On Sliding Window Schemes For Discrete Least-Squares Approximation By Trigonometric Polynomials

Heike Fassbender

Universität Bremen, Fachbereich 3 Mathematik und Informatik, Zentrum für Technomathematik, 28334 Bremen, Germany

Abstract

Fast, efficient, and reliable algorithms for up- and downdating discrete least-squares approximations of a real-valued function given at arbitrary distinct nodes in $[0, 2\pi)$ by trigonometric polynomials are presented. A combination of the up- and downdating algorithms yields a sliding window scheme. The algorithms are based on schemes for the solution of (inverse) unitary eigenproblems and require only $\mathcal{O}(mn)$ arithmetic operations as compared to $\mathcal{O}(mn^2)$ operations needed for algorithms that ignore the structure of the problem. Numerical examples show that the proposed algorithms produce consistently accurate results that are often better than those obtained by general QR decomposition methods for the least-squares problem.

Key words. trigonometric approximation, unitary Hessenberg matrix, Schur parameter, Szegö polynomial, updating, downdating, sliding window scheme

1 Introduction

A problem in signal processing is the approximation of a function known only at some measured points by a trigonometric polynomial. A number of different models for representing the measured points as a finite superposition of sine- and cosine-oscillations are possible. One choice could be to compute the trigonometric interpolating function. Then several numerical algorithms are available [1]. But in general a large number of measured points are given, such that this approach leads to a trigonometric polynomial with a lot of superposed oscillations (and a large linear system to be solved). In practical applications it is often sufficient to compute a trigonometric polynomial with only a small number of superposed oscillations. A different, often chosen approach is the (fast) Fourier transform [1]. In this case the frequencies of the sine- and cosine-oscillations have to be chosen equidistant. The following approach gives more freedom in the choice of the frequencies and the number of superposed oscillations. Given a

set of m arbitrary distinct nodes $\{\theta_k\}_{k=1}^m$ in the interval $[0, 2\pi)$, a set of m positive weights $\{\omega_k^2\}_{k=1}^m$, and a real-valued function f whose values at the nodes θ_k are explicitly known. Then the trigonometric function

$$t(\theta) = a_0 + \sum_{j=1}^{\ell}(a_j \cos j\theta + b_j \sin j\theta), \qquad a_j, b_j \in \mathbb{R}, \tag{1.1}$$

of order at most $\ell < m/2$ is sought that minimizes the discrete least-squares error

$$||f - t||_{\mathbb{R}} := \sqrt{\sum_{k=1}^{m} |f(\theta_k) - t(\theta_k)|^2 \omega_k^2}. \tag{1.2}$$

In general, m (the number of measured functional values) is much larger than $n = 2\ell + 1$ (the number of coefficients to be determined).

Let the trigonometric polynomial $t(\theta) = a_0 + \sum_{j=1}^{\ell}(a_j \cos j\theta + b_j \sin j\theta)$ be the optimal solution of the approximation problem (1.2) corresponding to the data $Y_m = \{\theta_k, \omega_k^2\}_{k=1}^m$. Suppose Y_{m+1} is obtained from Y_m by augmenting a new node-weight pair $(\theta_{m+1}, \omega_{m+1}^2)$. Solving the approximation problem for Y_{m+1} assuming the knowledge of its solution for Y_m is called *updating* the least-squares fit. Solving the approximation problem for Y_m assuming the knowledge of its solution for Y_{m+1} is called *downdating* the least-squares fit.

Standard algorithms for solving the approximation problem (1.2) require $\mathcal{O}(mn^2)$ arithmetic operations. It can be observed however, that Szegö polynomials, that is polynomials that are orthogonal with respect to an inner product on the unit circle, arise naturally as a convenient basis for solving the least-squares problem (1.2). Updating and downdating of polynomial approximations when all nodes z_k are real has received a lot of attention in the literature, see [2] and the references therein. A collection of algorithms for updating and downdating based on orthogonal polynomials is presented in [3]. Downdating Szegö polynomials is considered in [4], while the updating process is the topic of [5, 6, 7].

Section 2 reviews fast algorithms for solving (1.2) via updating procedures which make use of the special structure of the problem (1.2). Fast downdating methods are presented in Section 3. The updating and downdating procedures can be combined to yield a sliding window scheme, in which one node is replaced by another.

2 Updating

The problem (1.2) can easily be reformulated as the standard least-squares problem of minimizing

$$||D\widetilde{A}\widetilde{t} - D\widetilde{f}||_2 = \min \tag{2.1}$$

over all coefficient vectors $\widetilde{t}$ in the Euclidean norm, where $\widetilde{f} = (f(\theta_1), \ldots, f(\theta_m))^T$ is a vector of the measured values of the function f, $D = diag(\omega_1, ..., \omega_m) \in$

$\mathbb{R}^{m\times m}$ is a diagonal matrix with the given weights on the diagonal, and

$$\widetilde{A} = \begin{pmatrix} 1 & \sin\theta_1 & \cos\theta_1 & \cdots & \sin\ell\theta_1 & \cos\ell\theta_1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & \sin\theta_m & \cos\theta_m & \cdots & \sin\ell\theta_m & \cos\ell\theta_m \end{pmatrix} \in \mathbb{R}^{m\times n}.$$

A different approach is used by Reichel, Ammar, and Gragg in [5]. They noted that the problem (1.2) can be reformulated as the following standard least-squares problem: Minimize

$$||DAc - Dg||_2 = \min, \tag{2.2}$$

where A is a transposed Vandermonde matrix

$$A = \begin{pmatrix} 1 & z_1 & \cdots & z_1^{n-1} \\ 1 & z_2 & \cdots & z_2^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & z_m & \cdots & z_m^{n-1} \end{pmatrix} \in \mathbb{C}^{m\times n}$$

with $z_k = exp(\imath\theta_k), \imath = \sqrt{-1}$. $g = (g(z_1), ..., g(z_m))^T \in \mathbb{C}^m$ is a vector of the values of a complex function $g(z)$ and $c = (c_0, ..., c_{n-1})^T \in \mathbb{C}^n$ is the solution vector. With the proper choice of g ($g = \Lambda^\ell \widetilde{f}$, $\widetilde{f}$ as above, $\Lambda = diag(z_1, \ldots, z_m)$) it is easy to see that the coefficients of the trigonometric polynomial (1.1) that minimizes the error (1.2) can be read off of the least-squares solution $\widehat{c}$ of (2.2) (see [5])

$$a_0 = \widehat{c}_\ell, \qquad a_j = 2Re(\widehat{c}_{j+\ell}), \qquad b_j = -2Im(\widehat{c}_{j+\ell}), \qquad 1 \le j \le \ell.$$

The usual way to solve these least-squares problems is to compute the QR decomposition of DA or $D\widetilde{A}$. Ignoring the special structure of DA or $D\widetilde{A}$ this requires $\mathcal{O}(mn^2)$ arithmetic operations. It can be observed however, that Szegö polynomials, that is polynomials that are orthogonal with respect to an inner product on the unit circle, arise naturally as a convenient basis for solving the above standard least-squares problems. This observation can be used to develop fast, efficient and reliable algorithms for solving the approximation problem (2.2).

Observe that

$$\begin{aligned} DA &= \begin{pmatrix} \omega_1 & \omega_1 z_1 & \omega_1 z_1^2 & \cdots & \omega_1 z_1^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ \omega_m & \omega_m z_m & \omega_m z_m^2 & \cdots & \omega_m z_m^{n-1} \end{pmatrix} \\ &= (q, \Lambda q, \Lambda^2 q, ..., \Lambda^{n-1} q) \\ &= \sigma_0(q_0, \Lambda q_0, \Lambda^2 q_0, ..., \Lambda^{n-1} q_0) \end{aligned}$$

with $q = (\omega_1, ..., \omega_m)^T$, $\sigma_0 = ||q||_2$, $q_0 := \sigma_0^{-1} q$ and $\Lambda = diag(z_1, ..., z_m)$. Thus, the matrix DA is given by the first n columns of the Krylov matrix $K(\Lambda, q_0, m) =$

$(q_0, \Lambda q_0, ..., \Lambda^{m-1} q_0)$. We may therefore use the following consequence of the Implicit Q Theorem [8] to compute the desired QR decomposition. If there exists a unitary matrix U with $Ue_1 = q_0$ such that $U^H \Lambda U = H$ is a unitary upper Hessenberg matrix with positive subdiagonal elements, then the QR decomposition of $K(\Lambda, q_0, m)$ is given by UR with $R = K(H, e_1, m)$. The construction of such a unitary Hessenberg matrix from spectral data, here contained in Λ and q_0, is an inverse eigenproblem. Hence the best trigonometric approximation to f can be computed via solving this inverse eigenproblem. It can be seen that the elements of U are the values of the Szegö polynomials at the nodes z_k. Thus solving the inverse unitary Hessenberg eigenvalue problem $U^H \Lambda U = H$ is equivalent to computing Szegö polynomials.

From the above observation, an updating formulation for the approximation problem (2.2) in terms of the inverse unitary eigenproblem can easily be given :

Given

$\sigma_0 > 0$	
H_m	unitary upper Hessenberg matrix of size $m \times m$
d_m	a vector of length m
(λ, ν^2)	a node-weight pair

$(\sigma_0, H_m, d_m$ representing the solution of (2.2) for some data set $Y_m)$ find $\sigma_0 > 0$, a vector d_{m+1} and a unitary upper Hessenberg matrix H_{m+1} such that

1. the eigenvalues of H_{m+1} are $e^{i\lambda}$ and those of H_m
2. the vector d_{m+1} contains the first components of the eigenvectors of H_{m+1}, that is if the entries of d_m are $\delta_1/\sigma_0, \ldots, \delta_m/\sigma_0$ and $\sigma_0 = (\sum_{k=1}^{m} \delta_k^2)^{\frac{1}{2}}$, then the new σ_0 will be $\sigma_0 = (\sigma_0^2 + \nu^2)^{\frac{1}{2}}$ and the entries of d_{m+1} are δ_k/σ_0 for $k = 1, \ldots, m$ and ν/σ_0.

Hence the approximation problem (1.2) can entirely be solved by updating, starting from the trivial solution for $m = 1$.

Unitary Hessenberg matrices have special properties which allow the development of efficient algorithms for this class of matrices. Any $n \times n$ unitary Hessenberg matrix with positive subdiagonal elements can be uniquely parameterized by n complex parameters, that is

$$H = G_1(\gamma_1) G_2(\gamma_2) \cdots G_n(\gamma_n)$$

for certain complex-valued parameters $|\gamma_k| < 1, 1 \leq k < n$, and $|\gamma_n| = 1$. Here $G_k(\gamma_k)$ denotes the $n \times n$ elementary reflector in the $(k, k+1)$ plane

$$G_k = G_k(\gamma_k) = diag(I_{k-1}, \begin{pmatrix} -\gamma_k & \sigma_k \\ \sigma_k & \overline{\gamma_k} \end{pmatrix}, I_{n-k-1})$$

with $\gamma_k \in \mathbb{C}$, $\sigma_k \in \mathbb{R}^+$, $|\gamma_k|^2 + \sigma_k^2 = 1$, and

$$G_n(\gamma_n) = diag(I_{n-1}, -\gamma_n)$$

with $\gamma_n \in \mathbb{C}$, $|\gamma_n| = 1$. The nontrivial entries γ_k are called *Schur parameters* and the σ_k are called *complementary Schur parameters*. This parameterization can be used to develop an efficient and reliable algorithm for solving the inverse unitary Hessenberg eigenvalue problem/the updating problem.

Such an algorithm was first described by Ammar, Gragg and Reichel in [6]. The idea is to build up the Hessenberg matrix successively by adding node-weight pairs (θ_k, ω_k^2) one at a time, in an updating fashion. The algorithm manipulates the n complex parameters instead of the n^2 matrix elements. An adaption of this scheme to the computation of the least-squares solution $\widehat{c}$ can be given, which requires $\mathcal{O}(mn + n^2)$ arithmetic operations. For details see [5].

The coefficients of the optimal trigonometric polynomial t of (1.2) can be recovered from $\widehat{c}$. This representation of t is convenient if we desire to integrate or differentiate the polynomial or if we wish to evaluate it at many equidistant points on a circle with a center at the origin. If we, on the other hand, only desire to evaluate t at a few points, then we can use the representation of t in terms of Szegö polynomials.

As $D\widetilde{A}$ in (2.1) is a real $m \times n$ matrix with full column rank, there exists a unique "skinny" real QR decomposition $\widetilde{Q}_1\widetilde{R}_1$ of $D\widetilde{A}$ where $\widetilde{Q}_1 \in \mathbb{R}^{m\times n}$ has orthonormal columns and $\widetilde{R}_1 \in \mathbb{R}^{n\times n}$ is upper triangular with positive diagonal entries. This ansatz leads to orthogonal Laurent polynomials and the (generalized) inverse unitary eigenproblem $\widetilde{U}^H(\Lambda - \lambda I)\widetilde{U}G_e = G_o - \lambda G_e$, where G_o and G_e are unitary block diagonal matrices with 1×1 or 2×2-blocks on the diagonal. The nonzero entries of G_o and G_e are just the Schur parameters and the complementary Schur parameters:

$$\begin{aligned} G_o &= G_1(\gamma_1)G_3(\gamma_3)\cdots G_{2[(n+1)/2]-1}(\gamma_{2[(n+1)/2]-1}) \\ &= \begin{pmatrix} -\gamma_1 & \sigma_1 & & & \\ \sigma_1 & \overline{\gamma_1} & & & \\ & & -\gamma_3 & \sigma_3 & \\ & & \sigma_3 & \overline{\gamma_3} & \\ & & & & \ddots \end{pmatrix} \end{aligned}$$

is the product of the odd numbered elementary reflectors and

$$\begin{aligned} G_e^H &= G_2(\gamma_2)G_4(\gamma_4)\cdots G_{2[n/2]}(\gamma_{2[n/2]}) \\ &= \begin{pmatrix} 1 & & & \\ & -\gamma_2 & \sigma_2 & \\ & \sigma_2 & \overline{\gamma_2} & \\ & & & \ddots \end{pmatrix} \end{aligned}$$

is the product of the even numbered elementary reflectors. The (generalized) inverse eigenproblem $\widetilde{U}^H(\Lambda - \lambda I)\widetilde{U}G_e = G_o - \lambda G_e$, where a *Schur parameter pencil* is constructed from spectral data, is equivalent to the inverse unitary Hessenberg eigenproblem $U^H\Lambda U = H = G_1(\gamma_1)\ldots G_n(\gamma_n)$ [9].

Observe that with

$$F = diag(2, \begin{pmatrix} -\imath & 1 \\ \imath & 1 \end{pmatrix}, \ldots, \begin{pmatrix} -\imath & 1 \\ \imath & 1 \end{pmatrix})$$

we have

$$\begin{aligned} D\widetilde{A} &= \frac{1}{2}(q, \Lambda q, \Lambda^H q, \Lambda^2 q, (\Lambda^H)^2 q, \ldots, \Lambda^\ell q, (\Lambda^H)^\ell q)F \\ &= \frac{1}{2}\sigma_0(q_0, \Lambda q_0, \Lambda^H q_0, \Lambda^2 q_0, (\Lambda^H)^2 q_0, \ldots, \Lambda^\ell q_0, (\Lambda^H)^\ell q_0)F \\ &= \frac{1}{2}\sigma_0 \kappa(\Lambda, q_0, \ell)F \end{aligned}$$

with q, σ_0, q_0 and Λ as before. A QR-like decomposition of $D\widetilde{A}$ can be obtained using the following result [10, 11] : If there exists a unitary matrix V such that $V(\Lambda - \lambda I)V^H G_e = G_o - \lambda G_e, V^H e_1 = q_0$, then the QR decomposition of $\kappa(\Lambda, q_0, \ell)$ is given by VR with $R = \kappa(G_o G_e^H, e_1, \ell)$. Hence $D\widetilde{A} = \frac{\sigma_0}{2} VRF$ and the optimal solution of (2.1) is given by $\widetilde{t} = 2\sigma_0^{-1} F^{-1} R^{-1} V^H D\widetilde{f}$. The construction of such a Schur parameter pencil from spectral data is a (generalized) inverse eigenproblem. Thus the best trigonometric approximation to f can be computed via solving this inverse eigenproblem. As explained in [7], the elements of V are the values of orthogonal Laurent polynomials at the nodes θ_k. Thus solving the inverse unitary eigenproblem $V(\Lambda - \lambda I)W = G_o - \lambda G_e$ is equivalent to computing orthogonal Laurent polynomials.

An updating formulation for the approximation problem (2.1) in terms of the generalized inverse unitary eigenproblem can be given analogous to the one for problem (2.2). The special structure of the inverse eigenproblem $V(\Lambda - \lambda I)V^H G_e = G_o - \lambda G_e$ can be used to develop an efficient and reliable algorithm for computing V, γ_j and $\sigma_j, j = 1, \ldots, n$. The advantage of this approach over (2.2) is that here an algorithm can be given which solves the real-valued problem (1.2) using only real arithmetic. For details see [11].

The two algorithms sketched above are updating procedures in the sense that the least-squares fit is obtained by incorporating the nodes of the inner product one at a time. In certain applications it may be desirable to replace certain node-weight pairs (θ_k, ω_k^2). This can be carried out by successively removing a node-weight pair from the current approximation, and then adding a new node-weight pair. Downdating Szegö polynomials/orthogonal Laurent polynomials and a given least-squares fit when one node is deleted from the inner product can easily be implemented solving unitary eigenproblems. The updating and downdating procedures, based on solving inverse unitary eigenproblems and unitary eigenproblems, can be combined to yield a sliding window scheme, in which one node is replaced by another.

3 Downdating

Assume the solution t of (1.2) corresponding to the data set Y_{m+1} is obtained by the updating method based on the inverse unitary Hessenberg eigenproblem discussed in the Section 2. The problem of downdating the optimal trigonometric approximation t of (1.2) can then be expressed as follows:

Given

$\sigma_0 > 0$

H_{m+1} unitary upper Hessenberg matrix of size $(m+1) \times (m+1)$

d_{m+1} a vector of length $m+1$

(λ, ν^2) a node-weight pair from Y_{m+1}

($\sigma_0, H_{m+1}, d_{m+1}$ representing the solution of (1.2) for some data set $Y_{m+1} = \{\theta_k, \omega_k^2\}_{k=1}^{m+1}$) find $\sigma_0 > 0$, a vector d_m and a unitary upper Hessenberg matrix H_m such that

1. the eigenvalues of H_m are $\{e^{\imath\theta_k}\}_{k=1}^{m+1} \setminus e^{\imath\lambda}$
2. the vector d_m contains the first components of the eigenvectors of H_m, that is if the entries of d_{m+1} are $\delta_1/\sigma_0, \ldots, \delta_{m+1}/\sigma_0$ and $\sigma_0 = (\sum_{k=1}^{m+1} \delta_k^2)^{\frac{1}{2}}$, then the new σ_0 will be $\sigma_0 = (\sigma_0^2 - \nu^2)^{\frac{1}{2}}$ and the entries of d_m are $\{\delta_k\}_{k=1}^{m+1} \setminus \nu$ normalized by σ_0.

Such a downdating procedure was first described by Ammar, Gragg and Reichel in [4]. They present an $\mathcal{O}(m)$ algorithm which is based on the unitary QR algorithm introduced by Gragg in [12]. Here we will develop a downdating procedure for the generalized unitary eigenproblem approach using these ideas.

Assume the optimal least-squares solution t of (1.2) corresponding to the data $Y_{m+1} = \{\theta_k, \omega_k^2\}_{k=1}^{m+1}$ has been computed via the updating algorithm based on the generalized inverse unitary eigenproblem discussed in Section 2. Then a unitary matrix Q and a Schur parameter pencil $G_o^{m+1} - \lambda G_e^{m+1}$ is known such that

$$Q(\Lambda - \lambda I)Q^H G_e^{m+1} = G_o^{m+1} - \lambda G_e^{m+1}, \quad Q^H e_1 = \sigma_0^{-1}(\omega_1, \ldots, \omega_{m+1})^T \quad (3.1)$$

where

$$\sigma_0 = (\sum_{k=1}^{m+1} \omega_k^2)^{\frac{1}{2}}, \qquad \Lambda = diag(\lambda_1, \ldots, \lambda_{m+1}), \qquad \lambda_k = e^{\imath\theta_k}.$$

Instead of (3.1) we can just as well consider the equivalent equation

$$Q\Lambda Q^H = G_o^{m+1}(G_e^{m+1})^H.$$

Let (θ_j, ω_j^2) be the node-weight pair to be deleted from the solution. Using the knowledge of the above solution, we wish to construct a Schur parameter pencil $G_o^m - \lambda G_e^m$ or equivalently a matrix $G_o^m(G_e^m)^H$ such that

$$\widehat{W}\widehat{\Lambda}\widehat{W}^H = G_o^m(G_e^m)^H,$$

where

$$\begin{aligned} \widehat{\Lambda} &= diag(\lambda_1, \ldots, \lambda_{j-1}, \lambda_{j+1}, \ldots, \lambda_{m+1}), \\ \widehat{W}^H e_1 &= \widehat{\sigma}_0^{-1}(\omega_1, \ldots, \omega_{j-1}, \omega_{j+1}, \ldots, \omega_{m+1})^T, \\ \widehat{\sigma}_0 &= (\sigma_0^2 - \omega_j^2)^{\frac{1}{2}} \end{aligned}$$

or some suitable permutation of $\widehat{\Lambda}$ and $\widehat{W}^H e_1$. In other words, we wish to determine a unitary matrix V such that

$$\begin{pmatrix} 1 & 0 \\ 0 & V \end{pmatrix} \begin{pmatrix} \delta & \sigma_0 e_1^T \\ \sigma_0 e_1 & G_o^{m+1}(G_e^{m+1})^H \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & V^H \end{pmatrix} = \begin{pmatrix} \delta & \widehat{\sigma}_0 \widehat{e}_1^T & \omega_j \\ \widehat{\sigma}_0 \widehat{e}_1 & G_o^m (G_e^m)^H & 0 \\ \omega_j & 0 & \lambda_j \end{pmatrix}.$$

Then, as $\begin{pmatrix} \widehat{\Lambda} & \\ & \lambda_j \end{pmatrix} = P_j^H \Lambda P_j$ where $P_j = (e_1, \ldots, e_{j-1}, e_{j+1}, \ldots, e_{m+1}, e_j)$

$$\begin{pmatrix} \widehat{W} & \\ & 1 \end{pmatrix} = P_j^H Q^H V^H.$$

Analogous to the ideas of Ammar, Gragg and Reichel in [4] we want to compute $\widehat{W}$ via a QR-type step. In [13] a QR-like algorithm for Schur parameter pencils is introduced. The method is based on the standard QR algorithm applied to a matrix of the form $G_o G_e^H$. No initial reduction to Hessenberg form is performed. It is shown that each iterate is then of the same form as $G_o G_e^H$ again. Hence, applying one QR-step with the exact shift λ_j to the matrix $G_o^{m+1}(G_e^{m+1})^H$ determines a unitary matrix $\widetilde{V}$ such that

$$\widetilde{V} G_o^{m+1}(G_e^{m+1})^H \widetilde{V}^H = \begin{pmatrix} X & \\ & \lambda_j \end{pmatrix}$$

because λ_j is an eigenvalue of $G_o^{m+1}(G_e^{m+1})^H$. X is a unitary matrix of the same form as $G_o^m(G_e^m)^H$. $\widetilde{V}$ is a matrix of the form $G_1(G_3G_2G_3)(G_5G_4G_5)\cdots$. Hence, the vector $\sigma_0 e_1$ will not be transformed as required as $\widetilde{V} e_1$ is a full vector.

Applying one step of an RQ algorithm with the exact shift λ_j to $G_o^{m+1}(G_e^{m+1})^H$ determines an upper triangular matrix R and a unitary U of the form

$$G_1(G_3G_2G_3)(G_5G_4G_5)\cdots = G_1G_3G_5\cdots G_2G_3G_4G_5\cdots$$

that is, $U = G_o H$ for some upper Hessenberg matrix H and a matrix of the form G_o. R and U are determined such that $G_o^{m+1}(G_e^{m+1})^H - \lambda_j I = RU$ and

$$U G_o^{m+1}(G_e^{m+1})^H U^H = UR + \lambda_j I = \begin{pmatrix} \widehat{X} & \\ & \lambda_j \end{pmatrix}.$$

The vector $\sigma_0 e_1$ is transformed such that only the first two entries are nonzero

$$\sigma_0 U e_1 = (x, x, 0, \ldots, 0)^T.$$

Observe that with the reversal matrix $J = (e_{m+1}, e_m, \ldots, e_1)$

$$\begin{aligned} G_o^{m+1}(G_e^{m+1})^H = RU \quad &\Leftrightarrow \quad JG_o^{m+1}(G_e^{m+1})^H J = (JRJ)(JUJ) \\ &\Leftrightarrow \quad J(G_o^{m+1}(G_e^{m+1})^H)^T J = (JU^T J)(JR^T J). \end{aligned}$$

Let $G^P := J(G_o^{m+1}(G_e^{m+1})^H)^T J$ and let us assume for simplicity that $m+1$ is odd (a similar argumentation can be given in the case that $m+1$ is even). Then

$$G_o^{m+1}(G_e^{m+1})^H = G_1(\gamma_1)G_3(\gamma_3)\cdots G_{m+1}(\gamma_{m+1})G_2^H(\gamma_2)G_4^H(\gamma_4)\cdots G_m^H(\gamma_m)$$

and

$$G^P = G_1(\overline{\gamma_m})G_3(\overline{\gamma_{m-2}})\cdots G_{m-1}(\overline{\gamma_2})G_0(\gamma_{m+1})G_2^H(\gamma_{m-1})G_4^H(\gamma_{m-3})\cdots G_m^H(\gamma_1)$$

where

$$G_0(\gamma) = diag(-\gamma, I).$$

That is, $G_o^{m+1}(G_e^{m+1})^H$ and G^P are of the same form. Further, as U is a matrix of the form $G_o H$ for some upper Hessenberg matrix H and a matrix of the form G_o, $JU^T J$ is a matrix of the same form. Hence, as $JR^T J$ is upper triangular again, applying the RQ algorithm to $G_o^{m+1}(G_e^{m+1})^H$ is equivalent to applying the QR algorithm to G^P. One iteration of the QR algorithm with the exact shift λ_j applied to G^P generates a unitary matrix V

$$G_1(\beta_1)G_3(\beta_3)\cdots G_{m-1}(\beta_{m-1})G_2(\delta_2)G_3(\delta_3)\cdots G_{m+1}(\delta_{m+1})$$

such that

$$V^H G^P V = \begin{pmatrix} X' & 0 \\ 0 & \lambda_j \end{pmatrix}. \tag{3.2}$$

Moreover, δ_{m+1} can be taken to be an arbitrary unimodular number, because deflation has taken place.

Only the last two components of $\sigma_0 V^H e_{m+1}$ are nonzero; they are given by

$$\begin{pmatrix} 1 & 0 \\ 0 & -\overline{\delta_{m+1}} \end{pmatrix} \begin{pmatrix} -\overline{\delta_m} & (1-|\delta_m|^2)^{\frac{1}{2}} \\ (1-|\delta_m|^2)^{\frac{1}{2}} & \delta_m \end{pmatrix} \begin{pmatrix} 0 \\ \sigma_0 \end{pmatrix} = \begin{pmatrix} (1-|\delta_m|^2)^{\frac{1}{2}}\ \sigma_0 \\ -\overline{\delta_{m+1}}\ \delta_m\ \sigma_0 \end{pmatrix}.$$

We can choose $\delta_{m+1} = -\delta_m/|\delta_m|$ to obtain

$$\begin{pmatrix} (1-|\delta_m|^2)^{\frac{1}{2}}\sigma_0 \\ |\delta_m|\sigma_0 \end{pmatrix}.$$

Transforming (3.2) by similarity using $\widehat{J} = (e_m, e_{m-1}, \ldots, e_1, e_{m+1})$ and transposing the result, we obtain

$$W G_o^{m+1} (G_e^{m+1})^H W^H = \begin{pmatrix} X'' & 0 \\ 0 & \lambda_j \end{pmatrix}, \quad \text{where} \quad W = J\overline{V}\widehat{J}.$$

Moreover,

$$\sigma_0 W^H e_1 = \begin{pmatrix} (1 - |\delta_m|^2)^{\frac{1}{2}} \sigma_0 \widehat{e}_1 \\ |\delta_m| \sigma_0 \end{pmatrix},$$

and by the uniqueness of the reduction, $X'' = G_o^m (G_e^m)^H$, $\sigma_0 (1 - |\delta_m|^2)^{\frac{1}{2}} = \widehat{\sigma}_0$, and $|\delta_m| \sigma_0 = \omega_j$.

Note that the downdating process requires knowledge of the node θ_j to be deleted, but not of the corresponding weight ω_j^2. In an implementation of the process the computed weight can therefore be used to assess the accuracy of the computation.

If the optimal least-squares solution t of (1.2) corresponding to the data Y_{m+1} has been computed via the algorithm based on the generalized inverse unitary eigenproblem discussed in Section 2, the least-squares solution $\widetilde{t} = \widetilde{R}_1^{-1} \widetilde{Q}_1^H D \widetilde{f} \in \mathbb{C}^{m+1}$ of (2.1) is known. The optimal least-squares solution t of (1.2) corresponding to the data Y_m is then obtained by applying $\widehat{W}^H$ to $t' = \widetilde{Q}_1^H D \widetilde{f}$ incrementally

$$\widehat{W}^H t' = \begin{pmatrix} t'' \\ \omega_j f(z_j) \end{pmatrix}.$$

In a second step a new $\widetilde{t} \in \mathbb{C}^m$ has to be computed from t'' using the Schur parameters of $G_o^m (G_e^m)^H$ via a simplified Levinson algorithm. This is analogous to the second step of the updating procedure. There first $\widetilde{Q}_1^H D \widetilde{f}$ is computed via solving the generalized inverse unitary eigenproblem. Then $\widetilde{R}_1^{-1}$ is computed using the Schur parameters of G_o and G_e via a simplified Levinson algorithm. Details are given in [11].

A QR step has to be applied to a matrix X of the form $G_o G_e^H$. We get a unitary matrix $X' = V^H X V$, where X' can be written as $G_o' (G_e')^H$ again. If X corresponds to an unreduced Schur parameter pencil $G_o - \lambda G_e$, then X' will correspond to an unreduced Schur parameter pencil $G_o' - \lambda G_e'$. The transformation to an unreduced Schur parameter pencil is uniquely determined, up to unitary scaling, if the first column of the transformation matrix is given. Therefore one can derive $G_o' - \lambda G_e'$, up to scaling from $G_o - \lambda G_e$ by any unitary transformation $Q^H (G_o - \lambda G_e) P$ to Schur parameter pencils, for which the first column of Q coincides with a scalar multiple of the first column of V. This was used by Bunse-Gerstner and Elsner in [13] to derive an implicit single shifted QR step.

In a preparatory step, a matrix $V_1 = G_1(\alpha_1)$ is determined such that

$$V_1^H (G_o - \lambda_j G_e) e_1 = \rho e_1.$$

The pencil $V_1^H(G_o - \lambda G_e)$ differs from a Schur parameter pencil only by three additional entries

$$\begin{pmatrix} x & x & & & & \\ x & x & & & & \\ & & x & x & & \\ & & x & x & & \\ & & & & \ddots & \end{pmatrix} - \lambda \begin{pmatrix} x & + & + & & & \\ + & x & x & & & \\ & x & x & & & \\ & & & x & x & \\ & & & x & x & \end{pmatrix}.$$

This "bulge" is then chased down along the diagonal to restore the Schur parameter pencil form again. This is an $\mathcal{O}(m)$ process.

4 Concluding remarks

Due to space limitation, we will refrain from giving numerical examples for the updating and downdating procedures. For numerical examples and a detailed discussion on the updating process see [5, 10, 11]. For numerical examples and a detailed discussion on downdating Szegö polynomials/the downdating process in terms of the unitary eigenproblem see [4].

Bibliography

1. P. Henrici. *Applied and Computational Analysis*, volume 3. Wiley, 1986.

2. L.B. Scott and L.R. Scott. Efficient methods for data smoothing. *SIAM J. Numer. Anal.*, 26:681 – 692, 1989.

3. S. Elhay, G.H. Golub, and J. Kautsky. Updating and downdating of orthogonal polynomials with data fitting applications. *SIAM J. Matrix Anal. Appl.*, 12:327 – 353, 1991.

4. G.S. Ammar, W.B. Gragg, and L. Reichel. Downdating of Szegö Polynomials and Data Fitting Applications. *Lin. Alg. and its Applic.*, 172:315 – 336, 1992.

5. L. Reichel, G. S. Ammar, and W. B. Gragg. Discrete Least Squares Approximation by Trigonometric Polynomials. *Math. Comp.*, 57:273 – 289, 1991.

6. G. S. Ammar, W. B. Gragg, and L. Reichel. Constructing a Unitary Hessenberg Matrix from Spectral Data. In G.H. Golub and P. Van Dooren, editors, *Numerical Linear Algebra, Digital Signal Processing, and Parallel Algorithms*, pages 385–396. Springer-Verlag, Berlin, 1991.

7. H. Faßbender. Inverse unitary eigenproblems and related orthogonal functions. *Numerische Mathematik*, to appear.

8. G. H. Golub and C. F. Van Loan. *Matrix Computation.* The John Hopkins University Press, second edition, 1989.

9. G. S. Ammar, W. B. Gragg, and L. Reichel. On the Eigenproblem for Orthogonal Matrices. In *Proc. 25th IEEE Conference on Decision and Control*, pages 1963 – 1966, 1986.

10. H. Faßbender. *Numerische Verfahren zur diskreten trigonometrischen Polynomapproximation.* Dissertation Universität Bremen, 1993.

11. H. Faßbender. On Numerical Methods for Discrete Least-Squares Approximation by Trigonometric Polynomials. *Math. Comp.*, to appear.

12. W. B. Gragg. The QR algorithm for unitary Hessenberg matrices. *J. Comp. Appl. Math.*, 16:1 – 8, 1986.

13. A. Bunse-Gerstner and L. Elsner. Schur Parameter Pencils for the Solution of the Unitary Eigenproblem. *Lin. Alg. and its Appl.*, 154 - 156:741 – 778, 1991.