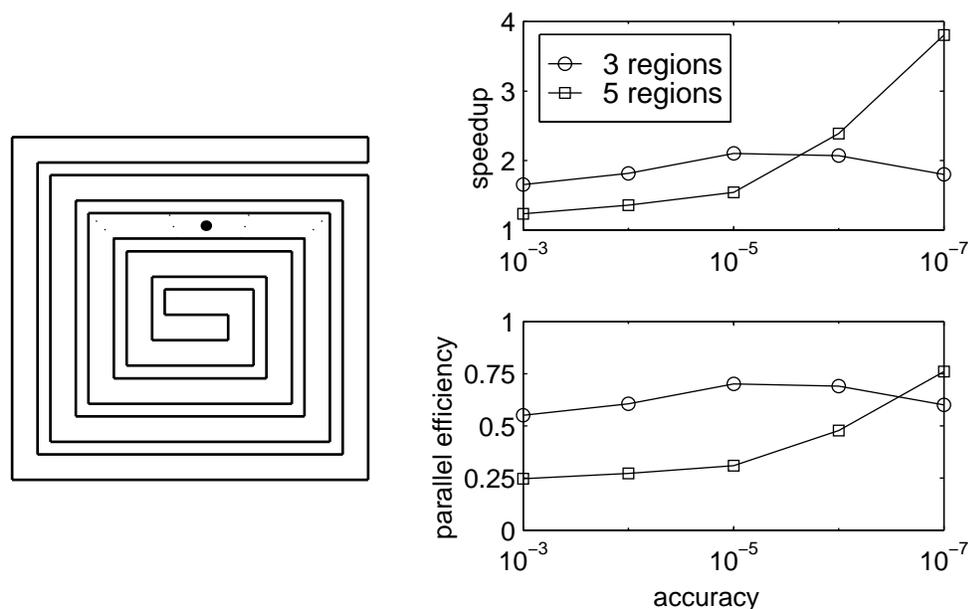


FIG. 6.5. Speedup and efficiency for a spiral with 36 vertices.

For example, consider the rectangle of Figure 6.3. The solution on the right edge of the rectangle should have order of magnitude roughly  $\exp(-16\pi) \approx 10^{-22}$ . Since the solution on the left edge is  $O(1)$  or  $O(0.1)$ , there is no hope of resolving the right edge in double precision with a single domain solution. But if we introduce unknowns along an interface halfway along the rectangle, where the solution is of an intermediate size, each individual subproblem should span a reasonable range of magnitudes. To exploit the situation appropriately, one probably needs to introduce scaling on the individual interfaces, because all interfaces are solved in a single system. The application of the domain decomposition method in the presence of crowding is a subject for future study.

**Acknowledgments.** The author thanks Nick Trefethen and Lars Wahlbin for their insights.

#### REFERENCES

- [1] K. AMANO, *A charge simulation method for the numerical conformal mapping of interior, exterior, and doubly-connected domains*, J. Comput. Appl. Math., 53 (1994), pp. 353–370.
- [2] J.-P. BERRUT, *Numerische Lösung der Symmschen Integralgleichung durch Fourier-Methoden*, Master's thesis, ETH Zürich, 1976.
- [3] J.-P. BERRUT, *A Fredholm integral equation of the second kind for conformal mapping*, J. Comput. Appl. Math., 14 (1985), pp. 99–110.
- [4] C. A. BREBBIA, J. C. F. TELLES, AND L. C. WROBEL, *Boundary Element Techniques: Theory and Applications in Engineering*, Springer, Berlin, New York, 1984.
- [5] T. F. CHAN AND T. P. MATHEW, *Domain Decomposition Algorithms*, Acta Numer., Cambridge University Press, Cambridge, UK, 1994, pp. 61–143.
- [6] T. A. DRISCOLL AND S. A. VAVASIS, *Numerical conformal mapping using cross-ratios and Delaunay triangulation*, SIAM J. Sci. Comput., 19 (1998), pp. 1783–1803.
- [7] D. GAIER, *Konstruktive Methoden der konformen Abbildung*, Springer-Verlag, Berlin, 1964.

- [8] D. GAIER, *Integralgleichungen erster Art und konforme Abbildung*, Math. Z., 147 (1976), pp. 113–129.
- [9] D. GAIER AND W. HAYMAN, *On the computation of modules of long quadrilaterals*, Constr. Approx., 7 (1991), pp. 453–467.
- [10] M. H. GUTKNECHT, *Numerical conformal mapping methods based on function conjugation*, J. Comput. Appl. Math., 14 (1986), pp. 31–77.
- [11] P. HENRICI, *Applied and Computational Complex Analysis*, vol. 3, John Wiley, New York, 1986.
- [12] D. M. HOUGH, *Conformal Mapping and Fourier–Jacobi Approximations*, IPS Research Report 89-06, ETH Zürich, 1989.
- [13] D. M. HOUGH, *User’s Guide to CONFPACK*, IPS Research Report 90-11, ETH Zürich, 1990.
- [14] D. M. HOUGH AND N. PAPAMICHAEL, *An integral equation method for the numerical conformal mapping of interior, exterior, and doubly-connected domains*, Numer. Math., 41 (1983), pp. 287–307.
- [15] M. A. JASWON AND G. T. SYMM, *Integral Equation Methods in Potential Theory and Electrostatics*, Academic Press, New York, 1977.
- [16] J. H. KANE, *Boundary Element Analysis in Engineering Continuum Mechanics*, Prentice–Hall, Englewood Cliffs, NJ, 1994.
- [17] N. KERZMAN AND M. R. TRUMMER, *Numerical conformal mapping via the Szegő kernel*, J. Comput. Appl. Math., 14 (1986), pp. 125–142.
- [18] R. MENIKOFF AND C. ZEMACH, *Methods for numerical conformal mapping*, J. Comput. Phys., 36 (1980), pp. 366–410.
- [19] S. T. O’DONNELL AND V. ROKHLIN, *A fast algorithm for the numerical evaluation of conformal mappings*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 475–487.
- [20] N. PAPAMICHAEL AND N. S. STYLIANOPOULOS, *A domain decomposition method for conformal mapping onto a rectangle*, Constr. Approx., 7 (1991), pp. 349–379.
- [21] N. PAPAMICHAEL AND N. S. STYLIANOPOULOS, *A domain decomposition method for approximating the conformal modules of long quadrilaterals*, Numer. Math., 62 (1992), pp. 213–234.
- [22] N. PAPAMICHAEL, M. K. WARBY, AND D. M. HOUGH, *The treatment of corner and pole-type singularities in numerical conformal mapping techniques*, J. Comput. Appl. Math., 14 (1986), pp. 163–191.
- [23] R. PIESSENS, E. DEONCKER KAPENGA, C. UBERHUBER, AND D. KAHANER, *QUADPACK: A Subroutine Package for Automatic Integration*, Springer-Verlag, Berlin, New York, 1983.
- [24] L. REICHEL, *A fast method for solving certain integral equations of the first kind with application to conformal mapping*, J. Comput. Appl. Math., 14 (1986), pp. 125–142.
- [25] G. T. SYMM, *An integral equation method in conformal mapping*, Numer. Math., 9 (1966), pp. 250–258.
- [26] T. THEODORSEN, *Theory of Wing Sections of Arbitrary Shape*, NACA Tech. Report 411, NASA, Hanover, MD, 1931.
- [27] A. E. TREFETHEN, V. S. MENON, C.-C. CHANG, G. J. CZAJKOWSKI, C. MYERS, AND L. N. TREFETHEN, *MultiMATLAB: MATLAB on Multiple Processors*, Tech. Report CTC96TR239, Cornell Theory Center, Cornell University, Ithaca, NY, 1996.
- [28] S. WARSCHAWSKI, *On the Solution of the Lichtenstein–Gersgorin Integral Equation in Conformal Mapping*, National Bureau of Standards Appl. Math. Ser. 42, U.S. Government Printing Office, Washington, DC, 1955.
- [29] R. WEGMANN, *An iterative method for conformal mapping*, J. Comput. Appl. Math., 14 (1986), pp. 7–18.