This norm has the following advantages.

(a) It allows us to evaluate partial derivatives of the norm in terms of polynomial and series manipulations. These can be used to express a sequence of least squares problems, whose solutions usually converge to a minimum perturbation $\|\Delta f\| = \|f - g \circ h\|$. The derivatives can also be used for Newton's method.

(b) Minimizing Δf gives a near-Chebyshev minimum on the unit disk [13].

(c) It permits fast algorithms for the solution of subproblems at each iteration.

The expression of f in the form (2) emphasizes the importance of the size of the values of $f(z)$ on the unit disk. This highlights the need for the following assumptions regarding the formulation of the problem:

(a) The location of the origin has been chosen (thus making explicit an implied assumption in previous numerical polynomial algorithms),

(b) The scale of $|z|$ has been chosen.

In particular, we assume that the problem context precludes a change of variable by an affine transformation $z \to bz + a$.

Remark. There is also a purely computational reason for avoiding such transformations, as is set out in the next theorem.

Theorem 2.2. *Shifting from z to $z - a$ can amplify any uncertainties in the coefficients of f by an amount as much as $(1 + |a|)^n / \sqrt{n+1}$ in norm. This is exponential in n, for any $a \neq 0$. Moreover, the relative uncertainties in each coefficient can be amplified by arbitrarily large amounts.*

In other words, such shifts are ill-conditioned.

Proof. By examining the condition of the matrix that determines the Taylor coefficients of the shifted polynomial

$$f_a(z) = \sum_{k=0}^{n} \frac{f^{(k)}(a)}{k!}(z - a)^k ,$$

one quickly finds the worst case for perturbation in the 1-norm: choose $f(z) = z^n$. Then $\|f\| = \|f\|_1 = 1$, but

$$f_a(z) = \sum_{k=0}^{n} \binom{n}{k} (-a)^k z^k$$

and hence $\|f_a\|_1 = (1 + |a|)^n$. Since the 1- and 2-norms are equivalent, with $\|f_a\|$

setting is that f is not monic, so

$$\begin{aligned}\tilde{f}(z) &= f_n + f_{n-1}z + f_{n-2}z^2 + \cdots f_0 z^m \\ &= \tilde{h}^{(0)}(z) = f_n^{1/m} + h_{d-1}z + \cdots + h_0 z^d\end{aligned}$$

and for definiteness we choose the positive (real) root for h_d (recall that we scaled f so that f = 1 with leading coeﬃcient greater than 0).

On the quality of the initial approximation

Let h_{min} and g_{min} be the functions that minimize Δf. Further, let $f + \Delta f_{min} = g_{min} \circ h_{min}$. A bound showing the quality of $h^{(0)}$ is given by the next theorem.

Theorem 3.1. *There exists a constant K, depending on m and on the leading coe cient of*

Repeat steps 2 and 3 until sufficient accuracy is attained or your patience is exhausted. When this method converges, it converges linearly since it is just functional iteration (similar to that discussed in [5]).

Essentially, we ignore the interactions between the changes in h and the changes in g. By doing so, we for-

The normal equations (12) can be arranged to get

$$\sum_{\ell=0}^{d} T_{k\ell}\, \Delta h_\ell = b_k, \qquad 0 \le k \le d,$$

where

$$\begin{aligned} T_{k\ell} &= [z^{k-\ell}]\, g'(h(z))\, \overline{g}'(\overline{h}(1/z)) \\ b_k &= [z^k]\, (f(z) - g(h(z)))\, \overline{g}'(\overline{h}(1/z)). \end{aligned}$$

This derivation allows for a very fast computation of the entries in T through series manipulation. To solve stably and efficiently such a system it is also necessary to know that it is non-singular and positive definite as well as Hermitian and Toeplitz. To see this, we observe that T factors as $T = B^* B$, where B is an $(n+d) \times (d+1)$ matrix with $B_{k\ell} = [z^k] z^\ell g'(h(z))$. This is a lower triangular Toeplitz matrix of full column rank when $g'(h(z))$ is non-zero, whence $B^* B$ is non-singular and positive definite.

Once we have computed a Δh using this linear least-squares formulation we may then update $h := h + \Delta h$. The entire process can then be iterated by re-linearizing around this new h and again approximating a Δh minimizing $\|f - g(h + \Delta h)\|$. Since this nonlinear least-squares problem is only a component in the entire solver it is not clear that it is necessary to repeat this local process (for fixed f and g) until convergence occurs. However, we have seen examples in which substantial convergence is required in this sub-problem for a globally minimal decomposition to be obtained.

Computationally, each iteration of this nonlinear least-squares solver has very low cost. Each system T can be easily constructed using only series manipulations. T can be constructed with $O(n \log^2 n)$ operations using the series manipulation algorithms of Brent & Kung [4] and an FFT for polynomial multiplication. The solution to the system can be obtained via the stable Toeplitz solvers of Trench [14] using (d^2) operations, or the fast and stable methods (for positive definite matrices) [3, 12] which require $O(d \log^2 d)$ operations. In summary, each iteration requires $O(n \log^2 n)$ floating point operations.

5 NEWTON ITERATION

In this section we explore the direct use of Newton's method to solve the nonlinear minimization problem: find $g, h \in \mathbb{R}[z]$ minimizing $\|g \circ h - f\|^2$. We give an effective method of computing the requisite derivatives analytically, and implement, test, and compare the method with the sequence of linear least-squares problems of the earlier section.

We consider

$$N_f(g + \Delta g, h + \Delta h) = \|f - (g + \Delta g) \circ (h + \Delta h)\|^2$$

with

$$\Delta h(z) = \sum_{\ell=0}^{d} \Delta h_\ell z^\ell, \quad \Delta g(z) = \sum_{\ell=0}^{m-1} \Delta g_\ell z^\ell.$$

Assume for the purpose of exposition that f, g, h, Δf, Δg, and Δh are all in $\mathbb{R}[z]$; however $z \in \mathbb{C}$. Lemma 2.1 is used to compute N_f. Denoting the integrand of the integral for N_f by I_f, we expand to second order in Δg, Δh.

$$\begin{aligned} I_f(g + \Delta g, h + \Delta h) = {} & (g(h(z)) - f(z))(g(h(\overline{z})) - f(\overline{z})) \\ & + g'(h(\overline{z}))(g(h(z)) - f(z))\Delta h(\overline{z}) \\ & + (g(h(z)) - f(z))\Delta g(h(\overline{z})) \\ & + g'(h(z))g'(h(\overline{z}))\Delta h(z)\Delta h(\overline{z}) \\ & + (\Delta g(h(z))\Delta g(h(\overline{z}))) \\ & + g'(h(\overline{z}))\Delta g(h(z))\Delta h(\overline{z}) \\ & + (g(h(z)) - f(z))\Delta g'(h(\overline{z}))\Delta h(z) \\ & + \frac{1}{2} g''(h(\overline{z}))(g(h(z)) - f(z))\Delta h^2(\overline{z}) \\ & + \text{c. c.}, \end{aligned}$$

where c. c. indicates the complex conjugate of all non-real summands.

We write

small eigenvalues, of course, may be greatly perturbed in a relative sense, but, as we will see below, to stabilize the Newton step we will ignore eigenvalues that are too small.

We write $A = Q\Lambda Q^t$, where Λ is the usual diagonal matrix of eigenvalues and Q is orthonormal. Now substitute $x = Qy$ in (13) to get

$$\begin{aligned} N_f + b^t x + x^t A x &= N_f + b^t Q y + y^t Q^t A Q y \\ &= b^t Q y + y^t \Lambda y. \end{aligned}$$

The constant N_f can be dropped. Denoting $b^t Q = -2p^t$, we have

$$\begin{aligned} b^t Q y + y^t \Lambda y = &- 2p_1 y_1 - 2p_2 y_2 - \cdots - 2p_m y_m \\ &+ \lambda_1 y_1^2 + \lambda_2 y \end{aligned}$$

Least Squares Iteration 2

$$g = 0.006265045950 + 0.$$