966

A STABLE AND FAST ALGORITHM FOR UPDATING THE SINGULAR VALUE DECOMPOSITION

MING GU* AND STANLEY C. EISENSTAT†

Abstract. Let $A \in \mathbf{R}^{m\times n}$ be a matrix with known singular values and singular vectors, and let A' be the matrix obtained by appending a row to A. We present stable and fast algorithms for computing the singular values and the singular vectors of A' in $O\left((m+n)\min(m,n)\log_2^2\epsilon\right)$ floating point operations, where ϵ is the machine precision. Previous algorithms can be unstable and compute the singular values and the singular vectors of A' in $O\left((m+n)\min^2(m,n)\right)$ floating point operations.

1. Introduction. The *singular value decomposition* (**SVD**) of a matrix $A \in \mathbf{R}^{m\times n}$ is

$$A = U\Omega V^T\,, \tag{1.1}$$

where $U \in \mathbf{R}^{m\times m}$ and $V \in \mathbf{R}^{n\times n}$ are orthonormal; and $\Omega \in \mathbf{R}^{m\times n}$ is zero except on the main diagonal, which has non-negative entries in decreasing order. The columns of U and V are the *left singular vectors* and the *right singular vectors* of A, respectively; the diagonal entries of Ω are the *singular values* of A.

In many least squares and signal processing applications (see [1, 14, 19] and the references therein), we repeatedly update A by appending a row or a column, or downdate A by deleting a row or a column. After each update or downdate, we compute the **SVD** of the resulting matrix. In [11] we consider the problem of downdating the **SVD**. In this paper we consider the problem of updating the **SVD**. The problem of updating the **SVD** is also related to the problem of updating the URV and ULV decompositions (see [14, 15, 17, 18]).

Since appending a column to A is tantamount to appending a row to A^T, we only consider the case where a row is appended. Without loss of generality, we further assume that the last row is appended. Thus, we can write

$$A' = \begin{pmatrix} A \\ a^T \end{pmatrix}\,, \tag{1.2}$$

where $A' \in \mathbf{R}^{(m+1)\times n}$ is the updated matrix. We would like to compute the **SVD** of A' by taking advantage of our knowledge of the **SVD** of A.

First consider the case $m \geq n$. We write

$$U = (U_1\ \ U_2) \quad\text{and}\quad \Omega = \begin{pmatrix} D \\ 0 \end{pmatrix}\,,$$

where $U_1 \in \mathbf{R}^{m\times n}$, $U_2 \in \mathbf{R}^{m\times(m-n)}$ and $D \in \mathbf{R}^{n\times n}$. Equation (1.2) can be rewritten as

$$A' = \begin{pmatrix} U_1 & 0 & U_2 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} D \\ z^T \\ 0 \end{pmatrix} V^T\,,$$

*Department of Mathematics, University of California, Los Angeles. Email address: mgu@math.ucla.edu. This research was supported in part by NSF Career Award CCR-9702866 and by Alfred Sloan Research Fellowship BR-3720.

†Department of Computer Science, Yale University. Email address: sce@cs.yale.edu.

where $z = V^T a \in \mathbf{R}^n$. Let $(W \; w) \begin{pmatrix} \Omega \\ 0 \end{pmatrix} Q^T$ be the **SVD** of $M = \begin{pmatrix} D \\ z^T \end{pmatrix} \in \mathbf{R}^{(n+1)\times n}$, with $W \in \mathbf{R}^{(n+1)\times n}$, $w \in \mathbf{R}^{n+1}$, and Ω, $Q \in \mathbf{R}^{n\times n}$. Then the **SVD** of A' is

$$A' = (U_1' \; U_2') \begin{pmatrix} \Omega \\ 0 \end{pmatrix} (VQ)^T , \tag{1.3}$$

where $(U_1' \; U_2') \in \mathbf{R}^{(m+1)\times(m+1)}$ is orthonormal with

$$U_1' = \begin{pmatrix} U_1 & 0 \\ 0 & 1 \end{pmatrix} W \in \mathbf{R}^{(m+1)\times n} \quad \text{and} \quad U_2' = \left(w' \; \begin{pmatrix} U_2 \\ 0 \end{pmatrix} \right) \in \mathbf{R}^{(m+1)\times(m+1-n)} , \tag{1.4}$$

where $w' = \begin{pmatrix} U_1 & 0 \\ 0 & 1 \end{pmatrix} w$.

Since M is not related to U, the singular values and right singular vectors of A' can be updated without it. When we need to update U_1, we compute U_1'; and when we need to update $U = (U_1 \; U_2)$, we further compute w'.

Since error is associated with computation, a *numerical* **SVD** of A' or M is usually defined as a decomposition of the form

$$A' = (\hat{U}_1' \; \hat{U}_2') \begin{pmatrix} \hat{\Omega} \\ 0 \end{pmatrix} \hat{Y}^T + O(\epsilon \, \|A'\|_2) \quad \text{or} \quad M = \left(\hat{W} \; \hat{w}\right) \begin{pmatrix} \hat{\Omega} \\ 0 \end{pmatrix} \hat{Q}^T + O(\epsilon \, \|M\|_2) \ , \tag{1.5}$$

where ϵ is the machine precision; $\hat{\Omega}$ is diagonal with non-negative entries in decreasing order; and $(\hat{U}_1' \;\; \hat{U}_2')$ and $\hat{Y}$ or $\hat{W}$ and $(\hat{W} \;\; \hat{w})$ are *numerically orthonormal.* An algorithm that produces such a decomposition is said to be *backward stable* [16].

While the singular values of A' and M are always well-conditioned with respect to a perturbation, the singular vectors can be extremely sensitive to such perturbations [5]. That is, $\hat{\Omega}$ can be guaranteed to be close to Ω, but $\hat{U}_1'$, $(\hat{W} \; \hat{w})$ and $\hat{Q}$ can be very different from U_1', $(W \; w)$ and Q, respectively. Thus one is usually content with backward stable algorithms for computing the eigendecompositions of A' and M.

We compute a numerical **SVD** of M of the form (1.5) by using the techniques in [8, 9, 10] (see Section 2). We compute the right singular vector matrix as $V\hat{Q}$. If the left singular vector matrix is updated, we compute it according to (1.4) with $(W \; w)$ replaced by $(\hat{W} \; \hat{w})$.

It takes $O(n^2)$ floating point operations to compute a numerical **SVD** of M (see Section 2). It takes $O(mn)$ floating point operations to compute $\hat{U}_2'$. Since both $\hat{Q}$ and $(\hat{W} \; \hat{w})$ are dense matrices, it takes ostensibly about $2n^3$ and $2mn^2$ floating point operations to compute $\hat{Y}$ and $\hat{U}_1'$, respectively. However, we show that by using the fast multipole method of Carrier, Greengard and Rokhlin [3, 7], the right and left singular vector matrices of A' can be stably computed in $O(n^2 \log_2^2 \epsilon)$ and $O(mn \log_2^2 \epsilon)$ floating point operations, respectively (see Sections 3 and 4).

The case $m < n$ is similar. We write

$$V = (V_1 \; V_2) \quad \text{and} \quad \Omega = (D \;\; 0) \, ,$$

where $V_1 \in \mathbf{R}^{n\times m}$, $V_2 \in \mathbf{R}^{n\times(n-m)}$ and $D \in \mathbf{R}^{m\times m}$. Equation (1.2) can be rewritten as

$$\begin{aligned} A' &= \begin{pmatrix} U & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} D & 0 \\ z_1^T & z_2^T \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \\ &= \begin{pmatrix} U & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} D & 0 & 0 \\ z_1^T & \zeta & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ (V_2 H)^T \end{pmatrix} , \end{aligned}$$

where $z_1 = V_1^T a \in \mathbf{R}^m$ and $z_2 = V_2^T a \in \mathbf{R}^{n-m}$; and $H^T z_2 = (\zeta, 0, \ldots, 0)^T$ is an orthonormal Householder transform. Let $W \Omega Q^T$ be the **SVD** of

$$\left(\begin{array}{cc} D & 0 \\ z_1^T & \zeta \end{array} \right) \in \mathbf{R}^{(m+1)\times(m+1)} . \tag{1.6}$$

Then the **SVD** of A' is

$$A' = U' \, (\Omega \;\; 0) \left(\begin{array}{c} V_1'^T \\ V_2'^T \end{array} \right) , \tag{1.7}$$

where

$$U' = \left(\begin{array}{cc} U & 0 \\ 0 & 1 \end{array} \right) W \quad , \quad V_1' = (V_1 \;\; v) Q \quad \text{and} \quad (v \;\; V_2') = V_2 H ,$$

with $v \in \mathbf{R}^n$ being the first column of $V_2 H$. Since M_1 is not related to U we can update the singular values and right singular vectors of A' without it.

We compute a numerical **SVD** of M_1 (cf. (1.5)) by using the techniques in [8, 9, 10] (see Section 2). We stably compute the right and left singular vector matrices of A' by using the fast multipole method in $O(mn \log_2^2 \epsilon)$ and $O(m^2 \log_2^2 \epsilon)$ floating point operations, respectively. Similar to the previous case, the singular values of A' and M_1 are always well-conditioned with respect to a perturbation, the singular vectors can be extremely sensitive to such perturbations [5].

In both cases, the problem of updating the **SVD** has been considered by Bunch and Nielsen [1], using results from [2, 4]. Their scheme for finding the **SVD** of M and M_1 can be unstable [1, 2]. And their algorithm takes about $2n \min^2(m,n)$ and $2m \min^2(m,n)$ floating point operations to update the right and the left singular vector matrices, respectively. The lack of a fast algorithm for updating the **SVD** is one of the reasons for the recent development of URV and ULV decomposition algorithms to *approximately* update the **SVD** [14, 15, 17, 18].

For the purpose of computing the **SVD** of M and M_1, we take the usual model of arithmetic

$$\mathbf{fl}(\alpha \circ \beta) = (\alpha \circ \beta)\,(1 + \xi) ,$$

where α and β are floating point numbers; $\circ$ is one of $+, -, \times,$ and $\div$; $\mathbf{fl}(\alpha \circ \beta)$ is the floating point result of the operation $\circ$; and $|\xi| \le \epsilon$. We also require that[1]

$$\mathbf{fl}(\sqrt{\alpha}) = \sqrt{\alpha}\,(1 + \xi)$$

for any positive floating point number α. For simplicity, we ignore the possibility of overflow and underflow.

2. Computing the SVD of M and M_1. In this section we present a scheme for finding the **SVD** of the matrix

$$M = \left(\begin{array}{c} D \\ z^T \end{array} \right) = \left(\begin{array}{ccc} d_1 & & \\ & \ddots & \\ & & d_n \\ \zeta_1 & \cdots & \zeta_n \end{array} \right) , \tag{2.1}$$

[1] This model excludes machines that do not have a guard digit, such as CRAYs and CDC Cybers. It also excludes machines that do not support binary floating point operations. Our algorithms can be modified for such machines.

where $D = \text{diag}(d_1, d_2, \ldots, d_n)$ with $0 \leq d_1 \leq d_2 \leq \ldots \leq d_n$; and $z = (z_1, z_2, \ldots, z_n)^T$. We further assume that

$$(2.2) \qquad d_1 \geq \tau \|M\|_2 \quad , \quad d_{i+1} - d_i \geq \tau \|M\|_2 \quad \text{and} \quad |z_i| \geq \tau \|M\|_2 \ ,$$

where τ is a small multiple of ϵ to be specified in Section 2.2. Any matrix of the form (2.1) can be reduced to one that satisfies these conditions by the deflation procedures described in [8, 9] and a simple permutation. The scheme for finding the **SVD** of M_1 (see (1.6)) appears in [8, 9]. The techniques for both problems are very similar.

The following lemma characterizes the singular values and singular vectors of M.

LEMMA 2.1 (Jessup and Sorensen [12]). *Let* $W \begin{pmatrix} \Omega \\ 0 \end{pmatrix} Q^T$ *be the* **SVD** *of* M *with*

$$W = (w_1, \ldots, w_n, w_{n+1}) \quad , \quad \Omega = \text{diag}(\omega_1, \ldots, \omega_n) \quad \textit{and} \quad Q = (q_1, \ldots, q_n) \ ,$$

where $0 < \omega_1 < \ldots < \omega_n$. *Then*

$$M^T M = D^2 + zz^T = Q\Omega^2 Q^T \ .$$

is the eigendecomposition of $M^T M$. *The singular values* $\{\omega_j\}_{j=1}^n$ *satisfy the* interlacing property

$$0 < d_1 < \omega_1 < d_2 < \ldots < d_n < \omega_n < d_n + \|z\|_2 \quad ,$$

and the secular equation

$$\mathcal{F}(\omega) = 1 + \sum_{i=1}^{n} \frac{z_i^2}{d_i^2 - \omega^2} = 0 \ .$$

The singular vectors satisfy

$$(2.3) \qquad w_j = \left(\frac{d_1 \, z_1}{d_1^2 - \omega_j^2}, \ldots, \frac{d_n \, z_n}{d_n^2 - \omega_j^2}, -1 \right)^T \Bigg/ \sqrt{1 + \sum_{i=1}^{n} \frac{d_i^2 \, z_i^2}{\left(d_i^2 - \omega_j^2\right)^2}} \quad ,$$

$$(2.4) \qquad w_{n+1} = \left(\frac{z_1}{d_1}, \ldots, \frac{z_n}{d_n}, -1 \right)^T \Bigg/ \sqrt{1 + \sum_{i=1}^{n} \frac{z_i^2}{d_i^2}} \quad ,$$

$$(2.5) \qquad q_j = \left(\frac{z_1}{d_1^2 - \omega_j^2}, \ldots, \frac{z_n}{d_n^2 - \omega_j^2} \right)^T \Bigg/ \sqrt{\sum_{i=1}^{n} \frac{z_i^2}{\left(d_i^2 - \omega_j^2\right)^2}} \quad ,$$

where $j = 1, \ldots, n$.

On the other hand, given D and all the singular values, we can construct a matrix with the same structure as (2.1).

LEMMA 2.2. *Given a diagonal matrix* $D = \text{diag}(d_1, d_2, \ldots, d_n)$ *and a set of numbers* $\{\hat{\omega}_j\}_{j=1}^n$ *satisfying the interlacing property*

$$(2.6) \qquad 0 < d_1 < \hat{\omega}_1 < d_2 < \ldots < d_n < \hat{\omega}_n \ ,$$

there exists a matrix

$$\hat{M} = \begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_n \\ \hat{z}_1 & \dots & \hat{z}_n \end{pmatrix}$$

whose singular values are $\{\hat{\omega}_j\}_{j=1}^n$. *The vector* $\hat{z} = (\hat{z}_1, \hat{z}_2, \dots, \hat{z}_n)^T$ *is determined by*

$$|\hat{z}_i| = \sqrt{(\hat{\omega}_n^2 - d_i^2) \prod_{j=1}^{i-1} \frac{(\hat{\omega}_j^2 - d_i^2)}{(d_j^2 - d_i^2)} \prod_{j=i}^{n-1} \frac{(\hat{\omega}_j^2 - d_i^2)}{(d_{j+1}^2 - d_i^2)}}\,, \tag{2.7}$$

where the sign of $\hat{z}_i$ *can be chosen arbitrarily.*

2.1. Computing the Singular Vectors of M. For each *exact* singular value ω_j, equations (2.3) and (2.5) give the corresponding *exact* singular vectors. Observe that if ω_j was given, then we could compute each difference $d_i^2 - \omega_j^2$ to high relative accuracy as $(d_i - \omega_j)(d_i + \omega_j)$. We could also compute each product and each ratio to high relative accuracy, and thus compute w_j and q_j to component-wise high relative accuracy.

In practice we can only hope to compute an approximation $\hat{\omega}_j$ to ω_j. But problems can arise if we approximate w_j and q_j by

$$\hat{w}_j = \left(\frac{d_1 z_1}{d_2^2 - \hat{\omega}_j^2}, \dots, \frac{d_n z_n}{d_n^2 - \hat{\omega}_j^2} \right)^T \Bigg/ \sqrt{1 + \sum_{i=1}^n \frac{(d_i z_i)^2}{\left(d_i^2 - \hat{\omega}_j^2\right)^2}}$$

and

$$\hat{q}_j = \left(\frac{z_1}{d_1^2 - \hat{\omega}_j^2}, \dots, \frac{z_n}{d_n^2 - \hat{\omega}_j^2} \right)^T \Bigg/ \sqrt{\sum_{i=1}^n \frac{z_i^2}{\left(d_i^2 - \hat{\omega}_j^2\right)^2}}$$

(i.e., replace ω_j by $\hat{\omega}_j$ in equations (2.3) and (2.5) as in [1, 12]). For even if $\hat{\omega}_j$ is close to ω_j, the approximate ratios $z_i/(d_i^2 - \hat{\omega}_j^2)$ and $d_i z_i/(d_i^2 - \hat{\omega}_j^2)$ can still be very different from the exact ratios $z_i/(d_i^2 - \omega_j^2)$ and $d_i z_i/(d_i^2 - \omega_j^2)$, resulting in singular vectors very different from w_j and q_j. And when all the approximate singular values $\{\hat{\omega}_j\}_{j=1}^n$ are computed and all the corresponding singular vectors are approximated in this manner, the resulting singular vector matrices may not be orthonormal.

Lemma 2.2 allows us to overcome this problem. After we have computed all the approximate singular values $\{\hat{\omega}_j\}_{j=1}^k$ of M, we find a *new* matrix $\hat{M}$ whose *exact* singular values are $\{\hat{\omega}_j\}_{j=1}^k$, and then compute the singular vectors of $\hat{M}$ using Lemma 2.1. Note that each difference

$$\hat{\omega}_j^2 - d_i^2 = (\hat{\omega}_j - d_i)(\hat{\omega}_j + d_i) \quad \text{and} \quad d_j^2 - d_i^2 = (d_j - d_i)(d_j + d_i)$$

in (2.7) can be computed to high relative accuracy. Each ratio and each product can also be computed to high relative accuracy. Thus $|\hat{z}_i|$ can be computed to high relative accuracy. We choose the sign of $\hat{z}_i$ to be the sign of z_i. Substituting the *exact* singular values $\{\hat{\omega}_j\}_{j=1}^n$ and the computed $\hat{z}$ into equations (2.3) and (2.5), each singular vector of $\hat{M}$ can again be computed to component-wise high relative accuracy. Consequently, after all the singular vectors are computed, the singular vector matrices of $\hat{M}$ will be numerically orthonormal.

To ensure the existence of $\hat{M}$, we need $\{\hat{\omega}_j\}_{j=1}^n$ to satisfy the interlacing property (2.6). But since the exact singular values of M satisfy the same interlacing property (see Lemma 2.1), this is only an accuracy requirement on the computed singular values, and is not an additional restriction on M. We can use the **SVD** of $\hat{M}$ as an approximation to the **SVD** of M. Since

$$M = \hat{M} + \begin{pmatrix} 0 & & \\ & \ddots & \\ & & 0 \\ z_1 - \hat{z}_1 & \dots & z_n - \hat{z}_n \end{pmatrix},$$

we have

$$|\hat{\omega}_j - \omega_j| \le \|M - \hat{M}\|_2 \le \|z - \hat{z}\|_2 .$$

Such a substitution is backward stable (see (1.5)) as long as $\hat{z}$ is close to z (cf. [8, 9, 10]).

2.2. Computing the Singular Values of M. In order to guarantee that $\hat{z}$ is close to z, we must ensure that the approximations $\{\hat{\omega}_j\}_{j=1}^n$ to the singular values are sufficiently accurate. The key is the stopping criterion for the root-finder, which requires a slight reformulation of the secular equation (cf. [1, 8, 9, 10]).

Consider the singular value $\omega_j \in (d_j, d_{j+1})$, where $1 \le j \le n-1$; the case $j = n$ is considered later. ω_j is a root of the secular equation

$$\mathcal{F}(\omega) = 1 + \sum_{i=1}^{n} \frac{z_i^2}{d_i^2 - \omega^2} = 0 .$$

We first assume that[2] $\omega_j \in (d_j, \frac{d_j + d_{j+1}}{2})$. Let $\delta_i = d_i - d_j$ and let

$$\psi(\mu) \equiv \sum_{i=1}^{i} \frac{z_i^2}{(\delta_i - \mu)(d_i + d_j + \mu)} \quad \text{and} \quad \phi(\mu) \equiv \sum_{i=j+1}^{n} \frac{z_i^2}{(\delta_i - \mu)(d_i + d_j + \mu)} .$$

Setting $\omega = d_j + \mu$, we have

$$\mathcal{F}(\mu + d_j) = 1 + \psi(\mu) + \phi(\mu) \equiv \mathcal{G}(\mu) .$$

We seek the root $\mu_j = \omega_j - d_j \in (0, \delta_{j+1}/2)$ of $\mathcal{G}(\mu) = 0$.

An important property of $\mathcal{G}(\mu)$ is that we can compute each difference $\delta_i - \mu$ to high relative accuracy for any $\mu \in (0, \delta_{j+1}/2)$. Indeed, since $\delta_j = 0$, we have $\mathbf{fl}(\delta_j - \mu) = -\mathbf{fl}(\mu)$; since $\mathbf{fl}(\delta_{j+1}) = \mathbf{fl}(d_{j+1} - d_j)$ and $0 < \mu < (d_{j+1} - d_j)/2$, we can compute $\mathbf{fl}(\delta_{j+1} - \mu)$ as $\mathbf{fl}(\mathbf{fl}(d_{j+1} - d_j) - \mathbf{fl}(\mu))$; and in a similar fashion, we can compute $\delta_i - \mu$ to high relative accuracy for any $i \ne j, j+1$.

Because we can also compute $d_i + d_j + \mu$ (a sum of positive terms) to high relative accuracy, we can compute each ratio $z_i^2/((\delta_i - \mu)(d_i + d_j + \mu))$ in $\mathcal{G}(\mu)$ to high relative

[2] This can easily be checked by computing $\mathcal{F}(\frac{d_j + d_{j+1}}{2})$. If $\mathcal{F}(\frac{d_j + d_{j+1}}{2}) > 0$, then $\omega_j \in (d_j, \frac{d_j + d_{j+1}}{2})$, otherwise $\omega_j \in [\frac{d_j + d_{j+1}}{2}, d_{j+1})$.

accuracy for any $\mu \in (0, \delta_{j+1}/2)$. And, since both $\psi(\mu)$ and $\phi(\mu)$ are sums of terms of the same sign, we can bound the error in computing $\mathcal{G}(\mu)$ by

$$\eta n(1 + |\psi(\mu)| + |\phi(\mu)|) \ ,$$

where η is a small multiple of ϵ that is independent of n and μ.

We now assume that $\omega_j \in [\frac{d_j+d_{j+1}}{2}, d_{j+1})$. Let $\delta_i = d_i - d_{j+1}$ and let

$$\psi(\mu) \equiv \sum_{i=1}^{j} \frac{z_i^2}{(\delta_i - \mu)(d_i + d_{j+1} + \mu)} \quad \text{and} \quad \phi(\mu) \equiv \sum_{i=j+1}^{n} \frac{z_i^2}{(\delta_i - \mu)(d_i + d_{j+1} + \mu)} \ .$$

Setting $\omega = d_{j+1} + \mu$, we seek the root $\mu_j = \omega_j - d_{j+1} \in [\delta_j/2, 0)$ of the equation

$$\mathcal{G}(\mu) \equiv \mathcal{F}(\mu + d_{j+1}) = 1 + \psi(\mu) + \phi(\mu) = 0 \ .$$

For any $\mu \in [\delta_j/2, 0)$, we can compute each difference $\delta_i - \mu$ to high relative accuracy. Since $|\mu| \leq |\delta_j|/2 \leq d_{j+1}/2$, we can compute each sum $d_i + d_{j+1} + \mu$ to high relative accuracy as $d_i + (d_{j+1} + \mu)$. Thus we can again compute each ratio $z_i^2/((\delta_i - \mu)(d_i + d_{j+1} + \mu))$ to high relative accuracy and bound the error in computing $\mathcal{G}(\mu)$ as before.

Finally we consider the case $j = n$. Let $\delta_i = d_i - d_n$ and let

$$\psi(\mu) \equiv \sum_{i=1}^{n} \frac{z_i^2}{(\delta_i - \mu)(d_i + d_n + \mu)} \quad \text{and} \quad \phi(\mu) \equiv 0 \ .$$

Setting $\omega = d_n + \mu$, we seek the root $\mu_n = \omega_n - d_n \in (0, \|z\|_2)$ of the equation

$$\mathcal{G}(\mu) \equiv \mathcal{F}(\mu + d_n) = 1 + \psi(\mu) + \phi(\mu) = 0 \ .$$

Again, for any $\mu \in (0, \|z\|_2)$, we can compute each ratio $z_i^2/((\delta_i - \mu)(d_i + d_n + \mu))$ to high relative accuracy, and we can bound the error in computing $\mathcal{G}(\mu)$ as before.

In practice the root-finder cannot make any progress at a point μ where it is impossible to determine the sign of $\mathcal{G}(\mu)$ numerically. Thus we propose the stopping criterion

$$|\mathcal{G}(\mu)| \leq \eta n \, (1 + |\psi(\mu)| + |\phi(\mu)|) \ , \tag{2.8}$$

where, as before, $\eta n(1 + |\psi(\mu)| + |\phi(\mu)|)$ is an upper bound on the round-off error in computing $\mathcal{G}(\mu)$. Note that for each j, there is at least one floating point number that satisfies this stopping criterion numerically, namely $\mathbf{fl}(\mu_j)$.

We have not specified the scheme for finding the root of $\mathcal{G}(\mu)$. We can use the bisection method or the rational interpolation strategies in [1, 6, 13]. What is most important is the stopping criterion and the fact that, with the reformulation of the secular equation given above, we can find a μ that satisfies it.

For each j, we denote the computed value of μ_j by $\hat{\mu}_j$. Thus the computed singular values $\{\hat{\omega}_j\}_{j=1}^n$ satisfy

$$\hat{\omega}_j = d_j + \hat{\mu}_j \quad \text{or} \quad \hat{\omega}_j = d_{j+1} + \hat{\mu}_j \tag{2.9}$$

and

$$0 < d_1 < \hat{\omega}_1 < d_2 < \ldots < d_n < \hat{\omega}_n \ . \tag{2.10}$$

An argument similar to that of this section shows that we can compute $\hat{\omega}_j^2 - d_i^2$ to high relative accuracy. Thus we can compute $\hat{z}$ and the singular vectors of $\hat{M}$ to component-wise high relative accuracy. Lemma 2.3 shows that this approach is stable.

LEMMA 2.3 (Gu and Eisenstat [9]). *Assume that the stopping criterion (2.8) is satisfied by every computed μ_j. Also assume that $\tau \geq 2\eta n^2$ in (2.2). Then*

$$|\hat{z}_i - z_i| \leq 4\eta n^2 \|z\|_2 ,$$

for $i = 1, \dots, n$.

3. Acceleration by the Fast Multipole Method.

Suppose that we want to evaluate the complex function

$$\Phi(\omega) = \sum_{i=1}^{n} x_i \varphi(d_i - \omega) \tag{3.1}$$

at n points in the complex plane, where $\{x_i\}_{i=1}^n$ and $\{d_i\}_{i=1}^n$ are constants and $\varphi(\omega)$ is one of $\log(\omega)$, $1/\omega$ and $1/\omega^2$. The direct computation takes $O(n^2)$ time. But the *fast multipole method* (**FMM**) proposed by Carrier, Greengard and Rokhlin [3, 7] takes only $O(n \log_2^2 \epsilon)$ time to compute $\Phi(\omega)$ at these points. In this section we describe a modified **FMM** to accelerate our algorithms for updating the **SVD**.

With the singular vector matrices of M or M_1, we compute the singular vector matrices of $A' \in \mathbf{R}^{(m+1)\times n}$ (see Section 1). In this section we only consider the problem of computing the right singular vector matrix of A' for the case $m \geq n$. The techniques for other singular vector matrix computations are basically the same.

From Sections 1 and 2 (see equations (1.3), (1.5) and Section 2.1) we have

$$A' = (\hat{U}_1' \;\; \hat{U}_2') \begin{pmatrix} \hat{\Omega} \\ 0 \end{pmatrix} (V\hat{Q})^T + O(\epsilon \, \|A'\|_2) ,$$

where $V \in \mathbf{R}^{n\times n}$ is orthonormal and $(\hat{W} \; \hat{w}) \begin{pmatrix} \hat{\Omega} \\ 0 \end{pmatrix} \hat{Q}^T$ is the **SVD** of $\hat{M} = \begin{pmatrix} D \\ \hat{z}^T \end{pmatrix} \in \mathbf{R}^{(n+1)\times n}$. The matrix-matrix product $V\hat{Q}$ is an approximate right singular vector matrix of A' (see Section 2). The singular values of $\hat{M}$ are given as (see (2.9) and (2.10))

$$\hat{\omega}_j = d_j + \hat{\mu}_j \quad \text{or} \quad \hat{\omega}_j = d_{j+1} + \hat{\mu}_j \tag{3.2}$$

with $\{d_j\}_{j=1}^n$ satisfying (2.2).

According to Lemma 2.1, we have $\hat{Q} = (\hat{q}_1, \dots, \hat{q}_n)$ with

$$\hat{q}_j = \left(\frac{\hat{z}_1}{d_1^2 - \hat{\omega}_j^2}, \dots, \frac{\hat{z}_n}{d_n^2 - \hat{\omega}_j^2} \right)^T \Bigg/ \sqrt{\sum_{i=1}^{n} \frac{\hat{z}_i^2}{\left(d_i^2 - \hat{\omega}_j^2\right)^2}} \; .$$

Let $v^T = (v_1, \dots, v_n)^T$ be a row of V. Then the corresponding row of $V\hat{Q}$ is $v^T\hat{Q} = (v^T\hat{q}_1, \dots, v^T\hat{q}_n)$ with $v^T\hat{q}_j = \Phi_1(\hat{\omega}_j) \Big/ \sqrt{\Phi_2(\hat{\omega}_j)}$, where

$$\Phi_1(\omega) = \sum_{i=1}^{n} \frac{v_i \hat{z}_i}{d_i^2 - \omega^2} \quad \text{and} \quad \Phi_2(\omega) = \sum_{i=1}^{n} \frac{\hat{z}_i^2}{\left(d_i^2 - \omega^2\right)^2} \; . \tag{3.3}$$

Thus we can compute $v^T \hat{Q}$ by evaluating $\Phi_1(\omega)$ and $\Phi_2(\omega)$ at n points $\{\hat{\omega}_j\}_{j=1}^n$. Note that for each different row of V, there is a different function $\Phi_1(\omega)$, whereas $\Phi_2(\omega)$ remains the same. Thus the major cost in computing $V\hat{Q}$ is for each v to evaluate $\Phi_1(\omega)$ at the same points $\{\hat{\omega}_j\}_{i=1}^n$. The direct computation takes $O(n^3)$ floating point operations.

Note that $\Phi_1(\omega)$ is similar to the form in (3.1). In this section, we present a modified **FMM** for computing

$$\Phi(\omega) = \sum_{i=1}^{n} \frac{x_i}{d_i^2 - \omega^2} \tag{3.4}$$

at n points $\{\omega_j\}_{j=1}^n$ satisfying the interlacing property

$$0 < d_1 < \omega_1 < d_2 < \ldots < d_n < \omega_n, \tag{3.5}$$

where the singularities $\{d_i\}_{i=1}^n$ of $\Phi(\omega)$ satisfy (??). The modified **FMM** takes advantage of the fact that all the computations are real. We then show how to use the modified **FMM** to compute $\Phi(\omega)$ at $\{\hat{\omega}_j\}_{j=1}^n$. Finally we use the modified **FMM** to stably compute the matrix-matrix product $V\hat{Q}$ in $O(n^2 \log_2^2 \epsilon)$ floating point operations. Most of the results in Section 3 parallel those of [3, 7].

3.1. Chebyshev Interpolation. Our modified FMM is based on polynomial interpolation. Define the Chebyshev polynomials

$$T_0(x) = 1, \quad T_1(x) = x, \quad \text{and} \quad T_{k+1}(x) = 2\,x T_k(x) - T_{k-1}(x), \quad k = 1, 2, \cdots. \tag{3.6}$$

It is well-known that they satisfy $|T_k(x)| \leq 1$ for all $x \in [-1, 1]$ and for all k. For $k \geq 1, T_k(x)$ has exactly k roots in the interval $[-1, 1]$ given by

$$\theta_k^j = \cos\left(\frac{2j-1}{2k}\pi\right), \quad j = 1, 2, \cdots, k. \tag{3.7}$$

These roots will be referred to as k-th order Chebyshev nodes throughout the rest of this paper. The Chebyshev polynomials can also be written as

$$T_k(x) = \cos\left(k \cos^{-1} x\right) = \frac{1}{2}\left(\left(x + \sqrt{x^2 - 1}\right)^n + \left(x - \sqrt{x^2 - 1}\right)^n\right).$$

For $|x| > 1$, we have

$$|T_k(x)| = T_k(|x|) \approx \frac{1}{2}\left(|x| + \sqrt{|x|^2 - 1}\right)^n. \tag{3.8}$$

The function $|T_k(x)|$ monotonically increases to ∞ for modest values of k (see Table).

Lemma 3.1 below is quite elementary. However, it is the main analytic tool in our modified FMM. For its introduction we need the following polynomials

$$P_k^\alpha(x) = \frac{T_{k+1}(\alpha) - T_{k+1}(x)}{(\alpha - x)\; T_{k+1}(\alpha)},$$

where k is a non-negative integer and α is any real number. We note that while $P_k^\alpha(x)$ might appear to be a rational function, it is indeed a polynomial since the polynomial in the numerator has α as a root.

LEMMA 3.1. *Let k be any positive integer and let $|\alpha| > 1$. Then*

$$\frac{1}{\alpha - x} - P_k^{\alpha}(x) = \frac{T_{k+1}(x)}{T_{k+1}(\alpha)\ (\alpha - x)}, \tag{3.9-a}$$

$$\left|\frac{1}{\alpha - x} - P_k^{\alpha}(x)\right| \leq \frac{1}{|T_{k+1}(\alpha)|\ (|\alpha| - 1)}, \tag{3.9-b}$$

REMARK 3.1. *Since the numerator on the right-hand-side of equation (3.9-a) is $T_{k+1}(x)$, it follows from (3.7) that it must vanish at the $(k+1)$-st order Chebyshev nodes. Consequently, The polynomial $P_k^{\alpha}(x)$ is precisely the polynomial obtained by interpolating the function $1/(\alpha - x)$ at these Chebyshev nodes.*

REMARK 3.2. *For any given k, $P_k^{\alpha}(x)$ is the polynomial that approximates $1/(\alpha - x)$ with the smallest relative error, i.e.,*

$$\min_{\deg(p) \leq k} \max_{x \in [-1,1]} \left| \left(\frac{1}{\alpha - x} - p(x) \right) (\alpha - x) \right| = \min_{\deg(p) \leq k} \max_{x \in [-1,1]} |(1 - (\alpha - x)\ p(x))| = \frac{1}{|T_{k+1}(\alpha)|}.$$

Lemma 3.2 below is similar to Lemma 3.1.

LEMMA 3.2. *Let k be any positive integer and let $|\beta| < 1$. Then*

$$\frac{1}{1 - \beta x} - \frac{1}{\beta} P_k^{1/\beta}(x) = \frac{T_{k+1}(x)}{T_{k+1}(1/\beta)\ (1 - \beta x)},$$

$$\left|\frac{1}{1 - \beta x} - P_k^{1/\beta}(x)\right| \leq \frac{1}{|T_{k+1}(1/\beta)|\ (1 - |\beta|)}.$$

Another advantage of Chebyshev interpolation is that it is also numerically stable. Let $P(x)$ be the unique polynomial of degree at most k that interpolates a function $f(x)$ at the Chebyshev nodes:

$$P\left(r_k^i\right) = f_i, \quad i = 0, 1, \cdots, k.$$

$P(x)$ can be written as

$$P(x) = \sum_{j=0}^{k} \frac{\prod_{i \neq j}(x - x_i)}{\prod_{i \neq j}(x_j - x_i)} = \sum_{j=0}^{k} \frac{T_k(x)}{(x - x_j)\ T_k'(x_i)}.$$

Let $P(x)$ and $\widehat{P}(x)$ be polynomials of degree at most k that interpolate
posesses the optimality properties in Remark 3.2, it

3.2. Basic Ideas of FMM. In this section we first consider methods for fast evaluation of $\Phi(\omega)$ of (3.4) for two special distributions of $\{d_i\}_{i=1}^n$ and $\{\omega_j\}_{j=1}^n$. We then briefly show how to generalize these methods for distributions of $\{d_i\}_{i=1}^n$ and $\{\omega_j\}_{j=1}^n$ satisfying (??) and (3.5). This section is the basis of modified **FMM**.

First assume that we want to evaluate $\Phi(\omega)$ at $\omega = \omega_1, \ldots, \omega_n$ satisfying (see Figure 3.1)

$$|\omega^2 - c| \leq \widehat{r} \quad \text{and} \quad |d_i^2 - c| \geq 3\,\widehat{r} \quad \text{with} \quad \widehat{r} = \frac{r}{2}. \tag{3.10}$$

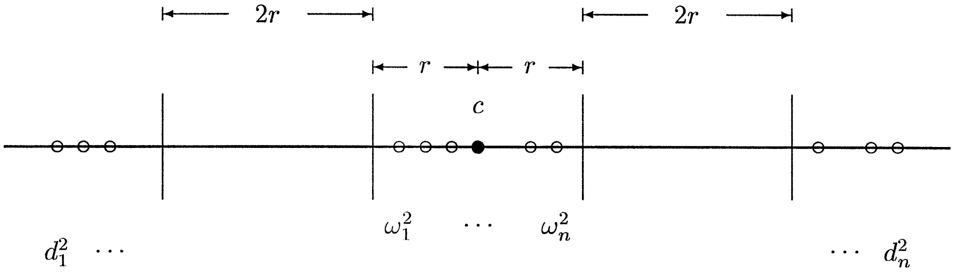

FIG. 3.1. *Local Expansion*

Under these conditions, the function $\Phi(\omega)$ is quite smooth at $\{\omega_j\}_{j=1}^n$, which are clustered around c. We take advantage of this smoothness by approximating $\Phi(\omega)$ via Chebyshev interpolation. Define $\xi = (\omega^2 - c)/r$ and $\alpha_i = (d_i^2 - c)/r$. It follows that $\omega = \sqrt{c + r\,\xi}$. With Lemma 3.1 and equation (3.10), we can rewrite (3.4) as

$$\Phi(\omega) = \sum_{i=1}^{n} \frac{x_i}{r}\,\frac{1}{\alpha_i - \xi} \approx \sum_{i=1}^{n} \frac{x_i}{r}\, P_k^{\alpha_i}(\xi) \overset{\text{def}}{=} \mathcal{L}_k(\xi)\,. \tag{3.11}$$

In our implementation, we compute the polynomial $\mathcal{L}_k(\xi)$ directly without using any of the $P_k^{\alpha_i}(\xi)$. To see how this is done, we observe that each $P_k^{\alpha_i}(\xi)$ is the polynomial resulting from interpolating $1/(\alpha_i - \xi)$ at the $(k+1)$-st order Chebyshev nodes (see Remark 3.1). Consequently, $\mathcal{L}_k(\xi)$ must the polynomial resulting from interpolating $\Phi(\omega)$ at the same nodes. We write $\mathcal{L}_k(\xi)$ in terms of Chebyshev polynomials as follows:

$$\mathcal{L}_k(\xi) = \sum_{j=0}^{k} \gamma_j T_j(\xi). \tag{3.12}$$

It follows from the above argument that

$$\Phi(\sqrt{c + r\,\theta_{k+1}^i}) = \mathcal{L}_k\left(\theta_{k+1}^i\right) = \sum_{j=0}^{k} \gamma_j T_j(\theta_{k+1}^i), \quad i = 1, 2, \cdots, k+1.$$

The above equations can be written in matrix form as

$$\mathcal{F} \begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_{k+1} \end{pmatrix} = \begin{pmatrix} \Phi(\sqrt{c + r\,\theta_{k+1}^1}) \\ \vdots \\ \Phi(\sqrt{c + r\,\theta_{k+1}^{k+1}}) \end{pmatrix}, \tag{3.13}$$

where the coefficients in the matrix have the form

$$\mathcal{F}_{i,j} = T_j(\theta_{k+1}^i) = \cos\left(j \cos^{-1}\left(\cos\left(\frac{2i-1}{2(k+1)}\pi\right)\right)\right) = \cos\left(\frac{j\,(2i-1)}{2(k+1)}\pi\right).$$

Hence $\mathcal{F}$ is a discrete cosine tranform matrix, and the linear system (3.13) can be solved by using the inverse discrete cosine tranform in $O\,(k \log k)$ flops. Since it costs

$O(n)$ flops to compute $\Phi(\omega)$ at each Chebyshev node, the total cost for computing the right hand side in (3.13) is $O\,(n\,k)$ flops.

Now we estimate the error in approximating $\Phi(\omega)$ using $\mathcal{L}_k\,(\xi)$. Since

$$\begin{aligned} \left|d_i^2 - \omega^2\right| &= \left|(d_i^2 - c) - (\omega^2 - c)\right| \le \left|d_i^2 - c\right| + \left|\omega^2 - c\right| \\ &\le \left|d_i^2 - c\right| + r \le 2\left(\left|d_i^2 - c\right| - r\right), \end{aligned}$$

by Lemma 3.1 we have

$$\begin{aligned} \left|\Phi(\omega) - \mathcal{L}_k\,(\xi)\right| &\le \sum_{i=1}^{n} \frac{|x_i|}{r}\,\frac{1}{|T_{k+1}(\alpha_i)|\;(|\alpha_i| - 1)} = \sum_{i=1}^{n} \frac{|x_i|}{|d_i^2 - c| - r}\,\frac{1}{|T_{k+1}(\alpha_i)|} \\ &\le 2\sum_{i=1}^{n} \frac{|x_i|}{|d_i^2 - \omega^2|}\,\frac{1}{|T_{k+1}(\alpha_i)|}. \end{aligned}$$

Furthermore, since $|\alpha_i| \ge 3$ for all i, it follows that

$$\left|\Phi(\omega) - \mathcal{L}_k\,(\xi)\right| \le \frac{2}{T_{k+1}(3)} \sum_{i=1}^{n} \frac{|x_i|}{|d_i^2 - \omega^2|}. \tag{3.14}$$

For a given relative precision ϵ, we choose k so that $2/T_{k+1}(3) \approx \epsilon$. Since

$$T_{k+1}(3) \approx \frac{1}{2}\left(3 + \sqrt{3^2 - 1}\right)^{k+1} \approx 5.8^{k+1}/2,$$

this implies that it is sufficient to choose

$$k \approx \log_{5.8}(4/\epsilon) - 1.$$

In our numerical experiments, we chose $k = 3$ for 3 digits of accuracy, 7 for single precision and 20 for double precision. See Section ?? for more detailed discussion on the choice of k.

Directly computing $\Phi(\omega)$ at $\{\omega_j\}_{j=1}^{n}$ takes about $4n^2$ floating point operations. On the other hand, we can compute $\Phi(\omega)$ at these points using $\mathcal{L}_k(\xi)$.

Computing the coefficients in (3.12) takes $(n\,k)$ flops, and computing $\mathcal{L}_k(\xi)$ at $\xi = (\omega_1^2)/r, \cdots, (\omega_n^2 - c)/r$ takes another $O(n\,k)$ flops. Hence the total cost for approximating $\Phi(\omega)$ at $\{\omega_j\}_{j=1}^{n}$ via $\mathcal{L}_k(\xi)$ is $O(n\,k)$ flops. This is a much smaller cost than $4n^2$ for large enough n.

Next assume that we want to evaluate $\Phi(\omega)$ at $\omega = \omega_1, \ldots, \omega_n$ satisfying (see Figure 3.2)

$$|d_i - c| \le r \quad \text{and} \quad |\omega - c| \ge 3r. \tag{3.15}$$

Under these conditions, $\{\omega_j\}_{j=1}^{n}$ are well-separated from the singularities $\{d_i\}_{i=1}^{n}$ of $\Phi(\omega)$. We take advantage of this well-separatedness by expressing $\Phi(\omega)$ by an expansion centered at c. Let $\beta_i = (c - d_i^2)/(3r)$ and $\zeta = 3r/(c - \omega^2)$. It follows that

$$|\beta_i| \le 1/3 \quad \text{and} \quad \omega = \sqrt{c - 3r/\zeta}.$$

For numerical stability reasons, we introduce the following function

$$\widehat{\Phi}(\zeta) \overset{\text{def}}{=} \sum_{i=1}^{n} \frac{x_i}{1 - \dfrac{c - d_i^2}{c - \omega^2}} = \sum_{i=1}^{n} \frac{x_i}{1 - \beta_i\,\zeta}.$$

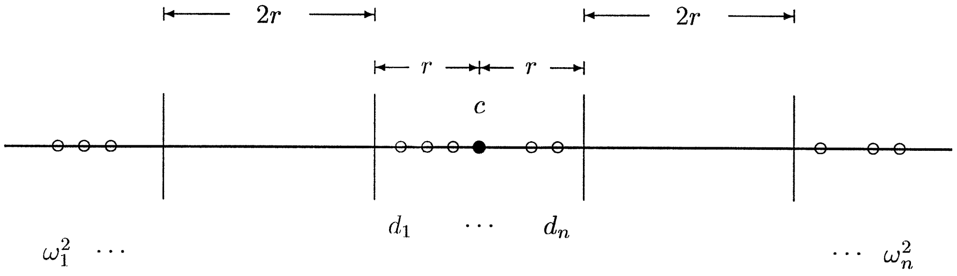

FIG. 3.2. *Multipole Expansion*

The function $\Phi(\omega)$ can be expressed as

$$\Phi(\omega) = \frac{1}{c-\omega^2}\,\widehat{\Phi}(\zeta).$$

Now we apply Lemma 3.2 to $\widehat{\Phi}(\zeta)$ to get

$$\widehat{\Phi}(\zeta) \approx \sum_{i=1}^{n} \frac{x_i}{\beta_i} P_k^{1/\beta_i}(\zeta) \stackrel{\text{def}}{=} \mathcal{M}_k(\zeta)\,. \tag{3.16}$$

As in (3.12), we compute the polynomial $\mathcal{M}_k(\zeta)$ directly without using any of the $P_k^{1/\beta_i}(\zeta)$. We do this by interpolating $\widehat{\Phi}(\zeta)$ at the $(k+1)$-st order Chebyshev nodes. We write $\mathcal{M}_k(\zeta)$ in terms of Chebyshev polynomials as follows:

$$\mathcal{M}_k(\zeta) = \sum_{j=0}^{k} \delta_j T_j(\zeta). \tag{3.17}$$

Similar to (3.13), we have

$$\mathcal{F}\begin{pmatrix} \delta_1 \\ \vdots \\ \delta_{k+1} \end{pmatrix} = \begin{pmatrix} \widehat{\Phi}(\theta_{k+1}^{1}) \\ \vdots \\ \widehat{\Phi}(\theta_{k+1}^{k+1}) \end{pmatrix}, \tag{3.18}$$

Once again, the linear system (3.18) can be solved by using the inverse discrete cosine tranform in $O(k\log k)$ flops. Since it costs $O(n)$ flops to compute $\widehat{\Phi}(\zeta)$ at each Chebyshev node, the total cost for computing the right hand side in (3.18) is $O(n\,k)$ flops.

Now we use Lemma 3.2 to estimate the error in approximating $\widehat{\Phi}(\zeta)$ using $\mathcal{M}_k(\zeta)$.

$$\left|\widehat{\Phi}(\zeta) - \mathcal{M}_k(\zeta)\right| \le \sum_{i=1}^{n} |x_i| \frac{1}{|T_{k+1}(1/\beta_i)|\,(1-|\beta_i|)} \le \frac{3}{2\,T_{k+1}(3)} \sum_{i=1}^{n} |x_i|,$$

where we have used equation (3.8) and the fact that $|\beta_i| \le 1/3$.

It is easy to verify that condition (3.15) implies that

$$\left|d_i^2 - \omega^2\right| \le \frac{4}{3}\left|c-\omega^2\right|.$$

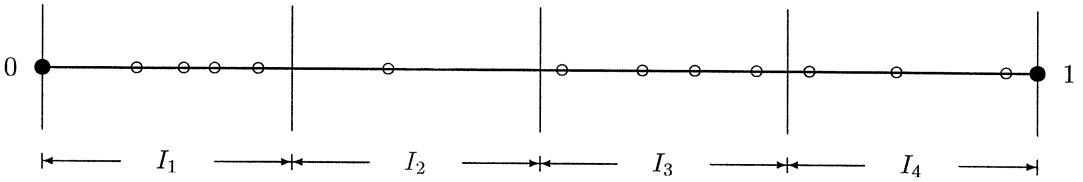

FIG. 3.3. *Mesh Level 2*

With this relationship, we have

$$\left|\Phi(\omega) - \frac{1}{c-\omega^2}\mathcal{M}_k(\zeta)\right| = \frac{\left|\widehat{\Phi}(\zeta) - \mathcal{M}_k(\zeta)\right|}{|c-\omega^2|} \leq \frac{3}{2\,|c-\omega^2|\,T_{k+1}(3)}\sum_{i=1}^{n}|x_i|$$

$$\leq \frac{2}{T_{k+1}(3)}\sum_{i=1}^{n}\frac{|x_i|}{|d_i^2-\omega^2|}. \tag{3.19}$$

REMARK 3.3. *We have avoided a multipole expansion directly on $\Phi(\omega)$. It turns out that such an expansion would not satisfy an error bound of the form (3.19), which will be essential in guaranteeing numerical stability in computing the expansions.*

Now we consider the problem of evaluating $\Phi(\omega)$ for distributions of $\{d_i\}_{i=1}^n$ and $\{\omega_j\}_{j=1}^n$ satisfying (??) and (3.5).

To illustrate how the expansions can be used to speed up the computation, we divide the interval $(0,1)$ into four disjoint subintervals of the same size (see Figure 3.3). Rewrite $\Phi(\omega)$ as:

$$\begin{aligned}\Phi(\omega) &= \sum_{i=1}^{n}\frac{x_i}{d_i^2-\omega^2}\\ &= \sum_{d_i\in I_1}\frac{x_i}{d_i^2-\omega^2} + \sum_{d_i\in I_2}\frac{x_i}{d_i^2-\omega^2} + \sum_{d_i\in I_3}\frac{x_i}{d_i^2-\omega^2} + \sum_{d_i\in I_4}\frac{x_i}{d_i^2-\omega^2}\\ &= \Phi_{I_1}(\omega) + \Phi_{I_2}(\omega) + \Phi_{I_3}(\omega) + \Phi_{I_4}(\omega)\,.\end{aligned}$$

We can use a local expansion to compute $\Phi_{I_l}(\omega)$ for $\omega \in I_k$ as long as I_l is separated from I_k. This is more efficient than the direct computation if the numbers of singularities in I_l and I_k are larger than p. For example, when $\{d_i\}_{i=1}^n$ and $\{\omega_j\}_{j=1}^n$ are evenly distributed on the interval $(0,1)$, each I_k contains roughly $n/4$ singularities. According to Section 3.2, using local expansions this way is more efficient than the direct computation when $n > 4p$.

On the other hand, to compute $\Phi_{I_k}(\omega)$ for $\omega \in I_l$ when I_k is adjacent or equal to I_l, we can continue to halve I_l and I_k into disjoint subintervals of the same size and apply the above techniques until the number of singularities in these intervals is small (see Figure 3.4).

This process generates a hierarchy of intervals of various sizes. We will compute a local expansion and a multipole expansion for each of these intervals. While the basic ideas in Section 3.2 indicate that such expansions could lead to an efficient way of evaluating $\Phi(\omega)$, the basic ideas alone are not enough for fast evaluation of $\Phi(\omega)$. In the following we provide a systematic scheme to find all the expansions quickly.

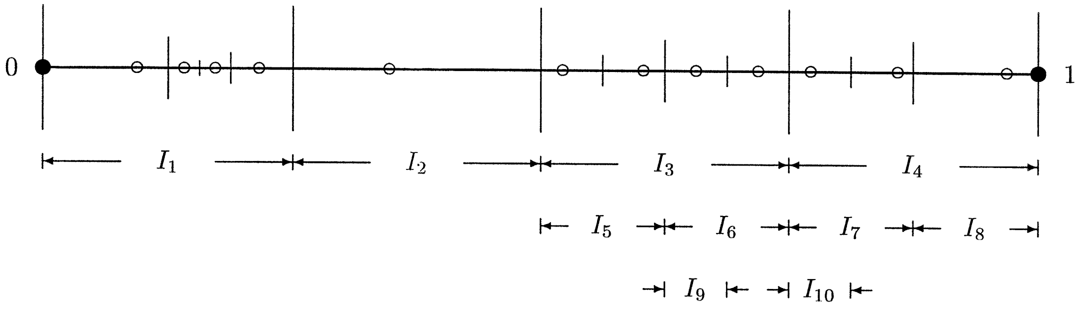

FIG. 3.4. *The Computation Tree*

3.3. The Computation Tree. Consider the interval depicted in Figure 3.3. n singularities are arbitrarily distributed in $(0, 1)$. In order to evaluate $\Phi(\omega)$ through local expansions, we introduce a hierarchy of meshes which refine the interval $(0, 1)$ into smaller and smaller intervals. A tree structure, the *computation tree*, is imposed on this mesh hierarchy.

Mesh level 2 corresponds to the four subintervals in Figure 3.3. Given a fixed parameter s to be specified in Section 3.5, for each interval I in mesh level l with $l \geq 2$, if the number of singularities in I is at least $s + 1$, we divide I into two disjoint subintervals of the same size. These two subintervals are placed in mesh level $l + 1$. But empty subintervals are not placed in mesh level $l + 1$ and are ignored by the subsequent process. The interval I is a *parent interval* and is the *parent* of the subintervals. The subintervals in mesh level $l + 1$ are the *children* of I. An interval is a *childless interval* if the number of singularities in it is at most s. Figure 3.4 is a computation tree with $n = 12$ and $s = 1$.

Let I and $\breve{I}$ be intervals on mesh levels l and k, respectively. If $k < l$, then $\breve{I}$ is a *coarser* interval. If $k = l$, then $\breve{I}$ and I are on the same mesh level.

Let I and $\breve{I}$ be adjacent intervals on mesh levels l and k, respectively, with $k \leq l$. Then $\breve{I}$ is a *neighbor* of interval I if $\breve{I}$ is childless or $k = l$. In Figure 3.4, interval I_8 has one neighbor I_7, and interval I_5 has two neighbors I_2 and I_6. However, I_5 is not a neighbor of I_2. This notion of neighborship is not commutative in general.

Let $\tilde{I}$ be a neighbor of interval I's parent. If $\tilde{I}$ is not adjacent to I and is childless, then $\tilde{I}$ is a *colleague* of I. If $\tilde{I}$ is not childless, then its children that are not adjacent to I are *colleagues* of I. In Figure 3.4, interval I_{10} has two colleagues I_8 and I_9. The colleagueship is not commutative in general.

LEMMA 3.3. *Let I be an interval on the computation tree.*

- *If I and its colleagues are on the same mesh level, then I can have at most* 3 *colleagues;*
- *I can have at most one coarser colleague, and when it does,*
  - *the colleague must be childless;*
  - *I can have at most two colleagues;*

Let c and r be the center and the length of I, respectively. And let $\tilde{I}$ be a colleague of I. Then relation (3.10) holds for $\omega \in I$ and all $d_i \in \tilde{I}$. Thus for $\omega \in I$, we can replace $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ by a local expansion centered at c. Similarly, relation (3.15) holds for $\omega \in \tilde{I}$ and and all $d_i \in I$. Thus for $\omega \in \tilde{I}$, we can replace $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ by a multipole expansion centered at c.

For each interval I, we denote by $\aleph(I)$ the union of I and its neighbors. We denote by $\Re(I)$ the union of the colleagues of I. The relationship in Lemma 3.4 is easy to establish.

LEMMA 3.4. *Let $\bar{I}$ be the parent of I, then*

$$\aleph(\bar{I}) = \aleph(I) + \Re(I) \ .$$

We denote by $\mathbf{N_{lev}}$ the number of mesh levels in the mesh hierarchy. Since the singularities are at least $\tau/2$ apart (see (??)), $\mathbf{N_{lev}}$ can be at most $|\log_2 \tau/2| = O(|\log_2 \epsilon|)$.

We denote by $\mathbf{N_{chl}}$, $\mathbf{N_{par}}$ and $\mathbf{N_{tree}}$ the number of child intervals, parent intervals and intervals in the computation tree. Then

LEMMA 3.5.

$$\mathbf{N_{chl}} \leq \frac{2n}{s}\,\mathbf{N_{lev}} \quad , \quad \mathbf{N_{par}} \leq \frac{n}{s}\,\mathbf{N_{lev}} \quad \textit{and} \quad \mathbf{N_{tree}} \leq \frac{3n}{s}\,\mathbf{N_{lev}} \ .$$

Proof. By construction, every parent interval can have at most two child intervals. Thus, the number of childless intervals is at most twice the number of parent intervals. On the other hand, there can be at most $\lfloor n/s \rfloor$ parent intervals on each mesh level. Hence the results hold. □

3.4. Computing the Coefficients of Local and Multipole Expansions. Consider an interval I on the computation tree with center c and length $2r$. We let $\mathcal{M}_I(\zeta)$ be a multipole expansion of the form (3.17) with $\zeta = 3r/\left(c - \omega^2\right)$ such that $1/(c-\omega^2)\,\mathcal{M}_I(\zeta)$ approximates $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$. A multipole expansion will be computed for every parent interval. Since singularities that are not in $\aleph(I)$ are well-separated from points in I, we also let $\mathcal{L}_I(\xi)$ be a local expansion of the form (3.12) with $\xi = \left(\omega^2 - c\right)/r$ such that $\mathcal{L}_I(\xi)$ approximates $\sum_{d_i \notin \aleph(I)} x_i/(d_i^2 - \omega^2)$.

Consider a split of the sum in $\Phi(\omega)$ as follows:

$$\begin{aligned} \Phi(\omega) &= \sum_{i=1}^{n} \frac{x_i}{d_i^2 - \omega^2} \\ &= \sum_{d_i \notin \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} + \sum_{d_i \in \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} \ . \end{aligned} \tag{3.20}$$

We can compute $\Phi(\omega)$ for $\omega \in I$ by evaluating both $\mathcal{L}_I(\xi)$ and $\sum_{d_i \in \aleph(I)} x_i/(d_i^2 - \omega^2)$. In the following we show how to systematically compute the coefficients of $\mathcal{L}_I(\xi)$ under the assumption that the multipole expansions have been given. We will consider the problems of computing the multipole expansions and evaluating $\sum_{d_i \in \aleph(I)} x_i/(d_i^2 - \omega^2)$ at the end of this section.

Let $\bar{I}$ be the parent of I with center $\bar{c}$ and length $2\bar{r}$. Instead of computing the coeficients in $\mathcal{L}_I(\xi)$ for I directly, we exploit the following relationship. Lemma 3.4 implies that

$$\sum_{d_i \notin \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} = \sum_{d_i \notin \aleph(\bar{I})} \frac{x_i}{d_i^2 - \omega^2} + \sum_{d_i \in \Re(I)} \frac{x_i}{d_i^2 - \omega^2} . \tag{3.21}$$

We approximate each of the sums on the right hand side in (3.21) separately, and add the results together to get $\mathcal{L}_I(\xi)$.

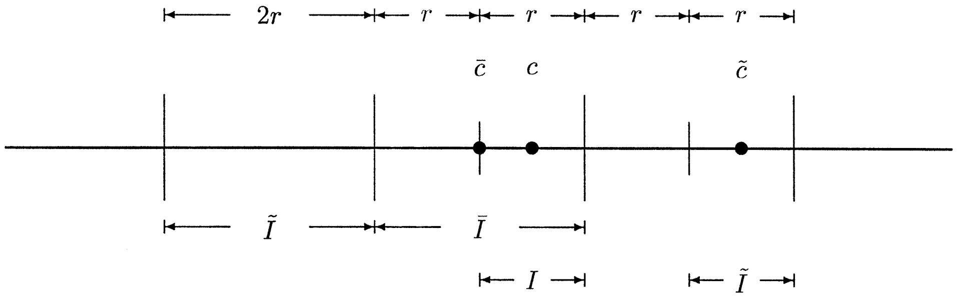

FIG. 3.5. *Conversion Among Local and Multipole Expansions*

The first sum is precisely the one that is approximated by the local expansion for the interval $\bar{I}$. To establish notation, we let

$$\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi}) = \sum_{j=0}^{k} \bar{\gamma}_j T_j(\bar{\xi}), \quad \text{where} \quad \bar{\xi} = \left(\omega^2 - \bar{c}\right)/\bar{r} \tag{3.22}$$

denote this expansion. Comparing with (3.12), we need to write $\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi})$ in terms of variable ξ (shifting the center of $\mathcal{L}_{\bar{I}}(\bar{\xi})$ to c.)

Since $\Re(I)$ is the union of at most 3 colleagues of I, the second sum in (3.21) can be split into at most 3 sums of the form $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$, where $\tilde{I}$ is a colleague of I. The interval I in Figure 3.5 has only 2 colleagues. If $\tilde{I}$ has a multipole expansion $\mathcal{M}_{\tilde{I}}(\zeta)$, then we compute a local expansion for $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ with it. Otherwise, we approximate $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ by a local expansion of the form (3.12). To find $\mathcal{L}_I(\xi)$, we sum up all these local expansions. Algorithm 3.1 below summarizes this process.

ALGORITHM 3.1. **Computing Local***(I, c, r)*

```
if $I$ does not have a parent then
    $\mathcal{L}_I(\xi) := 0$;
else
    Set $\bar{I}$ := parent of $I$;
    Shift center of $\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi})$ to that of $I$ to get $\mathcal{L}_I(\xi)$.
endif
for each colleague $\tilde{I}$ of $I$ with center $\tilde{c}$ and length $2\tilde{r}$ do
    if $\mathcal{M}_{\tilde{I}}(\tilde{\zeta})$ exists then
        convert $1/\left(\tilde{c} - \omega^2\right) \mathcal{M}_{\tilde{I}}(\tilde{\zeta})$ into a local expansion centered at c;
    else
        compute a local expansion centered at c for $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$;
    endif
    add this expansion to $\mathcal{L}_I(\xi)$;
endfor
```

To complete Algorithm (3.1), we now convert the expansion $\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi})$ in (3.22) into the form of (??). Note that we can write

$$\bar{\xi} = \mu + \nu\,\xi \quad \text{where} \quad \mu = (c - \bar{c})/\bar{r} \quad \text{and} \quad \nu = r/\bar{r}. \tag{3.23}$$

Hence the equation (3.22) can be rewritten as

$$\bar{\mathcal{L}}_{\bar{I}}(\xi) = \sum_{j=0}^{k} \bar{\gamma}_j T_j(\mu + \nu\,\xi). \tag{3.24}$$

Lemma 3.6 below rewrites $T_j(\mu + \nu\,\xi)$ in standard form.

LEMMA 3.6. *The polynomials* $\{T_j(\mu+\nu\,\xi)\}_{j=0}^{k}$ *can be rewritten in standard form as*

$$\begin{pmatrix} T_0(\mu+\nu\,\xi) \\ T_1(\mu+\nu\,\xi) \\ \vdots \\ T_k(\mu+\nu\,\xi) \end{pmatrix} = \mathcal{H}^{(\mu,\nu)} \begin{pmatrix} T_0(\xi) \\ T_1(\xi) \\ \vdots \\ T_k(\xi) \end{pmatrix},$$

where $\mathcal{H}^{(\mu,\nu)}$ *is a* $(k+1)\times(k+1)$ *lower triangular matrix*

$$\mathcal{H}^{(\mu,\nu)} = \begin{pmatrix} h_{0,0}^{(\mu,\nu)} & & & \\ h_{1,0}^{(\mu,\nu)} & h_{1,1}^{(\mu,\nu)} & & \\ \vdots & \vdots & \ddots & \\ h_{k,0}^{(\mu,\nu)} & h_{k,1}^{(\mu,\nu)} & \cdots & h_{k,k}^{(\mu,\nu)} \end{pmatrix},$$

with entries recursively defined as

$$h_{0,0}^{(\mu,\nu)} = 1, \quad h_{1,0}^{(\mu,\nu)} = \mu, \quad h_{1,1}^{(\mu,\nu)} = \nu,$$

and for $j = 1, 2, \cdots, k-1$,

$$\begin{aligned} h_{j+1,0}^{(\mu,\nu)} &= 2\mu h_{j,0}^{(\mu,\nu)} + \nu h_{j,1}^{(\mu,\nu)} - h_{j-1,0}^{(\mu,\nu)}, \\ h_{j+1,1}^{(\mu,\nu)} &= 2\mu h_{j,1}^{(\mu,\nu)} + 3\nu h_{j,0}^{(\mu,\nu)} + \nu h_{j,2}^{(\mu,\nu)} - h_{j-1,1}^{(\mu,\nu)}, \\ h_{j+1,l}^{(\mu,\nu)} &= 2\mu h_{j,l}^{(\mu,\nu)} + \nu \left(h_{j,l-1}^{(\mu,\nu)} + h_{j,l+1}^{(\mu,\nu)}\right) - h_{j-1,l}^{(\mu,\nu)}, \quad \textit{for} \quad l = 2, \cdots, j-1, \\ h_{j+1,j}^{(\mu,\nu)} &= 2\mu h_{j,j}^{(\mu,\nu)} + \nu h_{j,j-1}^{(\mu,\nu)}, \quad h_{j+1,j+1}^{(\mu,\nu)} = \nu h_{j,j}^{(\mu,\nu)}. \end{aligned}$$

Proof. We use mathematical induction. The formulas are obviously true for $j = 0, 1$. Now assume they hold for any $1 \le j < k$. Then according to (3.6),

$$\begin{aligned} T_{j+1}(\mu+\nu\xi) &= 2(\mu+\nu\xi)T_j(\mu+\nu\xi) - T_{j-1}(\mu+\nu\xi) \\ &= 2(\mu+\nu\xi)\sum_{l=0}^{j} h_{j,l}^{(\mu,\nu)} T_l(\xi) - \sum_{l=0}^{j-1} h_{j-1,l}^{(\mu,\nu)} T_l(\xi). \end{aligned}$$

In the above right hand side, we replace $2\xi T_l(\xi)$ by $T_{l+1}(\xi) + T_{l-1}(\xi)$ for $l \ge 1$ and replace $2\xi T_0(\xi)$ by $2T_1(\xi)$. The recursion for $\{h_{j+1,l}\}_{l=0}^{j+1}$ follows from simplifying the resulting expression. □

With this lemma, equation (3.24) becomes

$$\bar{\mathcal{L}}_{\bar{I}}(\xi) = (\bar{\gamma}_0 \;\; \bar{\gamma}_1 \;\; \cdots \;\; \bar{\gamma}_k) \begin{pmatrix} T_0(\mu+\nu\,\xi) \\ T_1(\mu+\nu\,\xi) \\ \vdots \\ T_k(\mu+\nu\,\xi) \end{pmatrix} = \left((\bar{\gamma}_0 \;\; \bar{\gamma}_1 \;\; \cdots \;\; \bar{\gamma}_k)\; \mathcal{H}^{(\mu,\nu)}\right) \begin{pmatrix} T_0(\xi) \\ T_1(\xi) \\ \vdots \\ T_k(\xi) \end{pmatrix}.$$

Comparing with (3.12), we see that $\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi})$ can be written as

$$\bar{\mathcal{L}}_{\bar{I}}(\bar{\xi}) = \sum_{j=0}^{k} \gamma_j T_j(\xi), \quad \textit{where} \quad (\gamma_0 \;\; \gamma_1 \;\; \cdots \;\; \gamma_k) = (\bar{\gamma}_0 \;\; \cdots \;\; \bar{\gamma}_1 \;\; \bar{\gamma}_k)\; \mathcal{H}^{(\mu,\nu)}.$$

(3.25)

The shifting formula in (3.25) is exact. Thus the approximation error in $\mathcal{L}_{\bar{I}}(\bar{\xi})$ remains unchanged when we shift its center.

In Algorithm 3.1 we also need to convert multipole expansions of colleagues into local expansions. Let $\tilde{I}$ be a colleague of I with center $\tilde{c}$ and length $2\tilde{r}$. When the multipole expansion

$$\mathcal{M}_{\tilde{I}}(\tilde{\zeta}) = \sum_{j=0}^{k} \tilde{\delta}_j T_j(\tilde{\zeta}), \quad \tilde{\zeta} = 3\tilde{r}/\left(\tilde{c} - \omega^2\right)$$

exists, the sum $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ is approximated by $1/(\tilde{c} - \omega^2)\, \mathcal{M}_{\tilde{I}}(\tilde{\zeta})$.

To approximate this sum using a local expansion of the form

$$\mathcal{L}_I(\xi) = \sum_{j=0}^{k} \gamma_j T_j(\xi), \quad \xi = \left(\omega^2 - c\right)/r,$$

we should interpolate the sum at the $(k+1)$-st order Chebyshev nodes, as suggested in Section 3.2. However, since a multipole expansion is already available, we will interpolate $1/(\tilde{c} - \omega^2)\, \mathcal{M}_{\tilde{I}}(\tilde{\zeta})$ at the same nodes instead. This allows us to avoid evaluating the sum directly, leading to a large reduction in flops when the number of singularities in $\tilde{I}$ is large. It is easy to verify that

$$\omega^2 = c + r\xi, \quad \text{and} \quad \tilde{\zeta} = 3\tilde{r}/\left(\tilde{c} - \omega^2\right) = \tilde{\nu}/\left(\tilde{\mu} - \xi\right),$$

where $\tilde{\nu} = 3\tilde{r}/r$ and $\tilde{\mu} = \left(\tilde{c} - c\right)/r$. Interpolating $1/(\tilde{c} - \omega^2)\, \mathcal{M}_{\tilde{I}}(\tilde{\zeta})$ on the Chebyshev nodes gives

$$\mathcal{L}_I(r_{k+1}^i) = 1/\left(\tilde{c} - c - r r_{k+1}^i\right) \mathcal{M}_{\tilde{I}}\left(\tilde{\nu}/\left(\tilde{\mu} - r_{k+1}^i\right)\right), \quad i = 1, 2, \cdots, k+1.$$

In terms of the ceofficients of $\mathcal{L}_I(\xi)$ and $\mathcal{M}_{\tilde{I}}(\tilde{\zeta})$, these equations take the matrix form

$$\mathcal{F} \begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_{k+1} \end{pmatrix} = \mathcal{G} \begin{pmatrix} \tilde{\delta}_0 \\ \tilde{\delta}_1 \\ \vdots \\ \tilde{\delta}_k \end{pmatrix}, \quad \text{where} \quad \mathcal{G}_{i,j} = \frac{T_j\left(\tilde{\nu}/\left(\tilde{\mu} - r_{k+1}^i\right)\right)}{r\left(\tilde{\mu} - r_{k+1}^i\right)},$$

or

$$\begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_{k+1} \end{pmatrix} = \tilde{\mathcal{G}} \begin{pmatrix} \tilde{\delta}_0 \\ \tilde{\delta}_1 \\ \vdots \\ \tilde{\delta}_k \end{pmatrix}, \quad \text{where} \quad \tilde{\mathcal{G}} = \mathcal{F}^{-1}\, \mathcal{G}. \tag{3.26}$$

Now we consider the problem of finding the multipole expansions. Note that for any parent interval $\bar{I}$, we have

$$\sum_{d_i \in \bar{I}} \frac{x_i}{d_i^2 - \omega^2} = \sum_{I \text{ is a child of } \bar{I}} \; \sum_{d_i \in I} \frac{x_i}{d_i^2 - \omega^2},$$

where $\bar{I}$ can have either one or two children. we replace each sum $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ by $\mathcal{M}_I(\omega)$. To compute $\mathcal{M}_{\bar{I}}(\omega)$, we convert the center of $\mathcal{M}_I(\omega)$ to the center of $\bar{I}$ for each I and sum up the converted multipole expansions (see Lemma 3.7). Procedure **Multipole**($\bar{I}$) summarizes this process.

```
PROCEDURE Multipole($\bar{I}$)
    1. If $\bar{I}$ is childless, find $\mathcal{M}_{\bar{I}}(\omega)$ directly by (??);
    2. Otherwise, set $\mathcal{M}_{\bar{I}}(\omega) := 0$;
            For each child $I$ of $\bar{I}$:
                $\alpha$. convert $\mathcal{M}_I(\omega)$ into a multipole expansion centered
                      at the center of $\bar{I}$;
                $\beta$. add the multipole expansion to $\mathcal{M}_{\bar{I}}(\omega)$.
```

Lemma 3.7 provides a formula for converting the multipole expansions.

LEMMA 3.7. *Let c and $\bar{c}$ be the centers of intervals I and $\bar{I}$, respectively, with $\bar{I}$ being the parent of I (see Figure 3.5). Let*

$$\sum_{d_i \in I} \frac{x_i}{d_i^2 - \omega^2} = \sum_{j=0}^{\infty} \frac{b_j}{(c^2 - \omega^2)^{j+1}} = \sum_{j=0}^{\infty} \frac{\bar{b}_j}{(\bar{c}^2 - \omega^2)^{j+1}} ,$$

where

$$b_j = \sum_{d_i \in I} x_i \left(d_i^2 - c^2\right)^j \quad \text{and} \quad \bar{b}_j = \sum_{d_i \in I} x_i \left(d_i^2 - \bar{c}^2\right)^j .$$

Then

$$\bar{b}_j = \sum_{k=0}^{j} b_k \left(\begin{array}{c} j \\ k \end{array} \right) \left(c^2 - \bar{c}^2\right)^{j-k} .$$

The coefficients $\{b_j\}$ and $\{\bar{b}_j\}$ are those given in (??) for $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ and $\sum_{d_i \in \bar{I}} x_i/(d_i^2 - \omega^2)$, respectively. The formula for computing $\{\bar{b}_j\}$ from $\{b_j\}$ is exact. Thus, when Procedure **Multipole**($\bar{I}$) is used to find $\mathcal{M}_{\bar{I}}(\omega)$, the truncation error is bounded by $(11/21)^p \sum_{d_i \in \bar{I}} |x_i|/|d_i^2 - \omega^2|$.

To compute the multipole expansions for all intervals on the computation tree, we run the procedure **Multipole**($\bar{I}$) bottom-up on the computation tree. The truncation error of $\mathcal{M}_{\bar{I}}(\omega)$ is (see Lemmas 3.3 and 3.7 and relation (??))

$$\left| \sum_{d_i \in \bar{I}} \frac{x_i}{d_i^2 - \omega^2} - \mathcal{M}_{\bar{I}}(\omega) \right| \leq \left(\frac{11}{21}\right)^p \sum_{d_i \in \bar{I}} \frac{|x_i|}{|d_i^2 - \omega^2|} \tag{3.27}$$

for ω in a colleague of $\bar{I}$.

To compute the local expansions for all intervals on the computation tree, we further run the procedure **Local**(I) top-down on the computation tree. Let $\tilde{I}$ be a colleague of I. For $\omega \in I$, the truncation error in converting $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ into a local expansion is bounded by $(11/21)^p \sum_{d_i \in \tilde{I}} |x_i|/|d_i^2 - \omega^2|$ (see Lemmas 3.3 and ?? and relation (??)). Together with Lemma ??, we have

$$\left| \sum_{d_i \notin \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} - \mathcal{L}_I(\omega) \right| \leq \left(\frac{11}{21}\right)^p \sum_{d_i \notin \aleph(I)} \frac{|x_i|}{|d_i^2 - \omega^2|} \tag{3.28}$$

for $\omega \in I$.

Finally, we consider the problem of computing the second sum $\sum_{d_i \in \aleph(I)} x_i/(d_i^2 - \omega^2)$ in (3.20) for $\omega \in I$. Since I can have at most two neighbors, the sum can be further split into at most 3 sums, one is of the form $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ and the other two are of the form $\sum_{d_i \in \breve{I}} x_i/(d_i^2 - \omega^2)$, where $\breve{I}$ is a neighbor of I. We compute $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ directly. We compute $\sum_{d_i \in \breve{I}} x_i/(d_i^2 - \omega^2)$ via the following procedure:

```
PROCEDURE $\mathbf{Sum}(I, \breve{I}, \omega)$
    1. If $\breve{I}$ is childless, compute $\sum_{d_i \in \breve{I}} x_i/(d_i^2 - \omega^2)$ directly and return;
    2. otherwise set $\mathbf{Sum} := 0$,
            for each child $\breve{I}$ of $\breve{I}$
                $\alpha$. if $\breve{I}$ is adjacent to $I$, then compute $\mathbf{Sum} := \mathbf{Sum} + \mathbf{Sum}(I, \breve{I}, \omega)$;
                $\beta$. if $\breve{I}$ is not adjacent to $I$, then compute $\mathbf{Sum} := \mathbf{Sum} + \mathcal{M}_{\breve{I}}(\omega)$;
    3. return $\mathbf{Sum}$.
```

For each $\breve{I}$, at most one child can be adjacent to I. If $\breve{I}$ is a child of $\breve{I}$ not adjacent to I, then I is a coarser colleague of $\breve{I}$. Similar to (3.27), the truncation error of $\mathbf{Sum}(I, \breve{I}, \omega)$ is

$$\left| \sum_{d_i \in \breve{I}} \frac{x_i}{d_i^2 - \omega^2} - \mathbf{Sum}(I, \breve{I}, \omega) \right| \leq \left(\frac{11}{21} \right)^p \sum_{d_i \in \breve{I}} \frac{|x_i|}{|d_i^2 - \omega^2|} . \tag{3.29}$$

Define

$$\Im(I) = \{\breve{I} : \ I \text{ is a colleague of } \breve{I}.\}$$

We denote $|\Im(I)|$ by the number of intervals in $\Im(I)$.

LEMMA 3.8.

$$\sum_{I \textit{ is childless}} |\Im(I)| \ \leq \ \mathbf{N_{tree}} \, .$$

Proof. We prove Lemma 3.8 by showing that it holds for all the trees generated through the process of constructing the computation tree.

Note that Lemma 3.8 holds for the simplest tree in Figure 3.3. We assume that Lemma 3.8 holds for a tree T (see the tree in Figure 3.6 without the thick vertical bar in the middle). We also assume that on tree T intervals up to mesh level k have been generated and we are in the process of generating intervals on mesh level $k+1$. Let I be a childless interval of T on mesh level k. We divide I into two disjoint subintervals I_1 and I_2 to get tree T' (see Figure 3.6). We prove Lemma 3.8 by showing that it holds for T'.

For simplicity, we further assume that both I_1 and I_2 are non-empty so that they are all on mesh level $k+1$. The case where one of I_1 and I_2 is empty is similar. By dividing I into two disjoint subintervals, we loss one childless interval (I becomes a parent interval). In the mean time, we gain two childless intervals. Since there does not exist an interval that is in both $\Im(I_1)$ and $\Im(I_2)$, and since any interval in $\Im(I_1)$ or $\Im(I_2)$ must be in $\Im(I)$, we have

$$|\Im(I_1)| + |\Im(I_2)| \leq |\Im(I)| \, .$$

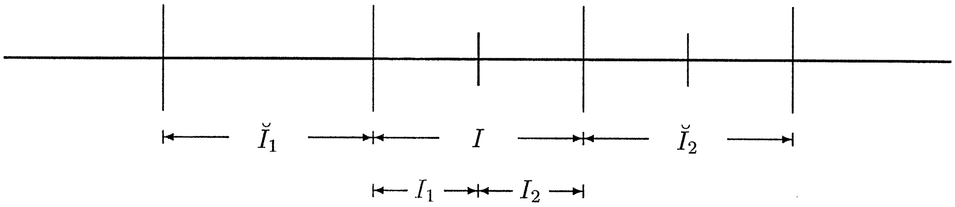

FIG. 3.6. *Tree T'*

On the other hand, for any other childless interval $\check{I}$ in T, the value $\Im(\check{I})$ can change from T to T' only if $\check{I}$ is not on level $k+1$ and I is a neighbor of $\check{I}$. There can be at most two such childless intervals in T. Assume that $\check{I}$ is such an interval. The situation is illustrated in Figure 3.6 with $\check{I} = \check{I}_1$. $\Im(\check{I})$ gets a new childless interval I_2. Thus $|\Im(\check{I})|$ is increased by 1 and the sum

$$\sum_{I \text{ is childless}} |\Im(I)|$$

increases by at most 2. Since $\mathbf{N_{tree}}$ is increased by 2, Lemma 3.8 holds for T'. □

3.5. The Algorithm. The following algorithm computes $\Phi(\omega)$ of (3.4) for $\{d_i\}_{i=1}^n$ and $\{\omega_j\}_{j=1}^n$ satisfying (??) and (3.5).

```
ALGORITHM Modified FMM
  1. Establish a computation tree on $(0,1)$ (see Section 3.3);
  2. Do $j = \mathbf{N_{lev}}, 2, -1$
       For each interval $\bar{I}$ on the $j$-th mesh level, compute Multipole($\bar{I}$);
  3. Do $j = 2, \mathbf{N_{lev}}, 1$
       For each interval $I$ on the $j$-th mesh level, compute Local($I$);
  4. For each childless interval $I$ and each $\omega_j \in I$:
           $\alpha$. $\Phi(\omega_j) = \mathcal{L}_I(\omega_j) + \sum_{d_i \in I} x_i/(d_i^2 - \omega_j^2)$;
           $\beta$. For each neighbor $\check{I}$ of $I$:
                $\Phi(\omega_j) := \Phi(\omega_j) + \mathbf{Sum}(I, \check{I}, \omega_j)$.
```

For any ω, the truncation error of the modified **FMM** is the summation of the truncation errors in $\mathcal{L}_I(\omega)$ and $\mathbf{Sum}(I, \check{I}, \omega)$. Relations (3.28) and (3.29) imply that the truncation error of the modified **FMM** is

$$O\left(\epsilon \sum_{i=1}^{n} |x_i|/|d_i^2 - \omega^2|\right).$$

In Section 3.6, we discuss numerical issues related to finding and evaluating the local and multipole expansions.

Step 1 of the modified **FMM** takes $O(n\mathbf{N_{lev}})$ floating point operations.

In step 2 of the modified **FMM**, computing the multipole expansion of a childless interval takes about $2ps$ floating point operations; and computing the multipole expansion of a parent interval takes about $2p^2$ floating point operations. Thus the total cost of step 2 is about

$$2ps \cdot \mathbf{N_{chl}} + 2p^2 \cdot \mathbf{N_{par}} \leq 2pn\left(2 + \frac{p}{s}\right) \mathbf{N_{lev}}$$

floating point operations (see Lemma 3.5).

In step 3, shifting the center of a local expansion or converting a multipole expansion to a local expansion takes about p^2 floating point operations; and computing the local expansion of a childless colleague takes about $2ps$ floating point operations. By Lemma 3.3, an interval can have three colleagues only if they are on the same mesh level; and an interval can have at most one coarser colleague. Thus the total cost of step 3 is about

$$\left(p^2 + \max(p^2 + 2ps, 3p^2)\right) \mathbf{N_{tree}} \leq 6pn(p + \max(p, s))/s\mathbf{N_{lev}}$$

floating point operations (see Lemma 3.5).

In step 4, each childless interval I may interact with at most 3 childless intervals, $|\Im(I)|$ multipole expansions and one local expansion. Interacting with a childless interval and an expansion take about $4s^2$ and $2ps$ floating point operations, respectively. Thus the total cost of step 4 is

$$(12s^2 + 2ps) \cdot \mathbf{N_{chl}} + 2ps \quad \cdot \sum_{I \text{ is childless}} |\Im(I)| \leq 2n\ (12s + 5p)\ \mathbf{N_{lev}}$$

floating point operations (see Lemmas 3.5 and 3.8).

Summarizing, the total cost of the algorithm is about

$$\left(14p + 24s + \frac{8p^2 + 6p\max(p, s)}{s}\right) n\mathbf{N_{lev}}$$

floating point operations. To minimize the total cost, we take $s = p/\sqrt{2}$. The total cost under this choice of s is about $48pn\mathbf{N_{lev}}$ floating point operations. Since $p = O(|\log_2 \epsilon|)$ and $\mathbf{N_{lev}} = O(|\log_2 \epsilon|)$, this total cost is of the order $O(n \log_2^2 \epsilon)$.

Remark 1: The modified **FMM** is designed primarily for large n. As mentioned at the end of Section 3.2, the basic ideas there results in faster methods than the direct method when $n > 4p$.

Remark 2: The algorithm is for a distribution of $\{d_i\}_{i=1}^n$ satisfying (??). This implies that $n \leq 1/\epsilon$ and thus $\log_2 n \leq |\log_2 \epsilon|$. In practice $|\log_2 \epsilon|/\log_2 n$ is a moderate constant. Thus the total cost is like $O(n \log_2^2 n)$ floating point operations.

Remark 3: If the singularities $\{d_i\}_{i=1}^n$ are evenly distributed, then the tree height $\mathbf{N_{lev}}$ is more like $\log_2 n$; the numbers of parent and childless intervals are more like n/s; each interval is likely to have three colleague; and the sum of $|\Im(I)|$ becomes negligible. The total cost is more like

$$\left(6p + 12s + \frac{6p^2}{s}\right) n \cdot \log_2 n$$

floating point operations. When we take $s = p/\sqrt{2}$, the total cost is about $23pn \log_2 n$ floating point operations.

Remark 4: We can replace the local and multipole expansions (??) and (??) by faster convergent expansions, e.g, Chebyshev expansions. This leads to much smaller values of p and thus much faster versions of **FMM**.

Remark 5: Since we use the modified **FMM** for n different functions of the form $\Phi(\omega)$ of (3.4) at the same distributions of $\{d_i\}_{i=1}^n$ and $\{\hat{\omega}_j\}_{j=1}^n$ (see the begining of Section 3), we can pre-compute step 1 and all the quantities that are unrelated to $\{x_i\}_{i=1}^n$ for all $\Phi(\omega)$.

Remark 6: The modified **FMM** can be used to evaluate the function $\mathcal{G}(\mu)$ in Section 2.2 at $O(n)$ points in $O(n \log_2^2 \epsilon)$ floating point operations. It can also be generalized to evaluate the function

$$\sum_{i=1}^{n} \frac{|z_i|}{|d_i^2 - \omega^2|}$$

at $O(n)$ points in $O(n \log_2^2 \epsilon)$ floating point operations. This generalized version can in turn be used to check the stopping criterion (2.8) at $O(n)$ points in $O(n \log_2^2 \epsilon)$ floating point operations. Thus we can compute all the singular values of M in (2.1) in $O(n \log_2^2 \epsilon)$ floating point operations (see Section 2.2).

3.6. Some Numerical Issues in Finding and Evaluating the Expansions. In this section we discuss some numerical issues related to finding and evaluating the local and multipole expansions.

According to our arithmetic model (see Section 1), we can compute the left boundary, the right boundary, length, and center of each interval on the computation tree *exactly*. We show this by induction. Assume that we are given the left boundary $\bar{\xi}$, the right boundary $\bar{\nu}$, the length $\bar{r}$, and the center $\bar{c}$ of a parent interval $\bar{I}$ *exactly*. This assumption is true when $\bar{I}$ is an interval on mesh level 2. Then $\bar{r} = 2^{-l}$ where $l < |\log_2 \epsilon|$ is the mesh level of $\bar{I}$. We compute the corresponding quantities of the children of $\bar{I}$ exactly. In fact $r = \bar{r}/2$ is the length of the children of $\bar{I}$. For the left child, the left and right boundaries are $\bar{\xi}$ and $\bar{c}$, respectively, and the center is $\bar{\xi} + r$. For the right child, the left and right boundaries are $\bar{c}$ and $\bar{\nu}$, respectively, and the center is $\bar{c} + r$. The quantities $\bar{r}/2$, $\bar{\xi} + r$ and $\bar{c} + r$ are computed exactly.

In procedures **Local**(I) and **Multipole**(I) for finding $\mathcal{L}_I(\omega)$ and $\mathcal{M}_I(\omega)$, we need to compute quantities of the form $c^2 - d_i^2$ and $\bar{c}^2 - c^2$ and their powers, where c and $\bar{c}$ are the centers of intervals, and d_i is a singularity of $\Phi(\omega)$ of (3.4). To reduce the effects of rounding errors, we compute $c^2 - d_i^2$ and $\bar{c}^2 - c^2$ to high relative accuracy as $(c - d_i)(c + d_i)$ and $(\bar{c} - c)(\bar{c} + c)$, respectively. Similarly we computer their powers to high relative accuracy.

In procedure **Sum**($I, \check{I}, \omega$) and in evaluating the local and multipole expansions at the points $\{\omega_j\}_{j=1}^n$ (see (3.5)), we need to compute quantities of the form $c^2 - \omega_j^2$ and their powers, where c is the center of an interval. As before we can compute $c^2 - \omega_j^2$ to high relative accuracy as $(c - \omega_j)(c + \omega_j)$. Similarly we powers of $c^2 - \omega_j^2$ to high relative accuracy.

But the situation is different when we apply the modified **FMM** to compute $\Phi(\omega)$ at the points $\{\hat{\omega}_j\}_{j=1}^n$. Since $\{\hat{\omega}_j\}_{j=1}^n$ are given by sums (see (3.2) and (??)):

$$\hat{\omega}_j = d_j + \hat{\mu}_j \quad \text{or} \quad \hat{\omega}_j = d_{j+1} + \hat{\mu}_j$$

with

$$0 < d_1 < \hat{\omega}_1 < d_2 < \ldots < d_n < \hat{\omega}_n < 1 .$$

To reduce the effects of rounding errors, we want to compute $c^2 - \hat{\omega}^2$ to high relative accuracy, where $\hat{\omega}$ is one of $\{\hat{\omega}_j\}_{j=1}^n$. According to the arithmetic model, we have

$$\mathbf{fl}(\hat{\omega}) = \hat{\omega}(1 + \xi) ,$$

where $|\xi| \leq \epsilon$. Thus we can compute $c + \hat{\omega}$ to high relative accuracy as $\mathbf{fl}(c + \mathbf{fl}(\hat{\omega}))$. In the following we present a scheme for computing $c - \hat{\omega}$ to high relative accuracy. With this scheme, $c^2 - \hat{\omega}^2$ is computed to high relative accuracy as $(c - \hat{\omega})(c + \hat{\omega})$.

One of the basic ideas of this scheme is to compute a representation of $\hat{\omega}$ for each interval $\hat{\omega}$ is in. To be more specific, assume that for a parent interval $\bar{I}$ with $\hat{\omega} \in \bar{I}$, we are given an exact representation $\hat{\omega} = \bar{\xi} + \bar{d} + \bar{\mu}$, where $\bar{\xi}$ is the left boundary of $\bar{I}$. If $\bar{I}$ is the whole interval $(0, 1)$, then $\bar{\xi} = 0$ and $\bar{d}$ and $\bar{\mu}$ are given by (3.2). $\hat{\omega}$ must be in a child interval I of $\bar{I}$. Let $r = \bar{r}/2$ be the length of I, Procedure $\mathbf{Shift}(\bar{d}, \bar{\mu}, r)$ finds out I and computes d and μ in the representation $\hat{\omega} = \xi + d + \mu$, where ξ is the left boundary of I.

```
PROCEDURE Shift($\bar{d}, \bar{\mu}, r$)
    1. Let $d = \max(\bar{d}, \bar{\mu})$ and $\mu = \min(\bar{d}, \bar{\mu})$;
    2. If $d \geq r$, then
            $\alpha$. Let $d = d - r$;
            $\beta$. If $d + \mu > 0$, then
                    $I$ is the right child interval. Return $d$ and $\mu$;
            $\gamma$. Otherwise
                    $I$ is the left child interval. Return $\bar{d}$ and $\bar{\mu}$;
    3. Otherwise if $d \geq r/2$, then
            $d = d - r/2$ and return Shift($d, \mu, r$);
    4. Otherwise
            $I$ is the left child interval. Return $d$ and $\mu$.
```

We run procedure $\mathbf{Shift}(\bar{d}, \bar{\mu}, r)$ top-down on the computation tree to compute these representations of $\{\hat{\omega}_j\}_{j=1}^{n}$. The total cost of this computation is $O(n|\log_2 \epsilon|)$ floating point operations.

Let ξ, ν and r be the left boundary, the right boundary and the length of a childless interval I. Given a representation $\hat{\omega} = \xi + d + \mu$, we discuss the problem of computing $c - \hat{\omega}$ to high relative accuracy, where c is the center of an interval on the computation tree.

If $c \leq \xi$, then we compute $c - \xi < 0$ and $d + \mu > 0$ to high relative accuracy, and thus we can compute $c - \hat{\omega}$ to high relative accuracy as $(c - \xi) - (d + \mu)$.

If $c \geq \nu$, then

$$c - \hat{\omega} = (c - \nu) + (r - d - \mu)\ . \tag{3.30}$$

Since $c - \nu > 0$ and $r - d - \mu > 0$, if we can compute $r - d - \mu$ to high relative accuracy, then we can compute $c - \hat{\omega}$ to high relative accuracy as above. We may need to compute $c - \hat{\omega}$ for many values of c, but we only need to compute the value $r - d - \mu$ once.

If $\xi < c < \nu$, then c must be the center of I. Thus $c = \xi + r/2$ and hence

$$c - \hat{\omega} = r/2 - d - \mu\ . \tag{3.31}$$

Thus if we can compute $r/2 - d - \mu$ to high relative accuracy, then we can compute $c - \hat{\omega}$ to high relative accuracy.

Procedure $\mathbf{Rel}(d, \mu, \gamma)$ computes $d + \mu - \gamma$ to high relative accuracy, where d, μ and $\gamma = 2^{-k}$ with $k > 0$ are floating point numbers such that $0 \leq d + \mu \leq 2\gamma$.

```
PROCEDURE $\mathbf{Rel}(d, \mu, \gamma)$
    1. Let $\alpha = \max(d, \mu)$ and $\beta = \min(d, \mu)$;
    2. If $\alpha \geq \gamma$, then
           $\alpha = \alpha - \gamma$ and $\mathbf{Rel}(d, \mu, \gamma) = \alpha + \beta$;
    3. Otherwise if $\alpha \geq \gamma/2$, then
           $\alpha = \alpha - \gamma/2$ and $\mathbf{Rel}(d, \mu, k) = \mathbf{Rel}(\alpha, \beta, \gamma/2)$;
    4. Otherwise if $\alpha \geq \gamma/4$ and $\beta \geq \gamma/4$, then
           $\alpha = \alpha - \gamma/4$, $\beta = \beta - \gamma/4$ and $\mathbf{Rel}(d, \mu, k) = \mathbf{Rel}(\alpha, \beta, \gamma/2)$;
    5. Otherwise $\mathbf{Rel}(d, \mu, k) = \alpha + \beta - \gamma$.
```

For each $\hat{\omega}$, we run $\mathbf{Rel}(d, \mu, r)$ and $\mathbf{Rel}(d, \mu, r/2)$ to compute $r - d - \mu$ and $r/2 - d - \mu$ to high relative accuracy, respectively. Hence $c - \hat{\omega}$ is computed to high relative accuracy in (3.30) and (3.31). The total cost of this computation is $O(n|\log_2 \epsilon|)$ floating point operations.

4. Numerical Stability. As the major result of this section, we show that the modified **FMM** satisfies

$$\mathbf{fl}_{\mathbf{f}}\left(\Phi(\hat{\omega}_j)\right) = \Phi(\hat{\omega}_j) + O\left(\epsilon \sum_{i=1}^{n} \frac{|x_i|}{|d_i^2 - \hat{\omega}_j^2|}\right) \tag{4.1}$$

for the function $\Phi(\omega)$ in equation (3.4) and the points $\{\hat{\omega}_i\}_{i=1}^{n}$ of (3.2), where $\mathbf{fl}_{\mathbf{f}}\left(\Phi(\hat{\omega}_j)\right)$ is the floating point result of using the modified **FMM**. The error term in (4.1) includes both truncation errors and round-off errors. In Section 3.5 we have shown that the truncation errors satisfy the error term in (4.1). In this section, we show that the round-off errors also satisfy the error term in (4.1).

To be more specific, we note that the modified **FMM** splits $\Phi(\omega)$ as follows (see (3.20))

$$\Phi(\omega) = \sum_{d_i \notin \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} + \sum_{d_i \in \aleph(I)} \frac{x_i}{d_i^2 - \omega^2},$$

where I is the childless interval such that $\omega \in I$. The modified **FMM** replaces the first sum by a local expansion, and replaces the second sum by a sequence of multipole expansions and some direct sums of the form $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$.

As noted at the end of Section 2.2, we can compute each ratio in $\sum_{d_i \in I} x_i/(d_i^2 - \omega^2)$ to high relative accuracy. Thus the round-off error in this direct computation is of the form $O\left(\epsilon \sum_{d_i \in I} |x_i|/|d_i^2 - \omega^2|\right)$. In Section 4.1, we show that

$$\left|\mathbf{fl}\left(\mathcal{M}_{\bar{I}}(\hat{\omega})\right) - \mathcal{M}_{\bar{I}}(\hat{\omega})\right| = O\left(\epsilon \sum_{d_i \in \bar{I}} \frac{|x_i|}{|d_i^2 - \hat{\omega}^2|}\right). \tag{4.2}$$

And in Section 4.2, we show that

$$\left|\mathbf{fl}\left(\mathcal{L}_I(\hat{\omega})\right) - \mathcal{L}_I(\hat{\omega})\right| = O\left(\epsilon \sum_{d_i \notin \aleph(I)} \frac{|x_i|}{|d_i^2 - \hat{\omega}^2|}\right). \tag{4.3}$$

Relation (4.1) thus follows by combining these results.

As the second major result, in Section 4.3 we further show that relation (4.1) implies that the modified **FMM** stably computes the matrix-matrix product $V\hat{Q}$.

4.1. Round-off Errors in the Multipole Expansions.

We consider the rounding errors in computing the coefficients of

$$\mathcal{M}_{\bar{I}}(\omega) = \sum_{j=0}^{p-1} \frac{\bar{b}_j}{\left(\bar{c}^2 - \omega^2\right)^{j+1}} \approx \sum_{d_i \in \bar{I}} \frac{x_i}{d_i^2 - \omega^2} , \tag{4.4}$$

where (see Lemma 3.7)

$$\bar{b}_j = \sum_{d_i \in \bar{I}} x_i \left(d_i^2 - \bar{c}^2\right)^j . \tag{4.5}$$

First assume that $\bar{I}$ is a childless interval. In this case $\{\bar{b}_j\}_{j=1}^{p-1}$ are computed from (4.5). Since each difference $d_i^2 - \bar{c}^2$ is computed to high relative accuracy (see Section 3.6), each term in the sum in (4.5) is also computed to high relative accuracy. Thus

$$\mathbf{fl}(\bar{b}_j) = \sum_{d_i \in \bar{I}} x_i \left(d_i^2 - \bar{c}^2\right)^j \left(1 + \varrho_{i,j}\right) , \tag{4.6}$$

where[3] $|\varrho_{i,j}| \le \varrho$ with ϱ being a small multiple of $p\,\epsilon$ that is independent of ϵ. This implies that

$$\left|\mathbf{fl}(\bar{b}_j) - \bar{b}_j\right| \le \varrho \sum_{d_i \in \bar{I}} |x_i| \left|d_i^2 - \bar{c}^2\right|^j . \tag{4.7}$$

Now assume that $\bar{I}$ is a parent interval. For simplicity, we first assume that $\bar{I}$ has only one child interval I. We consider the rounding errors in computing the coefficients $\{\bar{b}_j\}_{j=1}^{p-1}$ of $\mathcal{M}_{\bar{I}}(\omega)$ in (4.4). According to Procedure **Multipole**($\bar{I}$) and Lemma 3.7, the coefficients $\{\bar{b}_j\}_{j=1}^{p-1}$ are computed as

$$\bar{b}_j = \sum_{k=0}^{j} b_k \begin{pmatrix} j \\ k \end{pmatrix} \left(c^2 - \bar{c}^2\right)^{j-k} \quad \text{and} \quad b_k = \sum_{d_i \in I} x_i \left(d_i^2 - c^2\right)^k . \tag{4.8}$$

We determine the rounding errors in $\{\bar{b}_j\}_{j=1}^{p-1}$ by induction. In light of (4.1) and (4.7), we assume that the rounding errors in $\{\bar{b}_j\}_{j=1}^{p-1}$ and $\{b_k\}_{k=1}^{p-1}$ have the form

$$\left|\mathbf{fl}(\bar{b}_j) - \bar{b}_j\right| \le \sum_{d_i \in \bar{I}} |x_i|\, \theta_{i,\bar{I}}\, f_{i,\bar{I}}^j \quad \text{and} \quad \left|\mathbf{fl}(b_j) - b_j\right| \le \sum_{d_i \in I} |x_i|\, \theta_{i,I}\, f_{i,I}^j . \tag{4.9}$$

Since the difference $c^2 - \bar{c}^2$ is computed to high relative accuracy (see Section 3.6), each term in the sum of $\bar{b}_j$ in (4.8) is also computed to high relative accuracy. Thus

$$\mathbf{fl}(\bar{b}_j) = \sum_{k=0}^{j} \mathbf{fl}(b_k) \begin{pmatrix} j \\ k \end{pmatrix} \left(c^2 - \bar{c}^2\right)^{j-k} \left(1 + \bar{\varrho}_{k,j}\right) , \tag{4.10}$$

[3] Here and elsewhere in Section 4 we use ϱ as the same upper bound for similar round-off errors.

where $|\bar{\varrho}_{k,j}| \leq \varrho$. This implies that, by using Lemma 3.2,

$$\begin{aligned}
\left|\mathbf{fl}(\bar{b}_j) - \bar{b}_j\right| \leq \; & (1+\varrho) \sum_{k=0}^{j} |\mathbf{fl}(b_k) - b_k| \begin{pmatrix} j \\ k \end{pmatrix} \left|c^2 - \bar{c}^2\right|^{j-k} \\
& + \varrho \sum_{k=0}^{j} |b_k| \begin{pmatrix} j \\ k \end{pmatrix} \left|c^2 - \bar{c}^2\right|^{j-k} \\
\leq \; & (1+\varrho) \sum_{d_i \in I} |x_i| \, \theta_{i,I} \sum_{k=0}^{j} f_{i,I}^{k} \begin{pmatrix} j \\ k \end{pmatrix} \left|c^2 - \bar{c}^2\right|^{j-k} \\
& + \varrho \sum_{d_i \in I} |x_i| \sum_{k=0}^{j} \left|d_i^2 - c^2\right|^{k} \begin{pmatrix} j \\ k \end{pmatrix} \left|c^2 - \bar{c}^2\right|^{j-k} \\
\leq & \sum_{d_i \in I} |x_i| \left((1+\varrho)\theta_{i,I} + \varrho\right) \left(\max\left(f_{i,I}, \left|d_i^2 - c^2\right|\right) + \left|c^2 - \bar{c}^2\right|\right)^j .
\end{aligned} \tag{4.11}$$

Thus in (4.9) we can set

$$\theta_{i,\bar{I}} = (1+\varrho)\theta_{i,I} + \varrho \quad \text{and} \quad f_{i,\bar{I}} = \max\left(f_{i,I}, \left|d_i^2 - c^2\right|\right) + \left|c^2 - \bar{c}^2\right| . \tag{4.12}$$

Now we assume that $\bar{I}$ has two children. In this case the sum of $\bar{b}_j$ in (4.8) is replaced by a summation of such sums over these two children; and the sum on the right hand side of (4.11) is similarly replaced by

$$\sum_{I \text{ is a child of } \bar{I}} \; \sum_{d_i \in I} |x_i| \left((1+\varrho)\theta_{i,I} + \varrho\right) \left(\max\left(f_{i,I}, \left|d_i^2 - c^2\right|\right) + \left|c^2 - \bar{c}^2\right|\right)^j .$$

Thus recursion (4.12) still holds.

Now we solve the recursion (4.12). For any $d_i \in \bar{I}$, let I_k be the childless interval such that $d_i \in I_k$. Also let $\bar{I} \equiv I_1, I_2, \ldots, I_k$ be intervals on the computation tree such that I_j is the parent of I_{j+1} for $1 \leq j \leq k-1$. Let c_j and r_j be the center and the length of I_j, respectively. Relations (4.7) and (4.12) imply that

$$f_{i,I_k} = \left|d_i^2 - c_k^2\right| \quad \text{and} \quad f_{i,I_j} = \max\left(f_{i,I_{j+1}}, \left|d_i^2 - c_{j+1}^2\right|\right) + \left|c_{j+1}^2 - c_j^2\right| ,$$

and

$$\theta_{i,I_k} = \varrho \quad \text{and} \quad \theta_{i,I_j} = (1+\varrho)\, \theta_{i,I_{j+1}} + \varrho .$$

Solving this recursion we have

$$f_{i,\bar{I}} = \left|d_i^2 - c_k^2\right| + \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \quad \text{and} \quad \theta_{i,\bar{I}} = (1+\varrho)^k - 1 . \tag{4.13}$$

Since multipole expansions are computed only for $\mathbf{N_{lev}} - 1$ mesh levels, we have $k \leq \mathbf{N_{lev}} - 1$ and thus

$$\theta_{i,\bar{I}} \leq (1+\varrho)^{\mathbf{N_{lev}}-1} - 1 \approx (\mathbf{N_{lev}} - 1)\varrho \tag{4.14}$$

for any singularity d_i and any interval $\bar{I}$.

Before deriving a bound for $f_{i,\bar{I}}$, we introduce the following lemma.

Lemma 4.1. *Let $I_1, I_2, \ldots, I_k$ be intervals on the computation tree such that I_j is the parent of I_{j+1} for $1 \le j \le k-1$. Let c_j and r_j be the center and the length of I_j, respectively. Then for any $d \in I_k$,*

$$\left|d^2 - c_1^2\right| \le \left|d^2 - c_k^2\right| + \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \le \left(c_1 + \frac{r_1}{2}\right)^2 - c_1^2 \,.$$

Proof. Since

$$d^2 - c_1^2 = \left(d^2 - c_k^2\right) + \sum_{j=1}^{k-1} \left(c_j^2 - c_{j+1}^2\right) \,,$$

taking absolute values on both sides gives the first inequality of Lemma 4.1. We prove the second inequality of Lemma 4.1 by induction on the number of intervals. Lemma 4.1 is true for the case of one interval. Assume that Lemma 4.1 is true for $k-1$ intervals $I_2, \ldots, I_k$, where $k \ge 2$. Then

$$\begin{aligned}\left|d^2 - c_k^2\right| + \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| &= \left|d^2 - c_k^2\right| + \sum_{j=2}^{k-1} \left|c_j^2 - c_{j+1}^2\right| + \left|c_2^2 - c_1^2\right| \\ &\le \left(c_2 + \frac{r_2}{2}\right)^2 - c_2^2 + \left|c_2^2 - c_1^2\right| \,.\end{aligned}$$

Since I_2 is a child of I_1, we have $c_2 = c_1 \pm r_2/2$. The value $c_2 = c_1 + r_2/2$ makes the last relation larger. Plugging this value into the last relation and using the fact that $r_2 = r_1/2$ we have

$$\left|d^2 - c_k^2\right| + \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \le \left(c_1 + \frac{r_1}{2}\right)^2 - c_1^2 \,.$$

Thus Lemma 4.1 is proved. □

Using Lemma 4.1, relation (4.13) implies

$$\left|d_i^2 - \bar{c}^2\right| \le f_{i,\bar{I}} \le \left(\bar{c} + \frac{\bar{r}}{2}\right)^2 - \bar{c}^2 \,. \tag{4.15}$$

We now consider the round-off errors in evaluating $\mathcal{M}_{\bar{I}}(\omega)$ of (4.4) at a point $\hat{\omega}$, where $\hat{\omega}$ is in a colleague of $\bar{I}$ and is one of $\{\hat{\omega}_j\}_{j=1}^{n}$ in (3.2).

Since the difference $\bar{c}^2 - \hat{\omega}^2$ is computed to high relative accuracy (see Section 3.6), similar to (4.10) we have

$$\left|\mathbf{fl}\left(\mathcal{M}_{\bar{I}}(\hat{\omega})\right) - \mathcal{M}_{\bar{I}}(\hat{\omega})\right| \le (1+\varrho) \sum_{j=0}^{p-1} \frac{\left|\mathbf{fl}(\bar{b}_j) - \bar{b}_j\right|}{\left|\bar{c}^2 - \hat{\omega}^2\right|^{j+1}} + \varrho \sum_{j=0}^{p-1} \frac{\left|\bar{b}_j\right|}{\left|\bar{c}^2 - \hat{\omega}^2\right|^{j+1}} \,.$$

Using relations (4.5), (4.9), (4.14) (4.14) and (4.15), and assuming that $\left|\bar{c}^2 - \hat{\omega}^2\right| > f_{i,\bar{I}}$, we have

$$\left|\mathbf{fl}\left(\mathcal{M}_{\bar{I}}(\hat{\omega})\right) - \mathcal{M}_{\bar{I}}(\hat{\omega})\right| \le (1+\varrho)\left((1+\varrho)^{\mathrm{N_{lev}}-1} - 1\right) \sum_{j=0}^{\infty} \frac{\sum_{d_i \in \bar{I}} |x_i| \, f_{i,\bar{I}}^{j}}{\left|\bar{c}^2 - \hat{\omega}_j^2\right|^{j+1}}$$

$$+ \varrho \sum_{j=0}^{\infty} \frac{\sum_{d_i \in \bar{I}} |x_i| \left|d_i^2 - \bar{c}^2\right|^j}{|\bar{c}^2 - \hat{\omega}^2|^{j+1}}$$

$$\leq \left((1+\varrho)^{\mathbf{N}_{\mathbf{lev}}} - 1\right) \sum_{j=0}^{\infty} \frac{\sum_{d_i \in \bar{I}} |x_i| \; f_{i,\bar{I}}^j}{|\bar{c}^2 - \hat{\omega}^2|^{j+1}}$$

$$= \left((1+\varrho)^{\mathbf{N}_{\mathbf{lev}}} - 1\right) \sum_{d_i \in \bar{I}} \frac{|x_i|}{|\bar{c}^2 - \hat{\omega}^2| - f_{i,\bar{I}}} \,.$$

Since $d_i \in \bar{I}$ we have $|d_i - \bar{c}| \leq \bar{r}/2$. Since $\hat{\omega}$ is in a colleague of I, from Lemma 3.3 we have either $|\hat{\omega} - \bar{c}| \leq 3\bar{r}/2$ with $\bar{c} \geq 5\bar{r}/2$ or $\hat{\omega} \geq \bar{c} + 3\bar{r}/2$ with $\bar{c} \geq \bar{r}/2$. These conditions imply

$$0 < \frac{\left|d_i^2 - \hat{\omega}^2\right|}{|\bar{c}^2 - \hat{\omega}^2| - f_{i,\bar{I}}} \leq \frac{\left|d_i^2 - \hat{\omega}^2\right|}{|\bar{c}^2 - \hat{\omega}^2| - (\bar{c} + \bar{c}/2)^2 + \bar{c}^2} \leq 4\,.$$

Thus

$$|\mathbf{fl}\left(\mathcal{M}_{\bar{I}}(\hat{\omega})\right) - \mathcal{M}_{\bar{I}}(\hat{\omega})| \leq 4\left((1+\varrho)^{\mathbf{N}_{\mathbf{lev}}} - 1\right) \sum_{d_i \in \bar{I}} \frac{|x_i|}{|d_i^2 - \hat{\omega}^2|} \approx 4\mathbf{N}_{\mathbf{lev}}\, \varrho \sum_{d_i \in \bar{I}} \frac{|x_i|}{|d_i^2 - \hat{\omega}^2|}\,,$$

which is (4.2).

4.2. Round-off Errors in the Local Expansions. We first consider the round-off errors in computing the coefficients of

$$\mathcal{L}_I(\omega) = \sum_{j=0}^{p-1} a_j \left(\omega^2 - c^2\right)^j \approx \sum_{d_i \notin \aleph(I)} \frac{x_i}{d_i^2 - \omega^2} \tag{4.16}$$

by bounding the round-off errors in the coefficients of local expansions obtained by (see Procedure **Local**(I))
expanding $\sum_{d_i \in \tilde{I}} x_i/(d_i^2 - \omega^2)$ for a coarser colleague $\tilde{I}$ of I;
transforming the multipole expansion of a colleague of I on the same mesh level;
shifting the center of the local expansion of the parent of I.
We then consider the round-off errors in evaluating the local expansions.

First assume that singularities not in $\aleph(I)$ are those in $\tilde{I}$, where $\tilde{I}$ is a coarser colleague of I. In this case $\{a_j\}_{j=1}^{p-1}$ are computed as in (**??**) (see Procedure **Local**(I)):

$$a_j = \sum_{d_i \in \tilde{I}} \frac{x_i}{\left(d_i^2 - c^2\right)^{j+1}}\,.$$

Since each difference $d_i^2 - c^2$ is computed to high relative accuracy (see Section 3.5), each term in the sum of a_j is computed to high relative accuracy. Similar to (4.6) we have

$$|\mathbf{fl}(a_j) - a_j| \leq \varrho \sum_{d_i \in \tilde{I}} \frac{|x_i|}{\left|d_i^2 - c^2\right|^{j+1}}\,. \tag{4.17}$$

We also have

$$|a_j| \leq \sum_{d_i \in \tilde{I}} \frac{|x_i|}{\left|d_i^2 - c^2\right|^{j+1}}\,.$$

Next assume that singularities not in $\aleph(I)$ are those in $\tilde{I}$, where $\tilde{I}$ is a colleague of I on the same mesh level. In this case $\{a_j\}_{j=1}^{p-1}$ are computed as in Lemma **??** (see Procedure **Local**(I)):

$$a_j = \sum_{k=0}^{p-1-j} \binom{j+k}{k} \frac{\tilde{b}_k}{(\tilde{c}^2 - c^2)^{k+j+1}} \quad \text{with} \quad \tilde{b}_k = \sum_{d_i \in \tilde{I}} x_i \left(d_i^2 - \tilde{c}^2\right)^k .$$

Since $\tilde{c}^2 - c^2$ is computed to high relative accuracy (see Section 3.5), each term in a_j is computed to high relative accuracy. Similar to (4.6) we have

$$\begin{aligned} |\mathbf{fl}(a_j) - a_j| \leq \quad & (1+\varrho) \sum_{k=0}^{p-1-j} \binom{j+k}{k} \frac{\left|\mathbf{fl}(\tilde{b}_k) - \tilde{b}_k\right|}{|\tilde{c}^2 - c^2|^{k+j+1}} \\ & + \varrho \sum_{k=0}^{p-1-j} \binom{j+k}{k} \frac{|\tilde{b}_k|}{|\tilde{c}^2 - c^2|^{k+j+1}} . \end{aligned}$$

According to Section 4.1, we can write the round-off errors in $\{\tilde{b}_k\}_{k=0}^{p-1}$ as

$$\left|\mathbf{fl}(\tilde{b}_k) - \tilde{b}_k\right| \leq \left((1+\varrho)^{\mathrm{N_{lev}}-1} - 1\right) \sum_{d_i \in \tilde{I}} |x_i| \, f_{i,\tilde{I}}^k ,$$

where $f_{i,\tilde{I}}$ satisfies (4.15). We also have

$$|\tilde{b}_k| \leq \sum_{d_i \in \tilde{I}} |x_i| \left|d_i^2 - \tilde{c}^2\right|^k \leq \sum_{d_i \in \tilde{I}} |x_i| f_{i,\tilde{I}}^k .$$

These relations imply

$$\begin{aligned} |\mathbf{fl}(a_j) - a_j| &\leq \left((1+\varrho)^{\mathrm{N_{lev}}} - 1\right) \sum_{d_i \in \tilde{I}} |x_i| \sum_{k=0}^{\infty} \binom{j+k}{k} \frac{f_{i,\tilde{I}}^k}{|\tilde{c}^2 - c^2|^{k+j+1}} \\ &\leq \left((1+\varrho)^{\mathrm{N_{lev}}} - 1\right) \sum_{d_i \in \tilde{I}} \frac{|x_i|}{\left(|\tilde{c}^2 - c^2| - f_{i,\tilde{I}}\right)^{j+1}} , \qquad (4.18) \end{aligned}$$

where we have used the fact that $|\tilde{c}^2 - c^2| > f_{i,\tilde{I}}$. Similarly we have

$$|a_j| \leq \sum_{d_i \in \tilde{I}} \frac{|x_i|}{\left(|\tilde{c}^2 - c^2| - f_{i,\tilde{I}}\right)^{j+1}} .$$

Now assume that singularities not in $\aleph(I)$ are those not in $\aleph(\bar{I})$, where $\bar{I}$ is the parent of I. In this case $\{a_j\}_{j=1}^{p-1}$ are computed as in Lemma **??** (see Procedure **Local**(I)):

$$a_j = \sum_{k=j}^{p-1} \bar{a}_k \binom{k}{j} \left(\bar{c}^2 - c^2\right)^{k-j} ,$$

where $\{\bar{a}_j\}_{j=1}^{p-1}$ are the coefficients of $\mathcal{L}_{\bar{I}}(\omega)$.

Since the difference $\bar{c}^2 - c^2$ is computed to high relative accuracy (see Section 3.6), each term in the sum in a_j is also computed to high relative accuracy. Similar to (4.10) we have

$$\begin{aligned} |\mathbf{fl}(a_j) - a_j| \leq \quad & (1+\varrho)\sum_{k=j}^{p-1} |\mathbf{fl}(\bar{a}_k) - \bar{a}_k| \left(\begin{array}{c} k \\ j \end{array} \right) \left|\bar{c}^2 - c^2\right|^{k-j} \\ & + \varrho \sum_{k=j}^{p-1} |\bar{a}_k| \left(\begin{array}{c} k \\ j \end{array} \right) \left|\bar{c}^2 - c^2\right|^{k-j} . \end{aligned} \tag{4.19}$$

Similar to Section 4.1, we determine the rounding errors in $\{a_j\}_{j=0}^{p-1}$ by induction. In light of (4.17), (4.18) and the corresponding bounds for $\{|a_j|\}_{j=0}^{p-1}$, we assume that the rounding errors in $\{a_j\}_{j=0}^{p-1}$ and $\{\bar{a}_j\}_{j=0}^{p-1}$ have the form

$$|\mathbf{fl}(\bar{a}_j) - \bar{a}_j| \leq \sum_{d_i \notin \aleph(\bar{I})} \frac{|x_i|\, \vartheta_{i,\bar{I}}}{g_{i,\bar{I}}^{j+1}} \quad \text{and} \quad |\mathbf{fl}(a_j) - a_j| \leq \sum_{d_i \notin \aleph(I)} \frac{|x_i|\, \vartheta_{i,I}}{g_{i,I}^{j+1}} . \tag{4.20}$$

We also assume that

$$|\bar{a}_k| \leq \sum_{d_i \notin \aleph(\bar{I})} \frac{|x_i|}{g_{i,\bar{I}}^{j+1}} \quad \text{and} \quad |a_k| \leq \sum_{d_i \notin \aleph(I)} \frac{|x_i|}{g_{i,I}^{j+1}} . \tag{4.21}$$

Plugging these relations into (4.19), we have

$$\begin{aligned} |\mathbf{fl}(a_j) - a_j| \leq \quad & (1+\varrho) \sum_{d_i \notin \aleph(\bar{I})} |x_i|\, \vartheta_{i,\bar{I}} \sum_{k=j}^{\infty} \left(\begin{array}{c} k \\ j \end{array} \right) \frac{\left|\bar{c}^2 - c^2\right|^{k-j}}{g_{i,\bar{I}}^{k+1}} \\ & + \varrho \sum_{d_i \notin \aleph(\bar{I})} |x_i| \sum_{k=j}^{\infty} \left(\begin{array}{c} k \\ j \end{array} \right) \frac{\left|\bar{c}^2 - c^2\right|^{k-j}}{g_{i,\bar{I}}^{k+1}} \\ \leq & \sum_{d_i \notin \aleph(\bar{I})} \frac{|x_i| \left((1+\varrho)\vartheta_{i,\bar{I}} + \varrho\right)}{\left(g_{i,\bar{I}} - |\bar{c}^2 - c^2|\right)^{j+1}} , \end{aligned} \tag{4.22}$$

provided that $g_{i,\bar{I}} > \left|\bar{c}^2 - c^2\right|$. Similiarly

$$\begin{aligned} |a_j| & \leq \sum_{k=j}^{p-1} |\bar{a}_k| \left(\begin{array}{c} k \\ j \end{array} \right) \left|\bar{c}^2 - c^2\right|^{k-j} \\ & \leq \sum_{d_i \notin \aleph(\bar{I})} \frac{|x_i|}{\left(g_{i,\bar{I}} - |\bar{c}^2 - c^2|\right)^{j+1}} . \end{aligned}$$

Comparing this with (4.20), (4.21) and (4.22), we can set

$$\vartheta_{i,I} = (1+\varrho)\vartheta_{i,\bar{I}} + \varrho \quad \text{and} \quad g_{i,I} = g_{i,\bar{I}} - \left|\bar{c}^2 - c^2\right| , \tag{4.23}$$

provided that $g_{i,I} > 0$. Similar to the recursion (4.12) for the multipole expansions, recursion (4.23) holds for any parent interval $\bar{I}$ and its child I.

Now we solve the recursion (4.23). For any $d_i \notin \aleph(I)$, let $I_1, I_2, \ldots, I_k \equiv I$ be intervals on the computation tree such that I_i is the parent of I_{i+1} for $1 \leq i \leq k-1$.

Also let $\tilde{I}_1$ be the colleague of I_1 such that $d_i \in \tilde{I}$. Let c_j and r_j be the center and the length of I_j, respectively.

If $\tilde{I}_1$ is a coarser colleague of I_1, then relations (4.17) and (4.23) imply that

$$g_{i,I_1} = \left|d_i^2 - c_1^2\right| \quad \text{and} \quad g_{i,I_{j+1}} = g_{i,I_j} - \left|c_j^2 - c_{j+1}^2\right| \ ,$$

and

$$\vartheta_{i,I_1} = \varrho \quad \text{and} \quad \vartheta_{i,I_{j+1}} = (1+\varrho)\vartheta_{i,I_j} + \varrho \ .$$

Solving this recursion we have

$$g_{i,I} = \left|d_i^2 - c_1^2\right| - \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \quad \text{and} \quad \vartheta_{i,I} = (1+\varrho)^k - 1 \ . \tag{4.24}$$

If $\tilde{I}_1$ is a colleague of I_1 on the same mesh level, then relations (4.17) and (4.23) imply that

$$g_{i,I_1} = \left|\tilde{c}_1^2 - c_1^2\right| - f_{i,\tilde{I}_1} \quad \text{and} \quad g_{i,I_{j+1}} = g_{i,I_j} - \left|c_j^2 - c_{j+1}^2\right| \ ,$$

and

$$\vartheta_{i,I_1} \le (1+\varrho)^{\mathbf{N_{lev}}} - 1 \quad \text{and} \quad \vartheta_{i,I_{j+1}} = (1+\varrho)\vartheta_{i,I_j} + \varrho \ .$$

Solving this recursion we have

$$g_{i,I} = \left|\tilde{c}_1^2 - c_1^2\right| - f_{i,\tilde{I}_1} - \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \quad \text{and} \quad \vartheta_{i,I} \le (1+\varrho)^{k+\mathbf{N_{lev}}-1} - 1 \ . \tag{4.25}$$

To complete the induction, we need to show that $g_{i,I} > 0$. We discuss this when we bound the round-off errors in evaluating the local expansions.

Since local expansions are computed only for $\mathbf{N_{lev}} - 1$ mesh levels, we have $k \le \mathbf{N_{lev}} - 1$. Thus relations (4.24) and (4.25) imply

$$\vartheta_{i,I} \le (1+\varrho)^{2\mathbf{N_{lev}}-1} - 1 \tag{4.26}$$

for any singularity d_i and any interval I.

Finally we consider the round-off errors in evaluating $\mathcal{L}_I(\omega)$ of (4.16) at the point $\hat{\omega} inI$, where $\hat{\omega}$ is one of $\{\hat{\omega}_j\}_{j=1}^n$ (see (3.2)).

Since the difference $\hat{\omega}^2 - c^2$ is computed to high relative accuracy (see Section 3.6), each term in the sum in (4.16) is also computed to high relative accuracy. Similar to (4.10) we have

$$\left|\mathbf{fl}\left(\mathcal{L}_I(\hat{\omega})\right) - \mathcal{L}_I(\hat{\omega})\right| \le (1+\varrho)\sum_{j=0}^{p-1} \left|\mathbf{fl}(a_j) - a_j\right| \left|\hat{\omega} - c^2\right|^j + \varrho \sum_{j=0}^{p-1} \left|a_j\right| \left|\hat{\omega}^2 - c^2\right|^j \ .$$

By (4.20), (4.21) and (4.26) , this implies

$$\left|\mathbf{fl}\left(\mathcal{L}_I(\hat{\omega})\right) - \mathcal{L}_I(\hat{\omega})\right| \le \left((1+\varrho)^{2\mathbf{N_{lev}}} - 1\right) \sum_{d_i \notin \aleph(I)} |x_i| \sum_{j=0}^{\infty} \frac{\left|\hat{\omega}^2 - c^2\right|^j}{g_{i,I}^{j+1}} \tag{4.27}$$

$$\le \left((1+\varrho)^{2\mathbf{N_{lev}}} - 1\right) \sum_{d_i \notin \aleph(I)} \frac{|x_i|}{g_{i,I} - \left|\hat{\omega}^2 - c^2\right|} \ . \tag{4.28}$$

provided that $g_{i,I} > \left|\hat{\omega}^2 - c^2\right|$. This assumption implies $g_{i,I} > 0$ and hence the completion of the induction.

By Lemma 4.1 we have

$$\left|\hat{\omega}^2 - c^2\right| + \sum_{j=1}^{k-1} \left|c_j^2 - c_{j+1}^2\right| \le \left(c_1 + \frac{r_1}{2}\right)^2 - c_1^2 \, . \tag{4.29}$$

In the following we bound the ratio $\left|d_i^2 - \hat{\omega}^2\right| / \left(g_{i,I_1} - \left|\hat{\omega}^2 - c_k^2\right|\right)$. For any $d_i \not\in \aleph(I)$, let $I_1, I_2, \ldots, I_k \equiv I$ be intervals on the computation tree such that I_i is the parent of I_{i+1} for $1 \le i \le k-1$. Also let $\tilde{I}_1$ be the colleague of I_1 such that $d_i \in \tilde{I}$. Let c_j and r_j be the center and the length of I_j, respectively. Since $\hat{\omega} \in I_1$ we have $|\hat{\omega} - c_1| \le r_1/2$. Since $\tilde{I}_1$ is a colleague of I_1, according to Lemma 3.3 we have either

$$d_i \ge c_1 + \frac{3r_1}{2} \quad \text{or} \quad d_i \le c_1 - \frac{3r_1}{2} \quad , \quad c_1 \ge \frac{5r_1}{2} \, .$$

If $\tilde{I}_1$ is a coarser colleague of I_1, then applying (4.29) to relation (4.24), we have

$$g_{i,I_1} - \left|\hat{\omega}^2 - c_k^2\right| \ge \left|d_i^2 - c_1^2\right| - \left(c_1 + \frac{r_1}{2}\right)^2 + c_1^2 \, ,$$

which implies

$$0 < \frac{\left|d_i^2 - \hat{\omega}^2\right|}{g_{i,I_1} - \left|\hat{\omega}^2 - c_k^2\right|} \le 4 \, .$$

If $\tilde{I}_1$ is a colleague of I_1 on the same level, then applying (4.29) and (4.15) to relation (4.25), we have

$$g_{i,I_1} - \left|\hat{\omega}^2 - c_k^2\right| \ge \left|\tilde{c}_1^2 - c_1^2\right| - \left(c_1 + \frac{r_1}{2}\right)^2 + c_1^2 - \left(\tilde{c}_1 + \frac{r_1}{2}\right)^2 + \tilde{c}_1^2 \, ,$$

Since $|\tilde{c}_1 - c_1| \ge 2r_1$, this relation implies

$$0 < \frac{\left|d_i^2 - \hat{\omega}^2\right|}{g_{i,I_1} - \left|\hat{\omega}^2 - c_k^2\right|} \le 4 \, .$$

Plugging these relations into (4.28) we have

$$\left|\mathbf{fl}\left(\mathcal{L}_I(\hat{\omega})\right) - \mathcal{L}_I(\hat{\omega})\right| \le 4\left((1+\varrho)^{2\mathbf{N}_{\mathbf{lev}}} - 1\right) \sum_{d_i \not\in \aleph(I)} \frac{|x_i|}{\left|d_i^2 - \hat{\omega}^2\right|} \approx 8\mathbf{N}_{\mathbf{lev}}\, \varrho \sum_{d_i \not\in \aleph(I)} \frac{|x_i|}{\left|d_i^2 - \hat{\omega}^2\right|} \, ,$$

which is (4.3).

4.3. Numerical Stability in Singular Vector Computations. As mentioned in Sections 1 and 3, we use the modified **FMM** to accelerate the computation of the singular vectors of A'. With the singular vector matrices of M or M_1, we compute the singular vector matrices of $A' \in \mathbf{R}^{(m+1)\times n}$. As in Section 3, we only consider the problem of computing a numerical right singular vector matrix $V\hat{Q}$ of A' for the case $m \ge n$, where V and $\hat{Q}$ are orthonormal.

Let $v^T = (v_1, \ldots, v_n)^T$ be a row of V. Then $\|v\|_2 = 1$ and the corresponding row of $V\hat{Q}$ is $v^T\hat{Q} = (v^T\hat{q}_1, \ldots, v^T\hat{q}_n)$ with $v^T\hat{q}_j = \Phi_1(\omega_j) \Big/ \sqrt{\Phi_2(\omega_j)}$, where (see Section 3)

$$\Phi_1(\omega) = \sum_{i=1}^{n} \frac{v_i z_i}{d_i^2 - \omega^2} \quad \text{and} \quad \Phi_2(\omega) = \sum_{i=1}^{n} \frac{z_i^2}{\left(d_i^2 - \omega^2\right)^2} \,.$$

We compute $\Phi_1(\omega_j)$ using the modified **FMM** with a precision satisfying (see (4.1))

$$\mathbf{fl_f}\left(\Phi_1(\omega_j)\right) = \Phi_1(\omega_j) + O\left(\epsilon \sum_{i=1}^{n} \frac{|v_i z_i|}{|d_i^2 - \omega_j^2|}\right) \,. \tag{4.30}$$

As in Section 2.1, we directly compute $\sqrt{\Phi_2(\omega_j)}$ to high relative accuracy

$$\mathbf{fl}\left(\sqrt{\Phi_2(\omega_j)}\right) = \sqrt{\Phi_2(\omega_j)}\,\left(1 + O(\epsilon)\right) \,. \tag{4.31}$$

By the Cauchy-Schwartz inequality, we get

$$\begin{aligned} \sum_{i=1}^{n} \frac{|v_i z_i|}{|d_i^2 - \omega_j^2|} &\leq \|v\|_2 \sqrt{\sum_{i=1}^{n} \frac{|z_i^2|}{\left(d_i^2 - \omega_j^2\right)^2}} \\ &= \sqrt{\Phi_2(\omega_j)} \,. \end{aligned}$$

Plugging this into (4.30) and using (4.31) we have

$$\begin{aligned} \mathbf{fl_f}\left(\frac{\Phi_1(\omega_j)}{\sqrt{\Phi_2(\omega_j)}}\right) &= \left(\frac{\Phi_1(\omega_j)}{\sqrt{\Phi_2(\omega_j)}}\right)\left(1 + O(\epsilon)\right) + O(\epsilon) \\ &= \frac{\Phi_1(\omega_j)}{\sqrt{\Phi_2(\omega_j)}} + O(\epsilon) \,, \end{aligned}$$

where we have used the fact that

$$\left| \Phi_1(\omega_j) \Big/ \sqrt{\Phi_2(\omega_j)} \right| \leq 1 \,.$$

Thus, each component of $v^T\hat{Q}$ is computed to high absolute accuracy. When all the rows of $V\hat{Q}$ are computed, the resulting matrix

$$\mathbf{fl_f}(V\hat{Q}) = V\hat{Q} + O(\epsilon)$$

is numerically orthonormal.

REFERENCES

[1] J. R. Bunch and C. P. Nielsen, *Updating the singular value decomposition*, Numer. Math., 31 (1978), pp. 111–129.

[2] J. R. Bunch, C. P. Nielsen, and D. C. Sorensen, *Rank-one modification of the symmetric eigenproblem*, Numer. Math., 31 (1978), pp. 31–48.

[3] J. Carrier, L. Greengard, and V. Rokhlin, *A fast adaptive multipole algorithm for particle simulations*, SIAM J. Sci. Stat. Comput., 9 (1988), pp. 669–686.

[4] G. H. Golub, *Some modified matrix eigenvalue problems*, SIAM Review, 15 (1973), pp. 318–334.

[5] G. H. Golub and C. F. van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, second ed., 1989.

[6] W. B. Gragg, J. R. Thornton, and D. D. Warner, *Parallel divide and conquer algorithms for the symmetric tridiagonal eigenproblem and bidiagonal singular value problem*, in Proceedings of 23rd Annual Pittsburgh Conference, University of Pittsburgh School of Engineering, vol. 23 of Modelling and Simulation, 1992.

[7] L. Greengard and V. Rokhlin, *A fast algorithm for particle simulations*, J. Comp. Phys., 73 (1987), pp. 325–348.

[8] M. Gu, *Numerical Linear Algebra Computations*, PhD thesis, Department of Computer Science, Yale University, November 1993.

[9] M. Gu and S. C. Eisenstat, *A divide-and-conquer algorithm for the bidiagonal SVD*, Research Report YALEU/DCS/RR-933, Department of Computer Science, Yale University, December 1992. To appear in SIMAX.

[10] ———, *A stable and efficient algorithm for the rank-one modification of the symmetric eigenproblem*, Research Report YALEU/DCS/RR-916, Department of Computer Science, Yale University, September 1992.

[11] ———, *Downdating the singular value decomposition*, Research Report YALEU/DCS/RR-939, Department of Computer Science, Yale University, May 1993.

[12] E. R. Jessup and D. C. Sorensen, *A parallel algorithm for computing the singular value decomposition of a matrix.* Revision of Tech. Report ANL-MCS-TM-102, Argonne National Laboratory, 1991.

[13] R.-C. Li, *Solving secular equations stably and efficiently.* Unpublished manuscript, October 1992.

[14] M. Moonen, P. van Dooren, and J. Vandewalle, *A singular value decomposition updating algorithm for subspace tracking*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 1015–1038.

[15] M. Moonen, P. van Dooren, and F. Vanpouche, *On the QR algorithm and updating the SVD and URV decomposition in parallel.* IMA preprint, 1992.

[16] G. W. Stewart, *Introduction to Matrix Computations*, Academic Press, New York, 1973.

[17] ———, *An updating algorithm for subspace tracking*, Tech. Report CS TR–2494, Department of Computer Science, University of Maryland, July 1990.

[18] ———, *Updating a rank-revealing ULV decomposition*, Tech. Report CS TR–2627, Department of Computer Science, University of Maryland, March 1991.

[19] ———, *Determining rank in the presence of error*, Tech. Report UMIACS TR–92–108 and CS TR–2972, Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, October 1992.