Numer Algor
https://doi.org/10.1007/s11075-018-0549-4

ORIGINAL PAPER

Comments on "Anderson Acceleration, Mixing and Extrapolation"

Donald G. M. Anderson[1]

Received: 3 May 2018 / Accepted: 10 May 2018

Abstract The Extrapolation Algorithm is a technique devised in 1962 for accelerating the rate of convergence of slowly converging Picard iterations for fixed point problems. Versions to this technique are now called Anderson Acceleration in the applied mathematics community and Anderson Mixing in the physics and chemistry communities, and these are related to several other methods extant in the literature. We seek here to broaden and deepen the conceptual foundations for these methods, and to clarify their relationship to certain iterative methods for root-finding problems. For this purpose, the Extrapolation Algorithm will be reviewed in some detail, and selected papers from the existing literature will be discussed, both from conceptual and implementation perspectives.

Keywords Fixed point problems · Picard iteration · Convergence acceleration · Anderson Acceleration · Anderson Mixing · Root-finding problems

1 Introduction

In 1962, during the course of my doctoral dissertation research, I devised a technique for accelerating the convergence of the Picard iteration associated with a fixed point problem, which I called the Extrapolation Algorithm. More recently, versions of this method have been labeled as Anderson Acceleration in the applied mathematics community and Anderson Mixing in the computational quantum mechanics community.

✉ Donald G. M. Anderson
anderson@fas.harvard.edu

[1] Gordon McKay Professor of Applied Mathematics, Emeritus, Harvard John A. Paulson School of Engineering and Applied Sciences, Cambridge, MA, USA

Published online: 05 June 2018

It would never have occurred to me to use the term Anderson Extrapolation, thence the quotation marks in the title.

I continued to work with the Extrapolation Algorithm, off and on, over the better part of two decades, but until recently have not had occasion to do so since. Only minimal records of this earlier work survive, so what follows is based on recollection and reconstruction, plus some new ideas. I am inclined to retire the Extrapolation language in favor of the Acceleration language, provided the purview of the latter is broadened to include the version that I shall outline hereafter. I shall also argue that that purview should be narrowed to focus on fixed point rather than root-finding problems. Mixing is a term of art in the computational quantum mechanics literature, with broader connotations, so this terminology seems likely to predominate in that community. A number of methods equivalent or related to versions of the Extrapolation Algorithm are now extant. Some of the existing literature will be reviewed from conceptual and implementation perspectives.

The technique was devised in a vastly different computational environment: imagine running Moore's Law backwards for 40 or 50 years. Correspondingly, the scale of the problems of interest is qualitatively as well as quantitatively different, and the relevant questions asked profoundly so. Adapting to current and projected computer capabilities is a task for a new generation.

2 The Extrapolation Algorithm

For $g : \mathbb{R}^N \to \mathbb{R}^N$, consider the problem of finding a fixed point $\hat{x} \in \mathbb{R}^N$ such that $g(\hat{x}) = \hat{x}$. We assume that a locally unique fixed point $\hat{x}$ exists and that g has all requisite or convenient smoothness properties. We seek iterants $x^{(\ell)} \to \hat{x}$, for $\ell = 0, 1, \ldots$. The Picard iteration associated with this fixed point problem is

$$x^{(\ell+1)} = g(x^{(\ell)}) =: y^{(\ell)},$$

for a given initial iterant $x^{(0)}$. The motivating assumption for the method is that the Picard iteration converges, but too slowly to be useful, so we seek a more rapidly converging sequence of iterants. We shall proceed on the basis of this assumption at the outset and reconsider it later together with other related issues.

For $u, v, w \in \mathbb{R}^N$, with $w > 0$, define the positive definite diagonal matrix $W = Diag(w) \in \mathbb{R}^{N\times N}$, so $w = diag(W)$. Define the inner product

$$< u|v >= \frac{1}{N}(Wu)^*(Wv) = \frac{1}{N}u^*W^2v,$$

and corresponding norm

$$\| u \| = \{< u|u >\}^{\frac{1}{2}} = \frac{1}{\sqrt{N}} \| Wu \|_2 .$$

Recall that the asterisk superscript denotes conjugate transposition for complex vectors and matrices. For real vectors and matrices, this reduces simply to transposition; its use in the real case usually leads naturally to the appropriate extension to the complex case. The complex analog of what follows is relevant, but we shall consider this

extension primarily implicitly via this and other notational devices. Most authors take $W = I$, which would simplify the foregoing and subsequent expressions. I shall later wish to consider $W \neq I$, so I will not make these simplifications here. Many authors also elide the $1/N$ factor, so $< u|v >= u^*v$ and $\| u \| = \| u \|_2$. For large and/or variable N, I find that inclusion of the $1/\sqrt{N}$ factor is often a more informative measure of size for large vectors. Formal extensions to more general inner products and to the Hilbert space context are straightforward.

For affine combination coefficients $\left\{\theta_k^{(\ell)}\right\}_{k=0}^{m^{(\ell)}}$ such that $\theta_k^{(\ell)} \in \mathbb{R}$,

$$\sum_{k=0}^{m^{(\ell)}} \theta_k^{(\ell)} = 1,\ \theta_0^{(\ell)} = 1 - \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)},$$

define affine combinations

$$u^{(\ell)} = \sum_{k=0}^{m^{(\ell)}} \theta_k^{(\ell)} x^{(\ell-k)} = x^{(\ell)} + \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)} (x^{(\ell-k)} - x^{(\ell)})$$

and

$$v^{(\ell)} = \sum_{k=0}^{m^{(\ell)}} \theta_k^{(\ell)} y^{(\ell-k)} = y^{(\ell)} + \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)} (y^{(\ell-k)} - y^{(\ell)}).$$

In outline form, the basic Extrapolation Algorithm proceeds as follows: Choose the maximal $m^{(\ell)}$ such that there are well-determined $\hat{\theta}_k^{(\ell)}, 0 \leq k \leq m^{(\ell)}$, minimizing $\| v^{(\ell)} - u^{(\ell)} \|$, with $0 \leq m^{(\ell)} \leq \min(\ell, M) \ll N$, and satisfying $\hat{\theta}_0^{(\ell)} > 0$. Take

$$x^{(\ell+1)} = \beta^{(\ell)} \hat{v}^{(\ell)} + (1 - \beta^{(\ell)}) \hat{u}^{(\ell)},$$

for $\beta^{(\ell)} > 0$. This sketch must be elaborated before implementation is possible, but there are more general issues to be addressed before doing so.

The minimizing coefficients $\hat{\theta}_k^{(\ell)}, 0 \leq k \leq m^{(\ell)}$, defining $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$ are of primary importance here; mathematically, unique $\hat{\theta}_k^{(\ell)}$ must exist, and numerically, they must be calculable sufficiently stably, accurately, and efficiently for the generation of a suitable $x^{(\ell+1)}$. The minimizer $\hat{v}^{(\ell)} - \hat{u}^{(\ell)}$ is of secondary interest, though one can write

$$x^{(\ell+1)} = \hat{u}^{(\ell)} + \beta^{(\ell)} (\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = \hat{v}^{(\ell)} - (1 - \beta^{(\ell)}) (\hat{v}^{(\ell)} - \hat{u}^{(\ell)}).$$

Most authors take a specified $\beta^{(\ell)} = \beta > 0$; many authors take $\beta = 1$, thereby simplifying the foregoing and subsequent expressions. I shall later wish to adaptively vary $\beta^{(\ell)}$, to choose β, once its role is understood.

The constraints $\hat{\theta}_0^{(\ell)} > 0$ and $\beta^{(\ell)} > 0$ ensure that new information, from $y^{(\ell)} = g(x^{(\ell)})$, is incorporated into $x^{(\ell+1)}$. They reflect the tacit assumption that the underlying Picard iteration is converging; however, this may not be essential. For $m^{(\ell)} > 0$, from

$$v^{(\ell)} - u^{(\ell)} = (y^{(\ell)} - x^{(\ell)}) + \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)} [(y^{(\ell-k)} + x^{(\ell)}) - (x^{(\ell-k)} + y^{(\ell)})]$$

we identify our task as that of finding the least squares solution of $A^{(\ell)}c^{(\ell)} = b^{(\ell)}$, where $b^{(\ell)} = W(x^{(\ell)} - y^{(\ell)})$, $e_k^* c^{(\ell)} = \theta_k^{(\ell)}$, and $A^{(\ell)}e_k = W[(y^{(\ell-k)} + x^{(\ell)}) - (x^{(\ell-k)} + y^{(\ell)})]$, for $1 \le k \le m^{(\ell)}$. For $m^{(\ell)} = 0$, we have $\hat{\theta}_0^{(\ell)} = 1$, $\hat{u}^{(\ell)} = x^{(\ell)}$, and $\hat{v}^{(\ell)} = y^{(\ell)}$. Clearly, the constraint $\hat{\theta}_0^{(\ell)} > 0$ can be satisfied for some admissible $m^{(\ell)}$, $0 \le m^{(\ell)} \le \min(\ell, M)$, and the constraint $\beta^{(\ell)} > 0$ is at our disposal.

To consider $g : \mathbb{C}^N \to \mathbb{C}^N$, it is most straightforward to admit $\theta_k^{(\ell)} \in \mathbb{C}$ and proceed as before. Restricting $\theta_k^{(\ell)} \in \mathbb{R}$ leads to a manageable, but different, computational problem: consider real and imaginary parts. If $\theta_0^{(\ell)}$ may not be real for $m^{(\ell)} > 0$, the constraint $\hat{\theta}_0^{(\ell)} > 0$ should be replaced by $| \hat{\theta}_0^{(\ell)} | > 0$ and handled similarly. This constraint could also be used for $g : \mathbb{R}^N \to \mathbb{R}^N$; most authors impose no constraint on $\hat{\theta}_0^{(\ell)}$. We continue to assume that $\beta^{(\ell)} > 0$.

This basic approach was later extended by introducing scaling, pivoting, and regularization. We replace $A^{(\ell)}c^{(\ell)} = b^{(\ell)}$ by the scaled equation $\tilde{A}^{(\ell)}\tilde{c}^{(\ell)} = \tilde{b}^{(\ell)}$, where $\tilde{c}^{(\ell)} = S^{(\ell)}c^{(\ell)}$, $\tilde{b}^{(\ell)} = (\sigma^{(\ell)})^{-1}b^{(\ell)}$, and $\tilde{A}^{(\ell)} = A^{(\ell)}(\sigma^{(\ell)}S^{(\ell)})^{-1}$. We then seek the least squares solution of

$$\begin{bmatrix} D^{(\ell)} \\ \tilde{A}^{(\ell)}P^{(\ell)} \end{bmatrix} (P^{(\ell)*}\tilde{c}^{(\ell)}) = \begin{bmatrix} D^{(\ell)}P^{(\ell)*} \\ \tilde{A}^{(\ell)} \end{bmatrix} \tilde{c}^{(\ell)} = \begin{bmatrix} 0 \\ \tilde{b}^{(\ell)} \end{bmatrix}.$$

The unitary permutation matrix $P^{(\ell)}$ and diagonal nonnegative definite regularization matrix $D^{(\ell)}$ are chosen during the pivoting process. Scaling and permutations can be carried out implicitly, to advantage for large N. We obtain $\tilde{c}^{(\ell)} = P^{(\ell)}(P^{(\ell)*}\tilde{c}^{(\ell)})$, $c^{(\ell)} = (S^{(\ell)})^{-1}\tilde{c}^{(\ell)} = \sigma^{(\ell)}(\sigma^{(\ell)}S^{(\ell)})^{-1}\tilde{c}^{(\ell)}$, and $b^{(\ell)} - A^{(\ell)}c^{(\ell)} = \sigma^{(\ell)}[\tilde{b}^{(\ell)} - \tilde{A}^{(\ell)}\tilde{c}^{(\ell)}]$. I suggest the choices $\sigma^{(\ell)} = \| b^{(\ell)} \|_2 > 0$, $S^{(\ell)} = Diag(s^{(\ell)})$, $\sigma^{(\ell)}S^{(\ell)} = Diag(\sigma^{(\ell)}s^{(\ell)})$, where $e_k^* s^{(\ell)} = \max\{1, \| A^{(\ell)}e_k \|_2 / \sigma^{(\ell)}\}$, $1 \le k \le m^{(\ell)}$.

I used Householder matrix triangularization with modifications of the standard pivoting strategy including right circular shifts rather than interchanges to privilege age ordering. The row-oriented form of the modified Gram–Schmidt process, with corresponding adjustments, could also be used. A key point here is that once the construction has been done for a candidate $m^{(\ell)}$, say $\min(\ell, M)$, the requisite quantities for smaller $m^{(\ell)}$, with the ordering chosen by the scaling and pivoting strategy, are readily available using byproducts thereof. Further details, motivation and rationale will be discussed after presentation of additional background material.

At this point, with some trepidation, I introduce the abbreviation $r^{(\ell-k)} = y^{(\ell-k)} - x^{(\ell-k)}$, for $0 \le k \le m^{(\ell)}$, assuming that $y^{(\ell-k)} \ne x^{(\ell-k)}$, so $r^{(\ell-k)} \ne 0$. The reasons for my trepidation will emerge later. At this stage, I simply emphasize that, in implementing the Extrapolation Algorithm, I always used $x^{(\ell-k)}$ and $y^{(\ell-k)}$, not $r^{(\ell-k)}$, as previously indicated! This abbreviation is just an expository or typographical convenience at times.

We have, using this abbreviation,

$$v^{(\ell)} - u^{(\ell)} = \sum_{k=0}^{m^{(\ell)}} \theta_k^{(\ell)} r^{(\ell-k)} = r^{(\ell)} + \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)} \left[r^{(\ell-k)} - r^{(\ell)} \right].$$

For $m^{(\ell)} = 0$, we have $\hat{v}^{(\ell)} - \hat{u}^{(\ell)} = r^{(\ell)}$. For $m^{(\ell)} > 0$, the set of all affine combinations of $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ constitute an affine subspace: a linear subspace translated by a nonzero shift vector, here chosen as $r^{(\ell)}$. $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is affinely independent or dependent according as that (affine or linear) subspace has dimension equal to or less than $m^{(\ell)}$, respectively. There will always be a unique $\hat{v}^{(\ell)} - \hat{u}^{(\ell)}$ in the affine subspace with minimal norm—closest to 0. There will be unique coefficients $\{\hat{\theta}_k^{(\ell)}\}_{k=0}^{m^{(\ell)}}$ characterizing $\hat{v}^{(\ell)} - \hat{u}^{(\ell)}$ if $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is affinely independent, and nonunique ones if it is affinely dependent. We see that the hypothesis or verification that $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is affinely independent is necessary for the Extrapolation Algorithm to be mathematically well-defined; for the coefficients to be well-determined involves further considerations of a numerical character. We see that $\{r^{(\ell-k)} - r^{(\ell)}\}_{k=1}^{m^{(\ell)}}$ spans the linear subspace associated with the shift vector $r^{(\ell)}$, so we infer that $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is affinely independent or dependent accordingly as $\{r^{(\ell-k)} - r^{(\ell)}\}_{k=1}^{m^{(\ell)}}$ is linearly independent or dependent. It is easily shown that linear independence of $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is a sufficient, but not a necessary, condition for the affine independence thereof and that linear dependence of $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is a necessary, but not a sufficient, condition for 0 to be a member of the affine span thereof, so $\hat{v}^{(\ell)} - \hat{u}^{(\ell)} = 0$. See Appendix.

We can write

$$\begin{aligned}
v^{(\ell)} - u^{(\ell)} &= r^{(\ell)} - \sum_{k=1}^{m^{(\ell)}} \theta_k^{(\ell)} \left[r^{(\ell)} - r^{(\ell-k)} \right], \\
&= r^{(\ell)} - \sum_{k=1}^{m^{(\ell)}} \sum_{j=1}^{k} \theta_k^{(\ell)} \left[r^{(\ell-j+1)} - r^{(\ell-j)} \right], \\
&= r^{(\ell)} - \sum_{j=1}^{m^{(\ell)}} \sum_{k=j}^{m^{(\ell)}} \theta_k^{(\ell)} \left[r^{(\ell-j+1)} - r^{(\ell-j)} \right], \\
&= r^{(\ell)} - \sum_{j=1}^{m^{(\ell)}} \xi_j^{(\ell)} \left[r^{(\ell-j+1)} - r^{(\ell-j)} \right],
\end{aligned}$$

with $\xi_j^{(\ell)} = \sum_{k=j}^{m^{(\ell)}} \theta_k^{(\ell)}$, $1 \le j \le m^{(\ell)}$. Setting $\xi_j^{(\ell)} = 0$ and 1 for $j = m^{(\ell)} + 1$ and 0 , respectively, we have $\theta_j^{(\ell)} = \xi_j^{(\ell)} - \xi_{j+1}^{(\ell)}$, for $j = m^{(\ell)}, m^{(\ell)} - 1, \ldots, 0$. Many authors use this reparameterization of the affine subspace, with variations in notation, sign, and indexing conventions. Since the linear span of $\{r^{(\ell-j+1)} - r^{(\ell-j)}\}_{j=1}^{m^{(\ell)}}$ is equal to that of $\{r^{(\ell-k)} - r^{(\ell)}\}_{k=1}^{m^{(\ell)}}$, both will be bases for the linear subspace associated with the shift vector $r^{(\ell)}$ if $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ is affinely independent. I shall call $\left\{r^{(\ell-j+1)} - r^{(\ell-j)}\right\}_{j=1}^{m^{(\ell)}}$ the difference basis, and $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m^{(\ell)}}$ the deviation basis. There are advantages, disadvantages, and pitfalls in choosing to use this reparameterization, rather than the original (and, I would argue, more natural) parameterization, as we shall see hereafter.

For example, observe that the argument above for the equivalence of the deviation and difference bases depends crucially on a telescoping sum and on an interchange of summations, which both require inclusion of the full sets of basis vectors. Deletion of the deviation basis vector $(r^{(\ell-i)} - r^{(\ell)})$, for $1 \leq i \leq m^{(\ell)}$, yields a proper subspace of the affine span of $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$ which is just the affine span of this set with $r^{(\ell-i)}$ deleted. For $i = 1$ or $m^{(\ell)}$, deletion of the difference basis vector $(r^{(\ell-i+1)} - r^{(\ell-i)})$ has the same consequence. For $1 < i < m^{(\ell)}$, this is not the case; we do obtain a proper subspace of the affine span of $\{r^{(\ell-k)}\}_{k=0}^{m^{(\ell)}}$, but one implicitly involving $r^{(\ell-k)}$, $0 \leq k \leq m^{(\ell)}$. This has implications for prospective reductions of $m^{(\ell)}$.

At this stage, we shall introduce some useful terminology applicable to the Extrapolation Algorithm and related techniques. We shall call the method stationary if $\beta^{(\ell)} = \beta > 0$ and $m^{(\ell)}$ is monotone nondecreasing. The method will be called quasistationary for $0 < m^{(\ell)} = \ell \leq M$, and equistationary for $0 < m^{(\ell)} = M < \ell$. The method will be called nonstationary if $m^{(\ell)}$ is allowed to decrease, though it might incidentally fail to do so in a particular instance. The method is also nonstationary if $\beta^{(\ell)}$ is allowed to vary adaptively. As envisioned above, the Extrapolation Algorithm is nonstationary. Most of the methods studied and used in the recent literature are ordinarily intended to be stationary, even quasistationary. This is one aspect of Anderson Acceleration, as presented in the widely cited Walker/Ni paper, that I would argue should be broadened. In fact, they do contemplate the possibility of decreasing $m^{(\ell)}$; the matter will be discussed in more detail later.

A related issue is the following: If $m^{(\ell)}$ is reduced, is data permanently discarded or just temporarily disregarded—in case it might prove useful in a subsequent iteration? The Walker/Ni approach most naturally discards data, based on age. In the Extrapolation Algorithm, it is most natural to disregard data by setting the associated affine combination coefficient equal to zero. It may be useful to accommodate zero coefficients when evaluating affine combinations. Data is discarded based strictly on age as needed to maintain $0 \leq m^{(\ell)} \leq M$: see further below.

Next, I shall introduce some ephemeral terminology. The distinction involved is of broader significance, but the terminology used is of narrower immediate concern and implies no value judgment. I shall characterize a "mathematical" fixed point problem as one for which the evaluation of g entails relative errors or uncertainties comparable to some moderate multiple of the unit roundoff error involved. I shall characterize a "scientific" fixed point problem as one for which the evaluation of g entails relative errors or uncertainties much larger than those for a "mathematical" problem, perhaps even providing only a small number of significant figures. Of course, a computer sees only "mathematical" problems, but may perceive "scientific" problems as lacking in smoothness.

Ideally, for large N and a reasonable initial iterant $x^{(0)}$, the cosine of the angle between $x^{(\ell)}$ and $y^{(\ell)} = g(x^{(\ell)})$ will be close to unity for most ℓ, or at least large enough ℓ, and their norms will be comparable. If the deviation or difference basis vectors at the heart of the computation are, in effect, evaluated as

$$(r^{(\ell-k)} - r^{(\ell)}) = [(y^{(\ell-k)} - x^{(\ell-k)}) - (y^{(\ell)} - x^{(\ell)})]$$

or

$$(r^{(\ell-j+1)} - r^{(\ell-j)}) = [(y^{(\ell-j+1)} - x^{(\ell-j+1)}) - (y^{(\ell-j)} - x^{(\ell-j)})],$$

there are three potentially cancellative subtractions, one of these combining results from the other two. If they are instead evaluated as

$$(r^{(\ell-k)} - r^{(\ell)}) = [(y^{(\ell-k)} + x^{(\ell)}) - (x^{(\ell-k)} + y^{(\ell)})]$$

or

$$(r^{(\ell-j+1)} - r^{(\ell-j)}) = [(y^{(\ell-j+1)} + x^{(\ell-j)}) - (x^{(\ell-j+1)} + y^{(\ell-j)})],$$

there are two potentially ameliorative additions and one potentially cancellative subtraction, the third of these combining the results of the first two. For "mathematical" problems, the consequences may be negligible, but for "scientific" problems, the former may magnify relative uncertainties significantly, especially in the later stages of the iteration when residuals are small. This is one reason to use $x^{(\ell-k)}$ and $y^{(\ell-k)}$ rather than $x^{(\ell-k)}$ and $r^{(\ell-k)}$ as the input, when implementing the algorithm.

Our final informal preliminary remarks concern qualitative aspects of the potential impact of anticipated ill-conditioning. At the outset, note that there are two sets of such issues involved. The fixed point problem itself may be ill-conditioned, and the least squares problems to be solved during the course of the iteration may be ill-conditioned. We shall focus here on the latter, which will inform later brief comments about the former. For our purposes, there are two distinguishable, but not separable, sources of ill-conditioning of the least squares problem $Ac = b$. The first is attributable to disparate sizes of the norms of the columns of A, or equivalently, the scaling of the coordinate system in which c is described. The second is attributable to near (or actual) linear dependence of the columns of A. For convenience, we shall assume hereafter that A has maximal rank so there is a unique least squares solution $\hat{c}$ minimizing $\| b - Ac \|_2$ and that $\hat{c} \neq 0$ and $\| b - A\hat{c} \|_2 > 0$.

Correspondingly, there are two distinguishable, but not separable, consequences of ill-conditioning of the least squares problem $Ac = b$. The first consequence is sensitivity of $\hat{c}$ to perturbations of A and/or b, thence also to errors involved in solving the problem approximately. By sensitivity is meant that the solution may suffer a disproportionately large change if the problem suffers a relatively small perturbation. The interested reader is referred to the literature for extensive quantitative discussion of sensitivity analysis for nonsingular linear equations and maximal rank least squares problems : see, for example, Björck [3] or Golub and Van Loan [9]. A qualitative appreciation will suffice for our purposes. The second consequence is ill-determination of $\hat{c}$. By ill-determination is meant that the residual may suffer a disproportionately small change if the solution suffers a relatively large perturbation. We shall establish later that, for $\check{c} \neq \hat{c}$,

$$0 \leq \frac{[\| b - A\check{c} \|_2 - \| b - A\hat{c} \|_2]}{\| A \|_2 \| \hat{c} \|_2} \leq \frac{\| A(\check{c} - \hat{c}) \|_2}{\| A \|_2 \| \hat{c} \|_2} \leq \frac{\| \check{c} - \hat{c} \|_2}{\| \hat{c} \|_2}.$$

If $\check{c}$ is close to $\hat{c}$ then $\| b - A\check{c} \|_2$ is close to $\| b - A\hat{c} \|_2$; $\hat{c}$ is ill-determined if there are $\check{c}$ not close to and possibly far from $\hat{c}$ such that $\| b - A\check{c} \|_2$ is close to $\| b - A\hat{c} \|_2$.

This occurs if $\| A(\check{c} - \hat{c}) \|_2 \ll \| A \|_2 \| \check{c} - \hat{c} \|_2$: near (or actual) linear dependence of the columns of A. Consequently, a $\| b - A\check{c} \|_2$ close to the minimum value $\| b - A\hat{c} \|_2$ does not entail that $\check{c}$ is close to the minimizer $\hat{c}$, so it is difficult to assess a putative approximate minimizer $\check{c}$. In the overall context of the problem, having $\| b - A\check{c} \|_2$ nearly minimal may suffice for the intended purposes, but if the focus is on $\check{c}$ as an approximation to $\hat{c}$, there is an issue to be addressed. It seems intuitively plausible that if $\hat{c}$ is ill-determined, then it may be sensitive to perturbations of A and/or b, with relatively large changes in directions $(\check{c} - \hat{c}) / \| \check{c} - \hat{c} \|_2$ corresponding to $\| A(\check{c} - \hat{c}) \|_2 \ll \| A \|_2 \| \check{c} - \hat{c} \|_2$.

Disparate sizes of the columns of A and reciprocal disparate sizes of the corresponding elements of $\hat{c}$ most directly influence sensitivity; this would have no effect on actual linear dependence, but does detract from our ability to meaningfully define and usefully detect near linear dependence, which most directly influences ill-determination, but indirectly impacts sensitivity. These pairs of sources and consequences are partially separable and mitigated if we scale the columns of A to be equal, or at least comparable, in norm. It is well known that this usually makes the scaled problem less ill-conditioned, which facilitates detection of near linear dependence. We can respond to diagnosed near linear dependence by redefining the problem to be solved as needed to achieve well-determination of the corresponding solution. Scaling is also essential to the efficacy of pivoting and regularization, as tools for accomplishing this goal. This is the overall motivation for invocation of our scaling, pivoting, and regularization strategy.

We must be cognizant of the fact that finding $\hat{\theta}_k^{(\ell)}$, $0 \leq k \leq m^{(\ell)}$, is a means to an end, not an end in itself. The end is accelerating the convergence of an iterative process to solve a fixed point problem.

3 Scaling, pivoting, and regularization

Implementation details given hereafter are included mainly to elucidate their intended consequences, which could be accomplished in other ways. Initially, as requisite background, we shall review the Householder matrix triangularization approach to solving the least squares problem $A^{(\ell)} c^{(\ell)} = b^{(\ell)}$, under the simplifying assumption that $A^{(\ell)}$ has maximal rank. We shall then extend this approach to incorporate scaling and pivoting, and finally, to accommodate near or actual rank deficiency, we shall incorporate regularization. Many of the tools used are standard, but their combination requires a special purpose algorithm and code.

We shall assume that the iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)} = g(x^{(\ell-k)})$, for $0 \leq k \leq \min(\ell, M)$, are stored and accessed as columns of $N \times (M+1)$ arrays X and Y, using a pointer evaluated as $(\ell - k)$ modulo $(M+1)$, for $\ell = 0, 1, \ldots$. Consequently, there is no need to realign the data as ℓ increases, which is desirable for $N \gg M$, and moreover, data is discarded for $\ell > M$ based strictly on age. The augmented matrix $[A^{(\ell)} \quad b^{(\ell)}]$ is formed using X and Y and stored in the first $\min(\ell, M) + 1$ columns of an $N \times (M+1)$ array AB. Consequently, multiplication of columns of $A^{(\ell)}$ by Householder matrices, as discussed hereafter, can readily be extended to include multiplication of $b^{(\ell)}$.

3.1 Background

For convenience, we simplify the notation to A, b and c, and set $m = \min(\ell, M)$ and $n = N$. We seek the QR decomposition of the $n \times m$ matrix A: that is, we seek a unitary $n \times n$ matrix Q and a regularly upper triangular $n \times m$ matrix R such that $A = QR$: that is, $Q^* = Q^{-1}$ and $e_i^* R e_j = 0$, for $i > j$, with $e_j^* R e_j \neq 0$. Note that $Q^*Q = QQ^* = I$ so Q^* is also unitary and that

$$\| Qx\|_2^2 = (Qx)^*(Qx) = x^*Q^*Qx = x^*x = \| x\|_2^2,$$

so $\| Qx \|_2 = \| x \|_2$ and $\| \cdot \|_2$ is said to be unitarily invariant. We shall do so by identifying unitary and Hermitian $n \times n$ Householder matrices H_k, $k = 1, 2, \ldots, m$, so $H_k^{-1} = H_k^* = H_k$ and thence H_k is self-inverse (or involutory), such that $Q = \prod_{k=1}^{m} H_k$, thence $Q^* = \prod_{k=m}^{1} H_k$, yielding $Q^*A = R$, thence $A = QR$. If we were to actually form Q and R, or at least $\hat{Q}$, we could write

$$A = QR = \left[\hat{Q} \;\; \check{Q}\right] \left[\begin{array}{c} \hat{R} \\ 0 \end{array}\right] = \hat{Q}\hat{R},$$

where $\hat{Q}$ and $\check{Q}$ are $n \times m$ and $n \times (n - m)$ column-rectangular orthonormal basis matrices for $R\{A\}$ and $R\{A\}^{\perp}$, respectively, so $\hat{Q}^*\hat{Q} = I$ and $\check{Q}^*\check{Q} = I$, and also $\hat{Q}^*\check{Q} = 0$ and $\check{Q}^*\hat{Q} = 0$, and $\hat{R}$ is an $m \times m$ regularly upper triangular, thence nonsingular (and conversely), matrix. Note that we have $\| \hat{Q}\hat{x} \|_2 = \| \hat{x} \|_2$ and $\| \check{Q}\check{x} \|_2 = \| \check{x} \|_2$, orthonormal invariance of $\| \cdot \|_2$, and also $(\check{Q}\check{x})^*(\hat{Q}\hat{x}) = 0$. We thereby obtain the QR factorization of A, $A = \hat{Q}\hat{R}$. It will suffice for our purposes to evaluate

$$d := Q^*b = \left[\begin{array}{c} \hat{Q}^*b \\ \check{Q}^*b \end{array}\right] =: \left[\begin{array}{c} \hat{d} \\ \check{d} \end{array}\right]$$

by using $Q^* = \prod_{k=m}^{1} H_k$ to calculate

$$Q^* [A \;\; b] = [R \;\; d].$$

Since $\| \cdot \|_2$ is unitarily invariant and Q^* is unitary, we see that

$$\| b - Ac\|_2^2 = \| Q^*(b - Ac)\|_2^2 = \| \hat{d} - \hat{R}c\|_2^2 + \| \check{d}\|_2^2.$$

It follows that the least squares solution $\hat{c}$ is obtained by solving $\hat{R}\hat{c} = \hat{d}$ and that $\| b - A\hat{c} \|_2 = \| \check{d} \|_2$. There is no need to actually form Q, thence $\hat{Q}$ and $\check{Q}$, or Q^*, thence $\hat{Q}^*$ and $\check{Q}^*$; $\hat{R}$, $\hat{d}$, and $\check{d}$ are at hand given $[R \;\; d]$. It will emerge that there is also no need to actually form H_k, $k = 1, 2, \ldots, m$. By hypothesis, we have $m \ll n$, so this is highly advantageous.

We shall digress briefly at this point to make some observations useful later. For notational convenience, we shall focus on the QR factorization $A = \hat{Q}\hat{R}$, but, as previously noted, the same information can be obtained from the QR decomposition $A = QR$.

Assume that $m > 1$ and choose any j such that $1 \leq j < m$. Partition A and $\hat{Q}$ after their jth column, $\hat{c}$ and $\hat{d}$ after their jth row, and $\hat{R}$ after its jth row and column, thence

$$[A_1\ A_2] = \left[\hat{Q}_1\ \hat{Q}_2\right]\begin{bmatrix}\hat{R}_{11} & \hat{R}_{12}\\ 0 & \hat{R}_{22}\end{bmatrix}.$$

Observe that $A_1 = \hat{Q}_1\hat{R}_{11}$, so the QR factorization of A_1 is embedded in that of A. We also have

$$\begin{bmatrix}\hat{d}_1\\ \hat{d}_2\end{bmatrix} = \begin{bmatrix}\hat{Q}_1^* b\\ \hat{Q}_2^* b\end{bmatrix}$$

and, provided $\hat{R}_{11}$ is nonsingular, we see that the least squares solution $\hat{c}_1$ of $A_1 c_1 = b$ can be obtained by solving $\hat{R}_{11}\hat{c}_1 = \hat{d}_1$. Moreover, we find that

$$\| b - A_1\hat{c}_1\|_2^2 = \| \check{d}\|_2^2 + \| \hat{d}_2\|_2^2.$$

This is equivalent to finding the basic least squares solution of $Ac = b$ obtained by setting $\hat{c}_2 = 0$, when A is rank deficient with rank $j < m$ so we have $\hat{R}_{11}$ nonsingular and $\hat{R}_{22} = 0$, and is often used when A is declared nearly rank deficient because $\hat{R}_{22}$ is regarded as negligibly small, in some specified sense. Thus, using the QR factorization or decomposition of A allows us to solve a family of embedded least squares problems, and to find corresponding basic least squares solutions.

We shall also consider the task of finding the minimal solution of

$$\left[\hat{R}_{11}\ \ \hat{R}_{12}\right]\begin{bmatrix}\hat{c}_1\\ \hat{c}_2\end{bmatrix} = \hat{d}_1.$$

This is equivalent to finding the minimal least squares solution of $Ac = b$, when A is rank deficient with rank $j < m$ so we have $\hat{R}_{22} = 0$ and is often used when A is nearly rank deficient with $\hat{R}_{11}$ nonsingular and $\hat{R}_{22}$ regarded as negligibly small. Defining $Z = (\hat{R}_{11})^{-1}\hat{R}_{12}$, we obtain

$$[I\ \ Z]\begin{bmatrix}\hat{c}_1\\ \hat{c}_2\end{bmatrix} = (\hat{R}_{11})^{-1}\hat{d}_1,$$

thence

$$\begin{bmatrix}\hat{c}_1\\ \hat{c}_2\end{bmatrix} = \begin{bmatrix}I\\ Z^*\end{bmatrix}(I + ZZ^*)^{-1}(\hat{R}_{11})^{-1}\hat{d}_1.$$

Since $\begin{bmatrix}-Z\\ I\end{bmatrix}$ is a standard basis matrix for the nullspace of $[I\ Z]$, we can write $\hat{c}$ as the sum of the counterpart basic solution and a member of the nullspace: to wit,

$$\begin{bmatrix}\hat{c}_1\\ \hat{c}_2\end{bmatrix} = \begin{bmatrix}(\hat{R}_{11})^{-1}\hat{d}_1\\ 0\end{bmatrix} + \begin{bmatrix}-Z\\ I\end{bmatrix}\hat{c}_2.$$

Choosing $\hat{c}_2$ to minimize $\| \hat{c}\|_2^2$, we obtain

$$\hat{c}_2 = (I + Z^*Z)^{-1}Z^*(\hat{R}_{11})^{-1}\hat{d}_1$$

and

$$\begin{aligned}\hat{c}_1 &= (\hat{R}_{11})^{-1}\hat{d}_1 - Z\hat{c}_2,\\ &= \left[I - Z(I + Z^*Z)^{-1}Z^*\right](\hat{R}_{11})^{-1}\hat{d}_1.\end{aligned}$$

Since Z is a $j \times (m-j)$ matrix, it would be more economical to find the Cholesky factorization of $(I+ZZ^*)$ when $j \leq (m-j)$ and of $(I+Z^*Z)$ when $j > (m-j)$. The latter situation would be more likely in our context, for moderately small M, but the former situation could arise for moderately large M. Note that the basic and minimal least squares solutions coincide if $\hat{R}_{12} = 0$, thence $Z = 0$. This "normal equations" approach, via Cholesky factorization using the standard scaling and pivoting strategy, will suffice for our purposes, but the more elegant approach via a QR factorization of $\left[\hat{R}_{11} \quad \hat{R}_{12}\right]^*$ might be preferable numerically.

We shall return now to the discussion of the Householder matrices H_k, $1 \leq k \leq m$. Construction of the QR decomposition of A and the resulting least squares solution of $Ac = b$ using Householder matrices is a standard topic in numerical linear algebra, covered in detail in any number of monographs or texts: for example, Björck [3] or Golub and Van Loan [9]. Professionally implemented general purpose codes for such algorithms are widely available and should be availed of when applicable. I shall focus here on those aspects requisite to understanding the modifications involved in designing a special purpose algorithm and code adapted to our purposes.

We shall adopt a formulation which extends gracefully from the real to the complex case. To this end, define $sgn(z) = z/ \mid z \mid$, for $0 \neq z \in \mathbb{C}$, and $sgn(0) = 1$. In particular, for $x \in \mathbb{R}$, we have $sgn(x) = 1, x \geq 0$, and $sgn(x) = -1, x < 0$. It is easily verified that $\mid sgn(z) \mid = 1, \overline{sgn(z)} = (sgn(z))^{-1}$, and $z = \mid z \mid sgn(z)$.

Again, we shall adopt generic notation $Ac = b$, with $A \in \mathbb{R}^{n\times m}, c \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$. We shall assume at the outset that $n > m \geq 1$, and usually $n \gg m > 1$, and that A has maximal rank, so $A \in \mathbb{R}_m^{n\times m}$. The rank-deficient situation and, more importantly, the nearly rank-deficient situation will be considered later. For $0 \neq v \in \mathbb{R}^n$, we shall call $I - (2/v^*v)vv^*$ an elementary reflector. It is easily verified that this elementary reflector is unitary and Hermitian and that it is unaltered by replacing v by αv, for $0 \neq \alpha \in \mathbb{R}$, the canonical choice being $\alpha = \| v \|_2^{-1}$. The term " elementary" customarily designates a matrix differing from the identity matrix by a matrix of rank one, representable by an outer product of two vectors. Householder used the terms elementary Hermitian or elementary unitary matrix, rather than elementary reflector (whose significance will be elucidated momentarily). The terms Householder reflector, Householder transformation, or Householder matrix are commonly employed in the current literature. My custom is to use the elementary reflector language at this level, and to attach the Householder name to the most commonly applied instances thereof.

We shall be interested in choosing v such that

$$\left(I - 2\left[\frac{vv^*}{v^*v}\right]\right)x = y,$$

for given $x \neq 0$, and suitable $y \neq x$. Since the elementary reflector is unitary, we must have $\| y \|_2 = \| x \|_2$. Recall that $\left[\frac{vv^*}{v^*v}\right]$ is the orthogonal projector onto the span of v, $spn(v)$, and

$$\left[\frac{vv^*}{v^*v}\right]x = \left[\frac{v^*x}{v^*v}\right]v$$

is the projection of x onto $spn(v)$. Moreover, recall that $\left(I - \left[\frac{vv^*}{v^*v}\right]\right)$ is the corresponding projector onto the orthogonal complement $spn(v)^\perp$, and

$$\left(I - \left[\frac{vv^*}{v^*v}\right]\right)x = x - \left[\frac{v^*x}{v^*v}\right]v$$

is the projection of x onto $spn(v)^\perp$. We then identify

$$y = \left(I - 2\left[\frac{vv^*}{v^*v}\right]\right)x = x - 2\left[\frac{v^*x}{v^*v}\right]v$$

as the reflection of x in $spn(v)^\perp$, thence the elementary reflector terminology. We see that

$$x - y = 2\left[\frac{v^*x}{v^*v}\right]v$$

is orthogonal to $spn(v)^\perp$, and $spn(v)^\perp$ (or the projection of x or y thereupon) bisects the angle between x and y. We shall now argue that we can choose $v = x - y$, which requires that $v^*x = \frac{1}{2}v^*v$. We have

$$(x - y)^*x = x^*x - y^*x$$

and

$$\begin{aligned}(x - y)^*(x - y) &= x^*x + y^*y - y^*x - x^*y,\\ &= 2\left\{x^*x + y^*x\right\} = 2(x - y)^*x,\end{aligned}$$

so $v^*x = \frac{1}{2}v^*v = x^*v$, since $x^*x = y^*y$ and, for real vectors, $y^*x = x^*y$. We also have

$$(x - y)^*y = x^*y - y^*y,$$

so we find that $-v^*y = \frac{1}{2}v^*v = -y^*v$. To have a graceful extension to the complex case, we must require not only that $\| y \|_2 = \| x \|_2$ but also that y^*x be real, so $y^*x = x^*y$; this will implicitly be arranged in what follows. (Note that one could then replace $v = x - y$ by $v = \alpha(x - y)$, for any $0 \neq \alpha \in \mathbb{C}$.)

My custom is to use the term Householder reflector for the elementary reflector $H(x) \in \mathbb{R}^{n\times n}$ (or $\mathbb{C}^{n\times n}$) such that

$$H(x)x = -sgn(e_1^*x) \| x \|_2 e_1 =: y,$$

for $0 \neq x \in \mathbb{R}^n$ (or $\mathbb{C}^n$), with $n \geq 2$. We see that $y \neq x$, $\| y \|_2 = | e_1^*y | = \| x \|_2$ and $y^*x = - \| x \|_2 | e_1^*x | = x^*y$. By the foregoing, we can write

$$H(x) = I - \left(2/v^*v\right)vv^*,$$

with

$$e_1^*v = sgn(e_1^*x)\left[| e_1^*x | + \| x \|_2\right]$$

and

$$e_i^*v = e_i^*x,\ 2 \leq i \leq n.$$

We see that

$$\frac{1}{2}v^*v = -y^*x = \| x \|_2 \left[| e_1^*x | + \| x \|_2\right],$$

so

$$(2/v^*v) = \left\{ \mid e_1^*y \mid\mid e_1^*v \mid \right\}^{-1}.$$

We could replace v by $\tilde{v} = v/e_1^*v$, so $e_1^*\tilde{v} = 1$, and write

$$H(x) = I - (2/\tilde{v}^*\tilde{v})\tilde{v}\tilde{v}^*,$$

where

$$(2/\tilde{v}^*\tilde{v}) = \left[\mid e_1^*x \mid + \parallel x \parallel_2 \right] / \parallel x \parallel_2 .$$

We have $1 \leq (2/\tilde{v}^*\tilde{v}) \leq 2$, thence $1 \leq \tilde{v}^*\tilde{v} \leq 2$. For $n \gg 1$, we expect $(2/\tilde{v}^*\tilde{v})$ to be near its lower bound. The reasons for considering $\tilde{v}$ will be explained below, but it will emerge that using v is most appropriate for our purposes. For $z \neq x$, we have

$$H(x)z = z - \left\{ (2/v^*v)(v^*z) \right\} v.$$

Observe, for later purposes, that if $v^*z = 0$ then $H(x)z = z$, and that if $e_i^*v = 0$ then $e_i^*H(x)z = e_i^*z$.

We seek $[R \;\; d] = Q^*[A \;\; b]$, where $Q^* = \prod_{k=m}^{1} H_k$. Defining $\left[R^{(0)} \;\; d^{(0)}\right] = [A \;\; b]$, we shall use Householder reflectors to construct Householder matrices H_k and

$$\left[R^{(k)} \;\; d^{(k)}\right] = H_k \left[R^{(k-1)} \;\; d^{(k-1)}\right],$$

for $k = 1, 2, \ldots, m$, such that $\left[R^{(m)} \;\; d^{(m)}\right] = [R \;\; d]$. Recall that, for any given $[A \;\; b]$, once we have extracted $\hat{R}$, $\hat{d}$, and $\check{d}$, from $[R \;\; d]$, all intermediate quantities involved in their calculation are of no further interest for our purposes. We shall consider a concise conceptual algorithm based on the foregoing and reconsider certain practical implementation issues related thereto.

For $k = 1$, set $H_1 = H(R^{(0)}e_1)$ and $R^{(1)}e_1 = H_1R^{(0)}e_1$. Set $R^{(1)}e_j = H_1R^{(0)}e_j$, for $j = 2, 3, \ldots, m$, and set $d^{(1)} = H_1d^{(0)}$. For $2 \leq k \leq m < n$, partition $R^{(k-1)}e_k$ after row $k-1$ as $R^{(k-1)}e_k = \begin{bmatrix} (R^{(k-1)}e_k)_1 \\ (R^{(k-1)}e_k)_2 \end{bmatrix}$. Set

$$H_k = \begin{bmatrix} I & 0 \\ 0 & H((R^{(k-1)}e_k)_2) \end{bmatrix}$$

and $R^{(k)}e_k = H_kR^{(k-1)}e_k$. Set $R^{(k)}e_j = H_kR^{(k-1)}e_j$, for $j = 1, 2, \ldots, k-1$ and $j = k+1, k+2, \ldots, m$, and set $d^{(k)} = H_kd^{(k-1)}$. Observe that $R^{(k)}e_j = R^{(k-1)}e_j$, for $1 \leq j \leq k-1$ and that $e_i^*R^{(k)}e_j = e_i^*R^{(k-1)}e_j$, for $1 \leq i \leq k-1$ and $k \leq j \leq m$. If $\left[R^{(k)} \;\; d^{(k)}\right]$ overwrites $\left[R^{(k-1)} \;\; d^{(k-1)}\right]$ in the AB array, for $k = 1, 2, \ldots, m$, we recognize that, for $2 \leq k \leq m$, the elements in the first $k-1$ rows and columns are unaltered. Consequently, for $2 \leq k \leq m$,
$H\left((R^{(k-1)}e_k)_2\right)$ operates on the submatrix of the augmented matrix obtained by deleting the first $k-1$ rows and columns isomorphically to the way $H(R^{(0)}e_1)$ operates on the entire augmented matrix. It therefore suffices to discuss implementation details for $\left[R^{(1)} \;\; d^{(1)}\right] = H_1\left[R^{(0)} \;\; d^{(0)}\right]$.

For notational convenience, set $x = R^{(0)}e_1$ and adopt the counterpart v, y, and z notation from the foregoing. Form $\parallel x \parallel_2 = \mid e_1^*y \mid$ and $e_1^*y = -sgn(e_1^*x) \parallel x \parallel_2$. Form $\left[\mid e_1^*x \mid + \parallel x \parallel_2 \right] = \mid e_1^*v \mid$ and $e_1^*v = sgn(e_1^*x)\left[\mid e_1^*x \mid + \parallel x \parallel_2 \right]$. Form

$$(2/v^*v) = \left\{ \parallel x \parallel_2 \left[\mid e_1^*x \mid + \parallel x \parallel_2 \right] \right\}^{-1}.$$

Since $e_i^* v = e_i^* x = e_i^* R^{(0)} e_1, 2 \leq i \leq n$, we can form v in the first column of AB in place of $R^{(0)} e_1$ by simply replacing $e_1^* R^{(0)} e_1$ by $e_1^* v$. We have characterized $H(x)$ since we now have v and $(2/v^* v)$ and can form $H(x)z$ for any designated z. Thus, we can form $R^{(1)} e_j = H(x) R^{(0)} e_j$, for $j = 2, 3, \ldots, m$, and $d^{(1)} = H(x) d^{(0)}$. Finally to form $R^{(1)} e_1 = y$, in the first column of AB, we can set $e_1^* R^{(1)} e_1 = e_1^* y$ and $e_i^* R^{(1)} e_1 = 0, 2 \leq i \leq n$. We recognize that $e_1^* \hat{R} = e_1^* R = e_1^* R^{(1)}$ and $e_1^* \hat{d} = e_1^* d = e_1^* d^{(1)}$ and, in particular, that $\mid e_1^* \hat{R} e_1 \mid = \mid e_1^* y \mid = \parallel x \parallel_2 = \parallel R^{(0)} e_1 \parallel_2$. Observe that to extract $\hat{R}$ from R, we really need only the significant elements of $\hat{R}$ on and above the diagonal, since the zero elements below the diagonal can be supplied automatically. Since we are interested only in $\hat{R}$, there is no need to set $e_i^* R^{(1)} e_1 = 0, 2 \leq i \leq n$, provided we leave the first column of AB unaltered thereafter. Note that if we save $e_1^* v$ separately and have $e_i^* v = e_i^* x, 2 \leq i \leq n$, below the diagonal in the first column of AB, we can reconstruct $H(x)$. This is the motivation for replacing v by $\tilde{v}$, since $e_1^* \tilde{v} = 1$ by construction and need not be saved separately, at the price of calculating $e_i^* \tilde{v} = e_i^* x / e_1^* v, 2 \leq i \leq n$, in the first column of AB. When $Q = \prod_{k=1}^{m} H_k$ or $Q^* = \prod_{k=m}^{1} H_k$ is needed in this factored form, for other purposes, this encoding would be advantageous. We are interested only in $\hat{R}$, $\hat{d}$, and $\check{d}$ and have no further need for Q or Q^*, so using v is more advantageous, since $e_i^* v = e_i^* x = e_i^* R^{(0)} e_1, 2 \leq i \leq n$.

For $2 \leq k \leq m$, we proceed to process the submatrix of the augmented matrix obtained by deleting the first $k-1$ rows and columns in isomorphic fashion, leaving the first $k-1$ rows and columns of AB unaltered.

In summary, the Householder matrix H_1 is the Householder reflector $H(R^{(0)} e_1)$, a special kind of elementary reflector. For $2 \leq k \leq m$, the Householder matrix H_k is also a special kind of elementary reflector:

$$H_k = I - (2/v^* v) v v^*.$$

Partitioning v after row $k-1$ as $\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$, we see that $v_1 = 0$ and v_2 is that associated with $H((R^{(k-1)} e_k)_2)$, so H_k is derived from a Householder reflector.

3.2 Scaling and pivoting

With this background at our disposal, we turn to the issues of scaling and pivoting, and later regularization. Least squares codes intended for statistical applications usually eschew scaling, because disparate character of the columns of A makes the issue problem dependent; the matter is left in the hands of the user. For codes intended for applications where the columns of A are of comparable character, a standard scaling strategy is commonly invoked. The original problem $Ac = b$ is replaced by a scaled problem $\tilde{A}\tilde{c} = \tilde{b}$, where $\tilde{A} = AS^{-1}$, $\tilde{c} = Sc$ and $\tilde{b} = b$, thence $c = S^{-1}\tilde{c}$. It would be relatively harmless, though somewhat redundant, to take $\tilde{b} = \sigma^{-1} b$, thence $c = \sigma S^{-1} \tilde{c}$. The customary choices would be $S = Diag(s)$, where $e_k^* s = \parallel Ae_k \parallel_2$, and $\sigma = \parallel b \parallel_2$. I suggest instead taking $\tilde{A} = A(\sigma S)^{-1}$, $\tilde{c} = Sc$ and $\tilde{b} = \sigma^{-1} b$, thence $c = S^{-1}\tilde{c} = \sigma(\sigma S)^{-1}\tilde{c}$ and $\parallel b - Ac \parallel_2 = \sigma \parallel \tilde{b} - \tilde{A}\tilde{c} \parallel_2$. My choice would be

$S = Diag(s), \sigma S = Diag(\sigma s)$, with $\sigma = \| b \|_2$ and $e_k^* s = \max\{1, \| Ae_k \|_2 / \sigma\}$, so $e_k^*(\sigma s) = \max\{\sigma, \| Ae_k \|_2\}$. Observe that if $\| Ae_k \|_2 \geq \| b \|_2$, $1 \leq k \leq m$, we obtain the same $\tilde{A}$ as with the standard scaling strategy, with $\| \tilde{A} e_k \|_2 = 1$. One might prefer to take $\sigma = \max\left\{\hat{\sigma}, \| b \|_2\right\}$, for $\hat{\sigma} > 0$, in case $\| b \|_2$ is too small. The motivation for this alternative to the standard scaling strategy will be discussed below, once we have reviewed the standard pivoting strategy.

For convenience, we return to generic notation $Ac = b$ in describing the pivoting process, with $A \in \mathbb{R}_m^{n \times m}$, $c \in \mathbb{R}^m$, and $b \in \mathbb{R}^n$. This will also facilitate discussion of the interaction between scaling and pivoting. The standard pivoting strategy generates a unitary permutation matrix $P \in \mathbb{R}^{m \times m}$, a unitary matrix $Q \in \mathbb{R}^{n \times n}$, and a regularly upper triangular matrix $R \in \mathbb{R}_m^{n \times m}$ such that $AP = QR$, thence also $Q^* AP = R$, $Q^* A = RP^*$ and $A = QRP^*$. As before, Q is obtained in factored form $Q = \prod_{k=1}^{m} H_k$, as a product of Householder matrices, thence $Q^* = \prod_{k=m}^{1} H_k$. P will be encoded in a permutation vector $p \in \mathbb{R}^m$, with $j = e_i^* p$ signifying that $Pe_i = e_j$, so $e_i^* P^* = e_j^*$. We see that $(AP)e_i = A(Pe_i) = Ae_j$.

P is chosen by the pivoting strategy so that

$$| e_k^* Re_k | \geq | e_{k+1}^* Re_{k+1} |, \quad 1 \leq k < m,$$

thence

$$| e_k^* Re_k | \geq | e_j^* Re_j |, \quad 1 \leq k < j \leq m.$$

As side effects, we will obtain

$$| e_k^* Re_k | \geq \left\{ \sum_{i=k}^{j} | e_i^* Re_j |^2 \right\}^{\frac{1}{2}},$$

thence

$$| e_k^* Re_k | \geq | e_k^* Re_j |,$$

for $1 \leq k < j \leq m$. We can usually expect, but cannot always guarantee, strict inequalities in the foregoing. Under the maximal rank assumption $A \in \mathbb{R}_m^{n \times m}$, R is regularly upper triangular, so $| e_k^* Re_k | > 0$, $1 \leq k \leq m$, thence $| e_k^* Re_k | > | e_k^* Re_j |$, $1 \leq k < j \leq m$. For rank-deficient $A \in \mathbb{R}_r^{n \times m}$, $1 \leq r < m$, we will have $| e_k^* Re_k | > 0$, $1 \leq k \leq r$, and $| e_k^* Re_k | = 0$, $r < k \leq m$, so we have $| e_i^* Re_j | = 0$, $r < i \leq j \leq m$. We can only assert that $| e_k^* Re_k | > | e_k^* Re_j |, k < j \leq m$, for $1 \leq k < r$, though it remains true (trivially for $r < k < m$) that $| e_k^* Re_k | \geq | e_k^* Re_j |$, for $1 \leq k < j \leq m$.

To work with the augmented matrix $[A \;\; b]$, to construct $[R \;\; d]$, we use instead

$$[A \;\; b] \begin{bmatrix} P & 0 \\ 0 & 1 \end{bmatrix} = Q [R \;\; d]$$

thence also

$$Q^* [A \;\; b] \begin{bmatrix} P & 0 \\ 0 & 1 \end{bmatrix} = [R \;\; d],$$

$$Q^* [A \;\; b] = [R \;\; d] \begin{bmatrix} P^* & 0 \\ 0 & 1 \end{bmatrix},$$

and

$$[A \;\; b] = Q\,[R \;\; d] \begin{bmatrix} P^* & 0 \\ 0 & 1 \end{bmatrix}.$$

These reduce to

$$[AP \;\; b] = [QR \;\; Qd],$$
$$\left[Q^*AP \;\; Q^*b\right] = [R \;\; d],$$
$$\left[Q^*A \;\; Q^*b\right] = \left[RP^* \;\; d\right],$$

and

$$[A \;\; b] = \left[QRP^* \;\; Qd\right].$$

Since $Ac = (AP)(P^*c)$, we can find the least squares solution $\hat{c}$ of $Ac = b$ by finding the least squares solution $(P^*\hat{c})$ of $(AP)(P^*c) = b$ and forming $\hat{c} = P(P^*\hat{c})$. This requires only extracting $\hat{R}$, $\hat{d}$, and $\check{d}$ from $[R \;\; d]$, solving for $(P^*\hat{c})$ as before and then recognizing that

$$e_i^*(P^*\hat{c}) = (e_i^*P^*)\hat{c} = e_j^*\hat{c}$$

where $j = e_i^*p$, $1 \le i \le m$. Though multiplications by P or P^* appear in mathematical expressions, their implementation requires only p, so they are never actually formed.

For our later purposes, we now introduce notation for two special classes of permutation matrices: $P_{k,i}$ and $P_{k:i}$, $1 \le k \le i \le m$. For later convenience, we set $P_{k,k} = P_{k:k} = I$. For $i > k$, define

$$\begin{aligned} P_{k,i} &= I - (e_i - e_k)(e_i - e_k)^*, \\ &= I - (e_ie_i^* + e_ke_k^*) + (e_ie_k^* + e_ke_i^*). \end{aligned}$$

We see that $(AP_{k,i})e_j = A(P_{k,i}e_j) = Ae_j$, for $j \ne i, k$; that $(AP_{k,i})e_k = A(P_{k,i}e_k) = Ae_i$; and that $(AP_{k,i})e_i = A(P_{k,i}e_i) = Ae_k$. Thus, multiplying A on the right by $P_{k,i}$ results in the interchange of the kth and ith columns. Examining $P_{k,i}^*A^* = P_{k,i}A^*$, we find that multiplying A^* on the left by $P_{k,i}^* = P_{k,i}$ results in interchange of the kth and ith rows.

For $i > k$, define

$$\begin{aligned} P_{k:i} &= \prod_{j=i-1}^{k} \left(I - (e_j - e_{j+1})(e_j - e_{j+1})^*\right) = \prod_{j=i-1}^{k} P_{j,j+1}, \\ &= I - \sum_{j=k}^{i-1} e_je_j^* + e_ie_k^* + \sum_{j=k}^{i-1} e_je_{j+1}^*. \end{aligned}$$

It is easily verified that multiplying A on the right by $P_{k:i}$ results in a right circular shift of columns k thru i, an interchange for $i = k + 1$. Likewise, multiplying A^* on the left by $P_{k:i}^*$ results in a right circular shift of rows k thru i, an interchange for $i = k + 1$. Since permutation matrices are unitary, so $P_{k:i}^* = P_{k:i}^{-1}$ we see that multiplying A on the right by $P_{k:i}^*$ results in a left circular shift of columns k thru i, an interchange for $i = k + 1$. Likewise, multiplying A^* on the left by $P_{k:i}$ results in a left circular shift of rows k thru i, an interchange for $i = k + 1$.

To preserve the correspondence between $P_{k,i}$ and $P_{k:i}$, we shall write $P^*_{k,i}$ rather than $P_{k,i}$ when multiplying on the left, in what follows.

For algorithmic purposes, we extend our previous notation to include a sequence of permutation matrices $P^{(k)}, 0 \leq k \leq m$, with $P^{(0)} = I$ and $P^{(m)} = P$. Correspondingly, we introduce a sequence of permutation vectors $p^{(k)}, 0 \leq k \leq m$, with $e_i^* p^{(0)} = i$ and $e_i^* p^{(m)} = e_i^* p, 1 \leq i \leq m$. At this point, we must choose between two alternative implementations. The more straightforward conventional alternative, in effect, involves explicitly forming $AP^{(k)}, 1 \leq k \leq m$. Choosing $Q^* = \prod_{k=m}^{1} H_k$ essentially as before, we ultimately obtain

$$Q^* [AP \;\; b] = \left[Q^* AP \;\; Q^* b\right] = [R \;\; d] .$$

The less straightforward unconventional alternative, in effect, involves implicitly forming $AP^{(k)}, 1 \leq k \leq m$, by accessing $(AP^{(k)})e_i = A(P^{(k)} e_i) = Ae_j$, where $j = e_i^* p^{(k)}, 1 \leq i \leq m$. In the end, we then obtain

$$Q^* [A \;\; b] = \left[Q^* A \;\; Q^* b\right] = \left[RP^* \;\; d\right];$$

and we would need to find $R = (RP^*)P$. More cogently, we recall that we just need $\hat{R} = (\hat{R}P^*)P$, available by extracting $(\hat{R}P^*)$ from (RP^*) and using

$$\hat{R}e_i = (\hat{R}P^*)(Pe_i) = (\hat{R}P^*)e_j,$$

where $j = e_i^* p, 1 \leq i \leq m$. This tacitly assumes that we actually form the zero elements below the diagonal of R, thence $\hat{R}$, but, as noted earlier, we can focus on the nontrivial elements on and above the diagonal and supply the zero elements of $\hat{R}$ below the diagonal as and if required. While the coding involved is somewhat more complex for the unconventional alternative, by comparison with the conventional alternative, this seems clearly to be a worthwhile investment of effort, since permuting m-vectors is much cheaper than permuting n-vectors under our assumption that $2 \leq m \ll n$. We shall therefore adopt the unconventional alternative in designing our algorithm, but will point out relevant aspects of a version based on the conventional alternative. This will later prove to have been an even more significant decision than now readily apparent.

We shall define $R^{(k)}$ and $d^{(k)}, 0 \leq k \leq m$, with $R^{(0)} = A, \;\; R^{(m)} = RP^*, d^{(0)} = b$, and $d^{(m)} = d$. Again, the first step, $k = 1$, is indicative. The available candidates for $\mid e_1^* R^{(1)} e_1 \mid = \mid e_1^* RP^* e_1 \mid$ are $\parallel R^{(0)} e_j \parallel_2, 1 \leq j \leq m$. Choose i as the smallest integer such that $1 \leq i \leq m$ and $\parallel R^{(0)} e_i \parallel_2 \geq \parallel R^{(0)} e_j \parallel_2$, for $i < j \leq m$. Note that choosing the smallest i is the standard tie-breaking rule, to make i unique. If the standard scaling strategy has been employed, so $\parallel R^{(0)} e_j \parallel_2 = 1, \;\; 1 \leq j \leq m$, the standard tie-breaking rule will yield $i = k = 1$; otherwise, we could have $i > k = 1$. Set $p^{(1)} = P^*_{1,i} p^{(0)}$. For $s = e_1^* p^{(1)} = e_i^* p^{(0)} = i$, take $H_1 = H(R^{(0)} e_s)$ and set

$$\left[R^{(1)} \;\; d^{(1)}\right] = H_1 \left[R^{(0)} \;\; d^{(0)}\right].$$

For $2 \leq k < m$, the available candidates for $\mid e_k^* R^{(k)} e_k \mid = \mid e_k^* RP^* e_k \mid$ are $\parallel (R^{(k-1)} e_t)_2 \parallel_2$, for $t = e_j^* p^{(k-1)}, k \leq j \leq m$. Choose i as the smallest integer

such that $k \le i \le m$ and $\| (R^{(k-1)}e_s)_2 \|_2 \ge \| (R^{(k-1)}e_t)_2 \|_2$, for $s = e_i^* p^{(k-1)}$ and $t = e_j^* p^{(k-1)}$, $i < j \le m$. Set $p^{(k)} = P_{k,i}^* p^{(k-1)}$. For $s = e_k^* p^{(k)} = e_i^* p^{(k-1)}$, take

$$H_k = \begin{bmatrix} I & 0 \\ 0 & H((R^{(k-1)}e_s)_2) \end{bmatrix}$$

and set

$$\begin{bmatrix} R^{(k)} & d^{(k)} \end{bmatrix} = H_k \begin{bmatrix} R^{(k-1)} & d^{(k-1)} \end{bmatrix}.$$

For $k = m$, there is only one available candidate for $| e_m^* R^{(m)} e_m | = | e_m^* R P^* e_m |$, namely $\| (R^{(k-1)}e_s)_2 \|_2$, for $s = e_m^* p^{(m-1)}$, and $p^{(m)} = p^{(m-1)}$. Identifying $s = e_i^* p^{(0)} = i$ and $t = e_j^* p^{(0)} = j$, we can largely combine the indicative case $k = 1$ with the subsequent cases $2 \le k \le m$. Again, we can focus attention, for $2 \le k \le m$, on the submatrix of the augmented matrix obtained by deleting the first $k-1$ rows and the $k-1$ columns designated by the first $k-1$ elements of $p^{(k-1)}$. We can also decide whether to form the zero elements below the diagonal in R.

For $2 \le k < m$, it will suffice to evaluate the candidates $\| (R^{(k-1)}e_t)_2 \|_2$, for $t = e_j^* p^{(k-1)}$, $k \le j \le m$, using the following observations, which also can be used to justify assertions about side effects and their consequences summarized above. However, when applying $H((R^{(k-1)}e_s)_2)$ thereafter, $\| (R^{(k-1)}e_s)_2 \|_2$ should be calculated directly, because of potential cancellations in the indirect calculations. Since Householder reflectors and matrices are unitary and $\| \cdot \|_2$ is unitarily invariant, we see that, for $1 \le k \le m$ and $1 \le j \le m$,

$$\| Ae_j \|_2 = \| R^{(0)}e_j \|_2 = \| R^{(k)}e_j \|_2 = \| RP^* e_j \|_2 .$$

Therefore, for $2 \le k < m$ and $t = e_j^* p^{(k-1)}$, $k \le j \le m$, we have

$$\| Ae_t \|_2^2 = \| R^{(k-1)}e_t \|_2^2 = \| (R^{(k-1)}e_t)_1 \|_2^2 + \| (R^{(k-1)}e_t)_2 \|_2^2,$$

thence

$$\begin{aligned} \| (R^{(k-1)}e_t)_2 \|_2 &= \left\{ \| Ae_t \|_2^2 - \| (R^{(k-1)}e_t)_1 \|_2^2 \right\}^{\frac{1}{2}}, \\ &= \left\{ \left[\| Ae_t \|_2 + \| (R^{(k-1)}e_t)_1 \|_2 \right] \left[\| Ae_t \|_2 - \| (R^{(k-1)}e_t)_1 \|_2 \right] \right\}^{\frac{1}{2}}. \end{aligned}$$

We likewise have

$$\| (R^{(k-1)}e_s)_2 \|_2 = | e_k^* R^{(k)} e_s | = | e_k^* R P^* e_s | = | e_k^* R e_k | .$$

We also observe more clearly the potential impact of scaling of A on the pivoting process. As we shall see shortly, scaling will also impact regularization, especially when conjoined with pivoting. For the class of problems of interest, I regard scaling as an essential precursor to pivoting, with or without conjoined regularization.

For the conventional alternative, we need only identify s with i and t with j throughout and write

$$[R^{(k)} \quad d^{(k)}] = H_k [R^{(k-1)} P_{k,i} \quad d^{(k-1)}]$$

for $k = 1, 2, \ldots, m$, with $[A \quad b] = [R^{(0)} \quad d^{(0)}]$ and $[R^{(m)} \quad d^{(m)}] = [R \quad d]$. If we wish to use a tie-breaking rule that respects the age ordering of the columns of A,

we can accomplish this simply by replacing $P_{k,i}$ with $P_{k:i}$ throughout. This would be prohibitively expensive for large n and the conventional alternative, but incurs negligible incremental cost for the unconventional alternative. For this reason, we adopt the unconventional alternative and use $P_{k:i}$, because privileging age ordering is a relevant feature of the problems of interest. Of course, if the occasion to invoke a tie-breaking rule does not arise, the same P will be obtained by using $P_{k,i}$ or $P_{k:i}$, and this is the likely outcome. At the potential price of giving up the monotone nonincreasing (usually decreasing) character of the diagonal elements of R, one could privilege age ordering further by not choosing the first candidate at least as large as the subsequent ones but rather the first candidate for which no subsequent one is significantly larger, according to some specified criterion. This may be more acceptable when used in conjunction with regularization, as described hereafter.

Recall that when combined with the standard scaling strategy, the standard pivoting strategy will always yield $APe_1 = Ae_1$, which is ideal from the age ordering perspective, but may be problematic in other respects in the problems of interest. Observed problem-dependent features motivated the choice of the nonstandard scaling strategy introduced above.

If the underlying Picard iteration is converging, and especially if the accelerated iteration is reasonably rapidly converging, the residuals $r^{(\ell-k)}$, $0 \leq k \leq m$, and the errors $\hat{x} - x^{(\ell-k)}$, can be expected to increase in norm significantly with increasing k : that is, increasing age. The same will be the case for the deviation basis vectors $r^{(\ell-k)} - r^{(\ell)}$; thence, the size of the columns of A can be expected to increase with age, since $Ae_k = W(r^{(\ell-k)} - r^{(\ell)})$. Because $b = -Wr^{(\ell)}$, it will usually be the case that the Ae_k will be larger than b; however, accidentally for smaller ℓ and systematically for larger ℓ, this may not be so for smaller k. Since $r^{(\ell)} \to 0$, the $r^{(\ell-k)}$ will tend to be more nearly linearly dependent for larger ℓ. There may be more useful information for discerning the convergence pattern, whose detection underlies the acceleration efficacy, in intermediate iterants than in the youngest ones. However, for nonlinear problems, we can anticipate the need to rely implicitly on local linearization in the neighborhood of $\hat{x}$, so older iterant data may be less representative and informative. These issues are accentuated for "scientific," as opposed to "mathematical," problems, where uncertainties are more significant. The nonstandard scaling strategy is designed to accommodate these observations, among other things by allowing the pivoting strategy to choose other than the youngest iterant data as APe_1. However, the nonstandard scaling strategy will essentially reduce to the standard scaling strategy in most instances where younger iterant data is most relevant. As discussed earlier, the combined scaling and pivoting strategies are intended to reorder based on redundancy, while privileging younger over older data where appropriate. Some authors prefer to prioritize age ordering, to the exclusion of scaling and/or pivoting—as I did myself at the outset: see further below.

3.3 Regularization

We turn now to a third mollifying device for assigning a generalized solution to $Ac = b$ when A is actually or nearly rank deficient: that is, the deviation basis vectors $r^{(\ell-k)} - r^{(\ell)}$, $1 \leq k \leq m$, are actually or nearly linearly dependent; thence, the

residuals $r^{(\ell-k)}$, $0 \leq k \leq m$, are actually or nearly affinely dependent. In this situation, the minimizer of $\| b - Ac\|_2^2$ is ill-defined or ill-determined. We therefore alter the minimization problem posed to determine $\hat{c}$ by a small change in the objective function sufficient to yield a sufficiently well-defined and well-determined $\hat{c}$.

As mentioned earlier, we seek instead the least squares solution of

$$\begin{bmatrix} DP^* \\ A \end{bmatrix} \hat{c} = \begin{bmatrix} D \\ AP \end{bmatrix} (P^*\hat{c}) = \begin{bmatrix} 0 \\ b \end{bmatrix},$$

using scaling and pivoting as previously described. D is a small diagonal nonnegative definite matrix chosen, together with P, as part of the pivoting strategy. Provided the nullspaces or near nullspaces of DP^* and A, or equivalently of D and AP, have trivial intersection, $\begin{bmatrix} DP^* \\ A \end{bmatrix}$ and $\begin{bmatrix} D \\ AP \end{bmatrix}$ will have maximal rank, and D can be chosen so that they are not nearly rank deficient according to some specified criterion: see further below. This then corresponds to minimizing $\| DP^*c\|_2^2 + \| b - Ac\|_2^2$; the small penalty term $\| DP^*c\|_2^2$ serves to resolve the ill-defined or ill-determined nature of the minimizer of $\| b - Ac\|_2^2$, without significantly altering the import of choosing $\hat{c}$ in the originating context of the problem.

We shall consider three approaches to choosing the regularization matrix D, which I characterize as broad, narrow, and dual regularization, the third being a combination of the first two. While any of these could be applied without scaling and pivoting, they are more sensible and effective if so conjoined; in particular, a smaller D may suffice. We shall proceed to modify the Householder matrix triangularization algorithm detailed above, incorporating the dual approach (which contains the broad and narrow approaches as special cases). Note that it is known (see Björck) that doing so with $D = 0$ is mathematically and numerically equivalent to using the modified Gram–Schmidt process to find the least squares solution of $Ac = b$, for maximal rank A. This is the basis for connecting the error analysis of the modified Gram–Schmidt process to that for the Householder matrix triangularization approach. (Strictly speaking, it would not be equivalent to the version of the modified Gram–Schmidt process reviewed below unless an extra Householder matrix was used to actually triangularize the augmented matrix, but this would serve only to calculate $\| \check{d} \|_2$, which can be done more efficiently directly.)

In the broad regularization approach, we take $D = \mu I$, with $\mu > 0$. There is a literature on choosing the regularization parameter μ, which we shall not pursue here. (I first encountered this in the Levenberg–Marquardt circle of ideas, but it would today usually be thought of in terms of Tikhonov regularization, ridge regression, or trust region methods.) In practice, μ is often chosen for a class of problems by experimenting with representative examples, though there are systematic methods in various contexts. The "broad" label connotes that all elements of c are treated alike in the penalty term.

In the narrow regularization approach, one takes $e_k^* D e_k$ as the smallest nonnegative quantity which will yield $| e_k^* R e_k | \geq \tau | e_1^* R e_1 |$, for $0 < \tau \ll 1$. In the dual regularization approach, one takes $e_k^* D e_k$ as the smallest nonnegative quantity greater than or equal to $\mu \geq 0$ which will yield $| e_k^* R e_k | \geq \tau | e_1^* R e_1 |$, for $0 \leq \tau \ll 1$. (We recover the broad approach for $\mu > 0$ and $\tau = 0$, and the narrow approach for $\mu = 0$

and $\tau > 0$.) The "narrow" label connotes that all elements of c are not treated alike in the penalty term. The "dual" label has an obvious significance. An advantage of the dual approach, as we shall see shortly, is that it can provide an adaptive way to choose μ, for a given τ, including the possibility of taking $D = 0$ if that will suffice. One can also choose τ adaptively.

If A is actually rank deficient, $A \in \mathbb{R}_r^{n\times m}$, for $1 \le r < m$, it is well known that the minimal least squares solution is the limit as $\mu \to 0$ of the broad regularization solution for $\mu > 0$, so the latter approximates the former for small positive μ, though this is not the most effective way to find the minimal solution. Clearly, the basic least squares solution is the limit as $\tau \to 0$ of the narrow regularization solution, and coincides with it for $\tau < \mid e_r^* R e_r \mid / \mid e_1^* R e_1 \mid$. In my final implementation, these limiting cases were included as available options, but calculated directly as discussed above. For the more common (and important) nearly rank-deficient case, I customarily used the adaptive dual regularization approach described hereafter.

To modify the foregoing algorithm to incorporate dual regularization, we work with an $(m + n) \times (m + 1)$ augmented matrix. We initialize by setting the elements of the first m rows of the augmented matrix equal to zero, choosing $e_k^* D e_k$ and modifying the kth row, for $k = 1, 2, \ldots, m$, as part of the pivoting process. Two observations, for $1 \le k \le m$, are crucial. First, the choice of $p^{(k)}$, thence $P^{(k)}$, can be based on $\| (R^{(k-1)} e_t)_2 \|_2$ with $t = e_j^* p^{(k-1)}$, $k \le j \le m$, obtained by partitioning $R^{(k-1)} e_t$ after the mth row. As noted previously, it will suffice, for $1 < k < m$, to calculate $\| (R^{(k-1)} e_t)_2 \|_2$ indirectly, but $\| (R^{(k-1)} e_s)_2 \|_2$, with $s = e_i^* p^{(k-1)} = e_k^* p^{(k)}$ should then be calculated directly. Second, for $1 < k < m$, recall that the first $k - 1$ rows and the columns corresponding to $e_j^* p^{(k-1)} = e_j^* p^{(k)}$, $1 \le j < k$, are unaltered, and note that while row k and rows $m+1$ thru $m+n$ of the remaining columns will be altered, rows $k + 1$ thru m will not be altered because the corresponding elements of $(R^{(k-1)} e_s)$ will be zero. This allows us to choose $e_k^* R^{(k-1)} e_s = e_k^* D P^* e_s = e_k^* D e_k$ so that $\mid e_k^* R^{(k)} e_s \mid = \mid e_k^* R P^* e_s \mid = \mid e_k^* R e_k \mid \ge \tau \mid e_1^* R e_1 \mid$, for $2 \le k \le m$, as intended and explained hereafter.

For $k = 1$, choose $e_1^* R^{(0)} e_s = \mu = e_1^* D e_1$, so we will obtain $\mid e_1^* R^{(1)} e_s \mid = \left\{ \| (R^{(0)}) e_s)_2 \|_2^2 + \mid e_1^* R^{(0)} e_s \mid^2 \right\}^{\frac{1}{2}} = \mid e_1^* R e_1 \mid$. Set $\delta_1 = 0$. For $2 \le k \le m$, proceed as follows: If $\| (R^{(k-1)} e_s)_2 \|_2 \ge \tau \mid e_1^* R e_1 \mid$, set $\delta_k = 0$. If $\| (R^{(k-1)} e_s)_2 \|_2 < \tau \mid e_1^* R e_1 \mid$, find δ_k such that

$$\| (R^{(k-1)} e_s)_2 \|_2^2 + (\mu + \delta_k)^2 = (\tau \mid e_1^* R e_1 \mid)^2,$$

thence $\mu + \delta_k > 0$ and

$$\begin{aligned} \mu + \delta_k &= \left\{ (\tau \mid e_1^* R e_1 \mid)^2 - \| (R^{(k-1)} e_s)_2 \|_2^2 \right\}^{\frac{1}{2}}, \\ &= \left\{ \left[\tau \mid e_1^* R e_1 \mid + \| (R^{(k-1)} e_s)_2 \|_2 \right] \left[\tau \mid e_1^* R e_1 \mid - \| (R^{(k-1)} e_s)_2 \|_2 \right] \right\}^{\frac{1}{2}}. \end{aligned}$$

Choose

$$e_k^* R^{(k-1)} e_s = \mu + \max(0, \delta_k) = e_k^* D e_k,$$

so we will obtain

$$\mid e_k^* R^{(k)} e_k \mid = \left\{ \parallel (R^{(k-1)} e_s)_2 \parallel_2^2 + \mid e_k^* R^{(k-1)} e_s \mid^2 \right\}^{\frac{1}{2}} = \mid e_k^* R e_k \mid .$$

By construction, we then have $\mid e_k^* R e_k \mid \geq \tau \mid e_1^* R e_1 \mid$, as intended.

Observe that we obtain broad regularization for $\mu > 0$ and $\tau = 0$, narrow regularization for $\mu = 0$ and $\tau > 0$, and dual regularization for $\mu > 0$ and $\tau > 0$. We can extend the foregoing to vary μ adaptively as follows: Define $\hat{\delta} = \min_k \delta_k$ and $\check{\delta} = \max_k \delta_k$. Take $\mu \geq 0$ and $\tau > 0$. Since $\delta_1 = 0$, we have $\hat{\delta} \leq 0$, and $\check{\delta} \geq 0$; for $\mu = 0$, we have $\hat{\delta} = 0$, and for $\mu > 0, \hat{\delta} > -\mu$. Using the standard pivoting strategy and tie-breaking rule, so $\parallel (R^{(k-1)} e_s)_2 \parallel_2$ is monotone nonincreasing (usually decreasing) as k increases, we see that if δ_k is ever nonzero then it is monotone nondecreasing (usually increasing) thereafter. If $\check{\delta} > 0$, take μ at the next iteration as $\mu + \frac{1}{2}\check{\delta}$. If $\check{\delta} = 0$, take μ at the next iteration as $\mu + \frac{1}{2}\hat{\delta}$. The $\frac{1}{2}$ factor in the adjustment of μ is a tuning parameter, and different values in $(0, 1]$ could be taken for increases than for decreases. If we start with $\mu = 0$ and find that $\hat{\delta} = \check{\delta} = 0$, we will have $D = 0$ and will take $\mu = 0$ at the next iteration; thus, if no regularization is ever required, none is employed, but it is available when needed. Of course, setting $\mu = \tau = 0$ would suppress regularization entirely.

In choosing to adopt the unconventional alternative in implementing the pivoting and regularization processes, we have—in effect—avoided manipulations of n-vectors, for large n, by carrying out permutations implicitly by manipulations of and with permutation vectors $p^{(k)}, 0 \leq k \leq m \ll n$, instead of and with the corresponding permutation matrices $P^{(k)}$. We could also carry out scaling implicitly by using a floating vector formalism for the relevant n-vectors. We can associate with each unscaled n-vector a nonzero scale factor by which it should be multiplied, initially one. While scale factors are usually envisioned as real and positive, this is not essential for our purposes. Observe that most of the manipulations of n-vectors we are concerned with here involve evaluation of norms, inner products, and linear combinations. These operations can easily be adjusted to accommodate the scale factors while manipulating the unscaled vectors. The ideas involved are simple enough to envision. We forbear from introducing the notation necessary to pursue the matter in more detail.

4 Choosing $m^{(\ell)}$

Several preliminary remarks are in order. First, we proceed on the assumption that the $\hat{R}, \hat{d}$, and $\parallel \check{d} \parallel_2$ quantities generated by the scaling, pivoting, and regularization strategies detailed in the previous section are at hand. We note, however, that some of the calculations discussed hereafter could be integrated into the earlier algorithms so their results are generated as byproducts thereof.

Second, we reiterate the premises that $1 < M \ll N$ and also that the cost of the ℓth iteration is dominated by that involved in the evaluation of $y^{(\ell)} = g(x^{(\ell)})$

and in the subsequent manipulation of N-vectors. By comparison, incremental costs involved in manipulating $M \times M$ matrices and M-vectors are relatively insignificant.

Third, we exploit the structure of $\hat{R}$ and $\hat{d}$ incident upon their mode of generation. Moreover, we recognize that computations that would rightly be regarded as prohibitively expensive in the context of solving a general nonsingular $M \times M$ linear system may be practical and productive for larger purposes within the context of the overall problem of interest. In particular, as discussed previously, we are interested not only in the generic linear equation $\hat{R}\hat{c} = \hat{d}$, with $\hat{R} \in \mathbb{R}^{m\times m}$, $\hat{c} \in \mathbb{R}^m$, and $\hat{d} \in \mathbb{R}^m$, but also in a family of related linear equations associated with

$$\begin{bmatrix} \hat{R}_{11} & \hat{R}_{12} \\ 0 & \hat{R}_{22} \end{bmatrix} \begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{bmatrix} = \begin{bmatrix} \hat{d}_1 \\ \hat{d}_2 \end{bmatrix},$$

where $\hat{c}$ and $\hat{d}$ are partitioned after the $(k-1)$th row and $\hat{R}$ is partitioned after the $(k-1)$th row and column, for $2 \le k \le m$. In the first instance, we shall assume that $\hat{R}$ is regularly upper triangular, thence nonsingular, so $\hat{R}_{11}$ and $\hat{R}_{22}$ have the same properties. We shall subsequently focus on the situation where $\hat{R}$ is nearly (or actually) singular.

It is easily verified that the diagonal elements of the inverse of a regularly upper triangular matrix are the reciprocals of the corresponding diagonal elements of the matrix. Because $\hat{R}$, as partitioned, is block upper triangular, it is also easily verified that

$$\begin{bmatrix} \hat{R}_{11} & \hat{R}_{12} \\ 0 & \hat{R}_{22} \end{bmatrix}^{-1} = \begin{bmatrix} \hat{R}_{11}^{-1} & -\hat{R}_{11}^{-1}\hat{R}_{12}\hat{R}_{22}^{-1} \\ 0 & \hat{R}_{22}^{-1} \end{bmatrix},$$

so

$$\begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{bmatrix} = \begin{bmatrix} \hat{R}_{11}^{-1}\hat{d}_1 - (\hat{R}_{11}^{-1}\hat{R}_{12})(\hat{R}_{22}^{-1}\hat{d}_2) \\ \hat{R}_{22}^{-1}\hat{d}_2 \end{bmatrix}.$$

Observe that $\hat{R}_{11}^{-1}$ is embedded in $\hat{R}^{-1}$, as is $\hat{R}_{22}^{-1}$, and that $\hat{d}_2 = 0$ implies $\hat{c}_2 = 0$ and $\hat{c}_1 = \hat{R}_{11}^{-1}\hat{d}_1$. The inverse could be calculated by recursion on k, see below, but it is equivalent and usually more straightforward to proceed column-by-column, solving upper triangular linear equations and exploiting the upper triangular character of the inverse.

As noted above, the initial candidate for $m^{(\ell)}$ is $\min(\ell, M) \ll N$, which is chosen if acceptable. Four factors may play a role in the acceptability of this initial candidate, or subsequent smaller candidates. The first factor is straightforward: the constraint $\hat{\theta}_0^{(\ell)} > 0$ must be satisfied. More concretely, we require that $\breve{\theta} \le \theta_0^{(\ell)}$, for a specified $\breve{\theta}$ such that $0 < \breve{\theta} < \frac{1}{2}$. If this constraint is not satisfied, the next smaller candidate for $m^{(\ell)}$ is considered. However, if the iterant data to be disregarded or discarded is the youngest available, and $2 \le m^{(\ell)} = \ell$, decrease $m^{(\ell)}$ by two instead of one, to assure that some older data has been disregarded or discarded. We know that the constraint must be satisfied for some nonnegative candidate. If the largest admissible candidate is 0 or 1, the constraint is dispositive; otherwise, other factors may motivate, or dictate, further reduction, as discussed hereafter.

4.1 Choice of M

Before proceeding, we shall pause briefly to consider the impact of the choice of M. I favor the modest values of M. In nonlinear problems, inclusion of unrepresentative older iterant data may be deliterious, and large $m^{(\ell)}$ may engender numerical difficulties. In the early days of the Extrapolation Algorithm (1960s), computational limitations restricted attention to $N \sim 10^2$ and $M \sim 3$, with relatively inexpensive g evaluations. It sufficed to solve the normal equations using Cholesky factorization, occasionally reducing $m^{(\ell)}$ based strictly on age as needed to keep the pivot elements retained large enough: see further below. Subsequently, with larger $N \sim 10^3$ and $M \sim 5$ the standard scaling and pivoting strategies, and broad regularization, were incorporated; these have equivalent counterparts for the normal equations. Later (1970s), available computational resources allowed $N \sim 10^4$ and $M \sim 10$; QR decomposition using Householder matrices was then employed, on numerical grounds: potential ill-conditioning.

In the early days of Anderson Mixing (1980s), and related methods for electronic structure calculations, storage limitations and costly g evaluations involving large N initially dictated $M \sim 2$. In recent years and a broad range of contexts, $N \sim 10^5 - 10^8$ and $M \sim 20 - 50$ have been considered by various authors. The normal equations and comparable approaches (see below) are still commonly employed, unfortunately.

Empirically, it is commonly observed that convergence acceleration performance initially increases with M, but tends to plateau for small to moderate values thereof, and may even decrease for larger values. The point of diminishing returns is problem dependent and is perhaps best chosen by preliminary experimentation for a given class of problems, if computational considerations do not intervene. Typical values observed range from 2 to 12, but may be larger in some cases. I prefer even values to better accommodate simple oscillatory, rather than monotone, convergence behavior. Larger M may be required if there are many significant oscillatory components with disparate periods. The acceleration process must be able to detect relevant patterns in the convergence of the iterants.

4.2 Triad

The plateauing behavior with increasing M is consistent with increasing influence of the triad of factors discussed hereafter, and the need to control $m^{(\ell)}$. The choices of $m^{(\ell)}$ determined by them may be indicative of an appropriate choice of M for a particular class of problems. The three factors in the triad are distinguishable, but not separable, and the relationships among them are important for our purposes.

The first factor in the triad is redundancy, which is operationally defined by the scaling, pivoting, and regularization strategies, as implemented in the previous section: see further below. The outcome is to arrange the columns of $\tilde{A}P$ in order of increasing redundancy, rather than increasing age as in $\tilde{A}$. We have tacitly already used this redundancy ordering in considering successive candidates for $m^{(\ell)}$ to satisfy the $\hat{\theta}_0^{(\ell)} > 0$ constraint; we shall continue to do so as additional criteria for choosing $m^{(\ell)}$ are invoked.

The second factor in the triad is relevance. The elements of $\hat{d}$ are the Fourier coefficients of $\tilde{b}$ with respect to the ordered orthonormal basis for the range of $\tilde{A}P$, $R\{\tilde{A}P\} = R\{\tilde{A}\}$, consisting of the columns of $\hat{Q}$ from the $\tilde{A}P = \hat{Q}\hat{R}$ factorization. We can regard $\mid e_k^* \hat{d} \mid^2 / \parallel \hat{d} \parallel_2^2$, $1 \leq k \leq m$, as a measure of the incremental relevance of the iterant data associated with $\tilde{A}Pe_k$, in approximating $\tilde{b}$ by a member of the range of $\tilde{A}P$, given that the contributions of previous columns of $\tilde{A}P$ have already been incorporated. Recall that $\parallel \hat{d} \parallel_2 = \parallel \tilde{A}\tilde{c} \parallel_2$, $\parallel \check{d} \parallel_2 = \parallel \tilde{b} - \tilde{A}\tilde{c} \parallel_2$ and $\parallel \hat{d} \parallel_2^2 + \parallel \check{d} \parallel_2^2 = \parallel \tilde{b} \parallel_2^2$, reflecting the fact that $\tilde{b} - \tilde{A}\tilde{c} \perp R\{\tilde{A}\}$. We can regard

$$\parallel \hat{d} \parallel_2^2 / \parallel \tilde{b} \parallel_2^2 = 1 - \parallel \check{d} \parallel_2^2 / \parallel \tilde{b} \parallel_2^2$$

as a measure of the collective relevance of the iterant data embodied in the columns of $\tilde{A}P$ in approximating $\tilde{b}$. The smaller the residual, the larger the collective relevance of the iterant data. The incremental relevance is then the fraction of the collective relevance contributed by each column, net of the prior contributions of the previous columns.

We would normally anticipate that data judged to be more redundant would be less relevant and that judged to be less redundant would be more relevant. However, anomalies are possible, for special $\tilde{b}$, with nonredundant data being irrelevant, or (less likely) redundant data being relevant. Note that redundance is a property of $\tilde{A}P$, while relevance is a property of $\tilde{A}P$ and $\tilde{b}$, and that scaling plays a role in redundancy and relevance, through the pivoting process. It is possible to use a nonstandard pivoting strategy to arrange for decreasing relevance, increasing irrelevance, rather than increasing redundance. However, this would respond to a desire to minimize rather than maximize $m^{(\ell)}$, for a given $\tilde{A}$ and $\tilde{b}$. On the other hand, irrelevance is of interest if $\parallel \hat{d}_2 \parallel_2 \ll \parallel \hat{d}_1 \parallel_2$, since we then would normally expect that $\parallel (P^*\tilde{c})_2 \parallel_2 \ll \parallel (P^*\tilde{c})_1 \parallel_2$: recall that if $\hat{R}_{22}$ is nonsingular, then $\hat{d}_2 = 0$ implies that $(P^*\tilde{c})_2 = 0$. Relevance can easily be monitored.

The third factor in the triad is conditioning—in particular, ill-conditioning. Since aspects of this are defined and quantified in terms of norms and condition numbers of matrices, we shall review—but basically take for granted—well-known facts about familiar examples: see Horn and Johnson [10], Björck [3], or Golub and Van Loan [9], et cetera. We take the occasion to amplify remarks made earlier, for later purposes.

4.3 Norms

Consider $\alpha, \beta \in \mathbb{C}$, $x, y \in \mathbb{C}^n$, and $A, B \in \mathbb{C}^{n \times n}$: square matrices. A vector norm defined on $\mathbb{C}^n$ induces a subordinate matrix norm defined on $\mathbb{C}^{n \times n}$ by

$$\parallel A \parallel = \max_{x \neq 0} (\parallel Ax \parallel / \parallel x \parallel),$$

or equivalently,

$$\parallel A \parallel = \max_{\parallel x \parallel = 1} \parallel Ax \parallel .$$

It follows that the matrix norm inherits the positive definiteness, homogeneity, and subadditivity properties characterizing the vector norm:

(1) $\| A \| \geq 0$ and $\| A \| = 0 \iff A = 0$
(2) $\| \alpha A \| = | \alpha | \| A \|$
(3) $\| A + B \| \leq \| A \| + \| B \|$.

Consequently, the matrix norm defines a norm on the linear vector space $\mathbb{C}^{n \times n}$. It also follows that the subordinate matrix norm has the submultiplicativity property

(4) $\| BA \| \leq \| B \| \| A \|$

so the matrix norm defines a norm on the linear algebra $\mathbb{C}^{n \times n}$. The subadditivity property is often called the triangle inequality, and the submultiplicativity property is often called the consistency condition—the matrix norm being termed consistent. Clearly, the vector and induced matrix norms satisfy the compatibility condition

(5) $\| Ax \| \leq \| A \| \| x \|, \forall x \in \mathbb{C}^n$

for any given A—the two norms being termed compatible. Moreover, for every A, the compatibility inequality is sharp (satisfied as an equality for some x), which characterizes a subordinate matrix norm induced by a compatible vector norm. We shall later encounter compatible vector and matrix norms satisfying (1)–(5) for which (5) may be sharp for some A, but not all A, so the matrix norm is not induced by and subordinate to the vector norm. Finally, we observe the normalization property of a subordinate matrix norm

(6) $\| I \| = 1$.

Any matrix norm not satisfying (6) cannot be a subordinate matrix norm; however, (6) can be satisfied for matrix norms which are not subordinate.

For compatible vector and matrix norms, if λ is an eigenvalue of B and $x \neq 0$ is an associated eigenvector, we have

$$| \lambda | \| x \| = \| \lambda x \| = \| Bx \| \leq \| B \| \| x \|,$$

so $| \lambda | \leq \| B \|$. The spectral radius $\rho(B)$ is the maximum of the magnitudes of the eigenvalues of B, so $\rho(B) \leq \| B \|$. Ordinarily, this inequality is strict, and $\| B \|$ may be substantially larger than $\rho(B)$. For special B and/or $\| B \|$, $\| B \|$ may equal or closely approximate $\rho(B)$.

We shall be primarily concerned with three vector norms, for which formulae can be given for their induced matrix norms:

$$\| x \|_1 = \sum_k | e_k^* x |,$$
$$\| x \|_\infty = \max_k | e_k^* x |,$$

and

$$\| x \|_2 = \left\{ \sum_k | e_k^* x |^2 \right\}^{\frac{1}{2}}.$$

By establishing that, for every A, the compatibility inequality is satisfied and is sharp, it can be shown that

$$\| A \|_1 = \max_j \| Ae_j \|_1,$$
$$\| A \|_\infty = \max_i \| A^* e_i \|_1,$$

and

$$\| A \|_2 = \sqrt{\rho(A^*A)}.$$

We see that $\| A \|_\infty = \| A^* \|_1$, thence $\| A^* \|_\infty = \| A \|_1$, and we observe that both $\| A \|_1$ and $\| A \|_\infty$ are readily computable. We have

$$\| A^* \|_2 = \sqrt{\rho(AA^*)}.$$

We shall outline a proof of this formula for $\| A \|_2$ because related results are useful for our later purposes. In particular, we shall argue below that A^*A and AA^* are Hermitian and nonnegative definite, and they share the same nonzero, thence positive, eigenvalues. Therefore, we infer that $\rho(A^*A) = \rho(AA^*)$, thence $\| A^* \|_2 = \| A \|_2$, and we observe that $\| A \|_2$ can be computed by an efficient iterative process, but not readily. Consequently, readily computable bounds on $\| A \|_2$ are of interest. From $\rho(A^*A) \leq \| A^*A \|_1 = \| A^*A \|_\infty$, we obtain using submultiplicativity and the foregoing

$$\rho(A^*A) \leq \| A \|_1 \| A \|_\infty,$$

thence

$$\| A \|_2 \leq \sqrt{\| A \|_1 \| A \|_\infty}.$$

We shall later study other readily computable upper bounds, and also counterpart lower bounds.

Recall that if $B \in \mathbb{C}^{n\times n}$ is Hermitian, $B^* = B$, the eigenvalues of B are real and can be labeled so that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Moreover, there is an orthonormal basis for $\mathbb{C}^n$ consisting of associated eigenvectors: $Bv_k = \lambda_k v_k$, $v_i^* v_j = \delta_{ij}$, $1 \leq i, j, k \leq n$. Consequently, for any $x \in \mathbb{C}^n$, we have $x = \sum_{k=1}^n \xi_k v_k$, with $\xi_k = v_k^* x$. It is easily shown that $\forall x \neq 0$, we have

$$\lambda_1 \geq x^*Bx/x^*x = \sum_{k=1}^n \lambda_k \mid \xi_k \mid^2 / \sum_{\ell=1}^n \mid \xi_\ell \mid^2 \geq \lambda_n.$$

These bounds are sharp (satisfied as equalities for the corresponding eigenvectors), so we obtain

$$\lambda_1 = \max_{x\neq 0} \left\{ x^*Bx/x^*x \right\}$$

and

$$\lambda_n = \min_{x\neq 0} \left\{ x^*Bx/x^*x \right\}$$

This is the basic form of the Rayleigh Principle, characterizing the extreme eigenvalues λ_1 and λ_n. We shall also be interested in the extended Rayleigh Principle, which follows similarly, characterizing the intermediate eigenvalues λ_k, $1 < k < n$. Define

$$S_k = spn\{v_1, v_2, \ldots, v_k\} = spn\{v_{k+1}, v_{k+2}, \ldots, v_n\}^\perp$$

and

$$T_k = spn\{v_k, v_{k+1}, \ldots, v_n\} = spn\{v_1, v_2, \ldots, v_{k-1}\}^{\perp}.$$

The orthogonal complement specification of S_k and T_k is most useful in applications of the results to follow, but their proofs flow most naturally from the other specification. We then obtain

$$\lambda_k = \min_{0 \neq x \in S_k} \left\{ x^* B x / x^* x \right\}$$

and

$$\lambda_k = \max_{0 \neq x \in T_k} \left\{ x^* B x / x^* x \right\}$$

The first characterization is most interesting for $\lambda_k > \lambda_{k+1}$, and the second for $\lambda_k < \lambda_{k-1}$. Clearly, we also have

$$\lambda_1 = \max_{0 \neq x \in S_k} \left\{ x^* B x / x^* x \right\}$$

and

$$\lambda_n = \min_{0 \neq x \in T_k} \left\{ x^* B x / x^* x \right\}.$$

We shall focus hereafter on A^*A, omitting the parallel arguments for AA^*. Since $(A^*A)^* = A^*A$, we may set $B = A^*A$ in the foregoing, so $\forall x \neq 0$ we have

$$\lambda_1 \geq x^*(A^*A)x/x^*x = \| Ax\|_2^2 / \| x\|_2^2 \geq \lambda_n \geq 0,$$

and conclude that A^*A, is nonnegative definite. If A is nonsingular, then $x \neq 0 \Rightarrow Ax \neq 0$, so we see that $\lambda_n > 0$ and A^*A is positive definite. If A is singular, there are $x \neq 0$ such that $Ax = 0$, including v_n, so we see that $\lambda_n = 0$ and A^*A is positive semidefinite. Consequently, we obtain

$$\max_{x \neq 0} \left\{ \| Ax\|_2^2 / \| x\|_2^2 \right\} = \lambda_1 = \rho(A^*A),$$

thence

$$\| A \|_2 = \max_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} = \left[\max_{x \neq 0} \left\{ \frac{\| Ax\|_2^2}{\| x\|_2^2} \right\} \right]^{\frac{1}{2}} = \sqrt{\lambda_1} = \sqrt{\rho(A^*A)},$$

as previously asserted.

In addition, we also obtain

$$\min_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} = \left[\min_{x \neq 0} \left\{ \frac{\| Ax\|_2^2}{\| x\|_2^2} \right\} \right]^{\frac{1}{2}} = \sqrt{\lambda_n} \geq 0.$$

The minimum value is zero for singular A and greater than zero for nonsingular A. If A is nonsingular, then A^*A is positive definite, thence nonsingular; the eigenvalues of $(A^*A)^{-1}$ are λ_k^{-1}, $1 \leq k \leq n$. We see that $\lambda_n^{-1} = \rho((A^*A)^{-1})$, thence $\lambda_n = \rho((A^*A)^{-1})^{-1}$ and $\sqrt{\lambda_n} = \rho((A^*A)^{-1})^{-\frac{1}{2}}$. Furthermore, we find that $(A^*A)^{-1} = A^{-1}(A^*)^{-1} = A^{-1}(A^{-1})^*$, and conclude that

$$\min_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} = \rho(A^{-1}(A^{-1})^*)^{-\frac{1}{2}} = \| A^{-1} \|_2^{-1},$$

using results anticipated above, that will now be established.

What remains to be shown is that A^*A and AA^* share the same nonzero, thence positive, eigenvalues. If $\lambda \neq 0$ is an eigenvalue of A^*A and $v \neq 0$ is an associated eigenvector, we have $(A^*A)v = \lambda v \neq 0$, and see that $(AA^*)(Av) = \lambda(Av)$. Since $Av = 0$ would imply that $A^*Av = 0$, which would contradict the fact that $\lambda v \neq 0$, we infer that $Av \neq 0$. We then identify λ as a nonzero eigenvalue of AA^* with Av as an associated eigenvector. Thus, all nonzero, thence positive, eigenvalues of A^*A are also eigenvalues of AA^*. The parallel argument for AA^*, which is omitted, then shows that A^*A and AA^* share the same positive eigenvalues, thence $\rho(A^*A) = \rho(AA^*)$ and $\| A \|_2 = \| A^* \|_2$, as asserted above. In this square matrix case, A^*A and AA^* will also share their zero eigenvalues if A, thence A^*, is singular: see further below.

If U is a unitary matrix, $U^* = U^{-1}$, we have (as above)

$$\| Ux \|_2^2 = (Ux)^*(Ux) = x^*(U^*U)x = x^*x = \| x \|_2^2,$$

so the $\| \cdot \|_2$ vector norm is unitarily invariant, with respect to left multiplication by a unitary matrix. It follows from $\| UAx \|_2 = \| Ax \|_2$ that $\| UA \|_2 = \| A \|_2$. For every $y \in \mathbb{C}^n$, there is a unique $x \in \mathbb{C}^n$, namely $x = U^*y$, such that $Ux = y$ and $\| x \|_2 = \| y \|_2$. Conversely, for every $x \in \mathbb{C}^n$, there is a unique $y \in \mathbb{C}^n$, namely $y = Ux$, such that $U^*y = x$ and $\| y \|_2 = \| x \|_2$. It follows that we have

$$\| AU \|_2 = \max_{x \neq 0} \left\{ \frac{\| AUx \|_2}{\| x \|_2} \right\} = \max_{y \neq 0} \left\{ \frac{\| Ay \|_2}{\| y \|_2} \right\} = \| A \|_2 .$$

Therefore, the $\| \cdot \|_2$ matrix norm is unitarily invariant, with respect to left or right multiplication by a unitary matrix. This motivates use of $\| \cdot \|_2$ in some theoretical contexts, but also motivates attention to more readily computable approximations to $\| A \|_2$ in practical contexts. These are themes to be explored further hereafter.

By definition of the subordinate matrix norm induced by a vector norm, for $A \in \mathbb{C}^{n \times n}$, and $x \in \mathbb{C}^n$, we have

$$\max_{x \neq 0} \left\{ \frac{\| Ax \|}{\| x \|} \right\} = \| A \| = \max_{\|x\| = 1} \| Ax \| .$$

In particular, for the $\| \cdot \|_2$ norms, this becomes

$$\max_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} = \| A \|_2 = \max_{\|x\|_2 = 1} \| Ax \|_2 .$$

We have shown above that, for nonsingular $A \in \mathbb{C}_n^{n \times n}$ and $x \in \mathbb{C}^n$, we also have

$$\min_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} = \| A^{-1} \|_2^{-1} = \min_{\|x\|_2 = 1} \| Ax \|_2 .$$

Therefore, for $x \neq 0$ we can write

$$\| A^{-1} \|_2^{-1} \leq \| Ax \|_2 / \| x \|_2 \leq \| A \|_2;$$

and, for $\| x \|_2 = 1$,

$$\| A^{-1} \|_2^{-1} \leq \| Ax \|_2 \leq \| A \|_2 .$$

These bounds are sharp. The upper bound is just the compatibility condition for the vector and matrix norms. At this point, we wish to extend the lower bound for any

vector norm and the subordinate matrix norm. The purpose in doing so is to highlight a crucial step in the argument, for our later purposes.

Because A is square and nonsingular, for every $y \in \mathbb{C}^n$, there is a unique $x \in \mathbb{C}^n$, namely $x = A^{-1}y$, such that $Ax = y$. Conversely, for every $x \in \mathbb{C}^n$, there is a unique $y \in \mathbb{C}^n$, namely $y = Ax$, such that $A^{-1}y = x$. It follows that $y \neq 0 \Leftrightarrow x \neq 0$ and

$$\| A^{-1} \| = \max_{y \neq 0} \left\{ \frac{\| A^{-1}y \|}{\| y \|} \right\} = \max_{x \neq 0} \left\{ \frac{\| x \|}{\| Ax \|} \right\} = \left[\min_{x \neq 0} \left\{ \frac{\| Ax \|}{\| x \|} \right\} \right]^{-1},$$

thence

$$\min_{x \neq 0} \left\{ \frac{\| Ax \|}{\| x \|} \right\} = \| A^{-1} \|^{-1}.$$

The key steps here are the recognition that $Ax = 0 \Leftrightarrow x = 0$ and that quantification over $y \neq 0$ and over $x \neq 0$ are equivalent because $R\{A\} = R\left\{A^{-1}\right\} = \mathbb{C}^n$. Therefore, for $x \neq 0$, we can write the sharp bounds

$$\| A^{-1} \|^{-1} \leq \| Ax \| / \| x \| \leq \| A \|,$$

and, for $\| x \| = 1$,

$$\| A^{-1} \|^{-1} \leq \| Ax \| \leq \| A \|.$$

For any compatible vector and matrix norms, we have $\| Ax \| \leq \| A \| \| x \|$ and also $\| x \| = \| A^{-1}Ax \| \leq \| A^{-1} \| \| Ax \|$. Therefore, the foregoing bounds are valid, but are not sharp for all A unless the matrix norm is subordinate to the vector norm. For the lower bounds, the essential hypothesis is that A is square and nonsingular.

We now wish to extend the foregoing from square to rectangular matrices. For $A \in \mathbb{C}^{n \times m}$, with $m \neq n$, and $x \in \mathbb{C}^m$, we have $Ax \in \mathbb{C}^n$, so there are two linear vector spaces and two vector norms involved. We restrict attention to companion norms in $\mathbb{C}^m$ and $\mathbb{C}^n$ differing only in the number of elements in the vectors, which allows notational simplifications through reliance upon implicit reference to the nature of the arguments involved to resolve any apparent ambiguities in expressions involving vector and matrix norms. We may then define the subordinate matrix norm induced by the pair of $\| \cdot \|$ vector norms by

$$\| A \| = \max_{x \neq 0} (\| Ax \| / \| x \|),$$

or equivalently,

$$\| A \| = \max_{\|x\| = 1} \| Ax \|.$$

The extensions to $\mathbb{C}^{n \times m}, m \neq n$, of (1) − (3) and (5) follow immediately. Since $\mathbb{C}^{n \times m}, m \neq n$, is a linear vector space but not a linear algebra, because products are not defined, the submultiplicativity property or consistency condition (4) is not meaningful if we focus on particular $m \neq n$. It is more productive to consider all m and n in $\mathbb{N}$ together, including square ($m = n$), column-rectangular ($m < n$), and row-rectangular ($m > n$) matrices simultaneously. Then $\| BA \| \leq \| B \| \| A \|$ is meaningful for $A \in \mathbb{C}^{n \times m}$ and $B \in \mathbb{C}^{\ell \times n}$ so $BA \in \mathbb{C}^{\ell \times m}$ is well-defined. There is one vector norm involved if $\ell = m = n$; there are two vector norms involved if any two of ℓ, m, and n are equal and distinct from the third; there are three vector norms involved if ℓ, m, and n are distinct from one another. There may be one, two, or three matrix norms involved. In this sense, the extended submultiplicativity property or

consistency condition (4) again follows. Since we include square matrices, (6) also holds: see further below.

4.4 Condition numbers

The formulae discussed above for $\| \cdot \|_1$, $\| \cdot \|_\infty$, and $\| \cdot \|_2$ in the square matrix context extend straightforwardly to the rectangular matrix context, including the derivations associated with the latter. Index sets for summations and maximizations are simply adjusted in obvious ways. We shall be concerned primarily with nonsingular square matrices and maximal-rank column-rectangular matrices, but with a focus on nearly singular and nearly rank deficient such matrices, which reflects near linear dependence of the columns thereof. Nonsingular square matrices A and maximal-rank column-rectangular matrices A have trivial nullspaces, $N\{A\} = \{0\}$ so $Ax = 0 \iff x = 0$. Singular square matrices, rank-deficient column-rectangular matrices and row-rectangular matrices have nontrivial nullspaces, so there are nonzero x such that $Ax = 0$. Initially, we employ the $\| \cdot \|_2$ vector and matrix norms, for reasons which will emerge shortly. Many, but not all, aspects of the discussion extend naturally to other norms.

The subset $\mathbb{C}_n^{n\times n}$ of nonsingular matrices is open and dense in $\mathbb{C}^{n\times n}$, and the subset of singular matrices $\mathbb{C}^{n\times n} - \mathbb{C}_n^{n\times n}$ consists of surfaces within the n^2-dimensional normed linear vector space (and linear algebra). Define the condition number of $A \in \mathbb{C}_n^{n\times n}$, for the $\| \cdot \|_2$ matrix norm, by

$$\kappa_2(A) = \| A \|_2 \| A^{-1} \|_2 .$$

It is common to elide the subscript on κ if only the $\| \cdot \|_2$ matrix norm is involved, but this is not the case here so we retain it. Omission of subscripts will signify a generic norm, usually a subordinate matrix norm induced by a vector norm. Observe that $\kappa_2(A^{-1}) = \kappa_2(A)$ and that $\kappa_2(\alpha A) = \kappa_2(A)$, for $\alpha \neq 0$. Observe also that

$$1 = \| I \|_2 = \| A^{-1} A \|_2 \leq \| A^{-1} \|_2 \| A \|_2 = \kappa_2(A).$$

From $\| A \|_2 \leq \sqrt{\| A \|_1 \| A \|_\infty}$ and $\| A^{-1} \|_2 \leq \sqrt{\| A^{-1} \|_1 \| A^{-1} \|_\infty}$, we obtain

$$\kappa_2(A) \leq \sqrt{\kappa_1(A)\kappa_\infty(A)}.$$

It is easily shown that $\| A^* A \|_2 = \| A \|_2^2$, thence

$$\kappa_2(A^* A) = \kappa_2(A)^2 = \kappa_2(A^*)^2 = \kappa_2(A A^*).$$

From earlier results, we identify

$$\kappa_2(A) = \| A \|_2 \| A^{-1} \|_2 = \max_{x\neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} / \min_{x\neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\}.$$

For our later purposes, it is most illuminating to rewrite this in the form

$$\min_{x\neq 0} \left\{ \frac{\| Ax \|_2}{\| A \|_2 \| x \|_2} \right\} = \kappa_2(A)^{-1} = \min_{\|x\|_2 = 1} \left\{ \frac{\| Ax \|_2}{\| A \|_2} \right\}.$$

By earlier results, a corresponding relationship is valid for any vector norm and the induced subordinate matrix norm, for square nonsingular matrices A. This means

that the reciprocal of the condition number (usually abbreviated as the reciprocal condition number) is a scale-invariant (but not scaling-invariant!) measure of near linear dependence of the columns of A, for $\kappa_2(A) \gg 1$. By scale-invariant, I mean invariant if A is replaced by αA, for $\alpha \neq 0$; by not scaling-invariant, I mean not ordinarily invariant if A is replaced by AS^{-1}, for $S \neq \alpha^{-1} I$. We have anticipated previously that for a suitable S we may have $\kappa_2(AS^{-1}) \ll \kappa_2(A)$ if the norms of the columns of A are of disparate sizes.

The reciprocal condition number is also the relative distance between A and the nearest singular matrix B: that is,

$$\| A - B \|_2 / \| A \|_2 = \kappa_2(A)^{-1}.$$

If $A \to B$, we see that $\kappa_2(A) \to \infty$. A is said to be ill-conditioned if $\kappa_2(A) \gg 1$; otherwise, well-conditioned—the precise characterization being problem dependent. The nonsingular linear equation $Ac = b$ is well-posed, with unique solution $\hat{c} = A^{-1}b$. $\kappa_2(A)$ is a measure of the sensitivity of A^{-1} to perturbations of A, and of $\hat{c}$ to perturbations of A and/or b. Ill-conditioning of A, and by extension $Ac = b$, corresponds to near singularity and to sensitivity to small perturbations, be they errors or uncertainties. The singular linear equation $Bc = b$ is ill-posed, with no solution unless b is in the range of B, $R\{B\}$; in which case, there is an affine subspace of solutions parallel to the nullspace of B, $N\{B\}$. In this context, it is customary to regard $\kappa_2(B)$ as undefined: see further below. Similar considerations apply for $\kappa_1(A)$ and $\kappa_\infty(A)$.

To extend the foregoing to rectangular and singular matrices, the inverse A^{-1} is replaced by the Moore-Penrose pseudoinverse A^+. For $A \in \mathbb{C}_r^{n\times m}$, $1 \leq r \leq \min(m, n)$, A^+ is the unique $X \in \mathbb{C}_r^{m\times n}$ satisfying the Moore-Penrose conditions

(1) $AXA = A$,
(2) $XAX = X$,
(3) $(AX)^* = AX$,
(4) $(XA)^* = XA$.

It is easily verified that $(A^*)^+ = (A^+)^*$, and $(A^+)^+ = A$; and, for A square and nonsingular, $A^+ = A^{-1}$. More to the point, A^+ is the unique member of $\mathbb{C}_r^{m\times n}$ such that, for all $b \in \mathbb{C}^n$, $\hat{c} = A^+ b$ is the unique minimizer of $\| \check{c} \|_2$ over the set of all minimizers $\check{c}$ of $\| b - Ac \|_2$: a single point for $N\{A\} = \{0\}$, and $N\{A\}$ or an affine subspace parallel to $N\{A\}$ for $N\{A\} \neq \{0\}$. Since A^+ is naturally associated with $\| \cdot \|_2$, it is customary to focus on $\kappa_2(A) = \| A \|_2 \| A^+ \|_2$ in this context: see below.

Consider the maximal-rank column-rectangular case, $A \in \mathbb{C}_m^{n\times m}$, $m < n$. From the normal equations, $A^*A\hat{c} = A^*b$, we obtain $\hat{c} = (A^*A)^{-1}A^*b$, thence $A^+ = (A^*A)^{-1}A^*$. More cogently numerically, given the QR factorization $AP = \hat{Q}\hat{R}$, we argued previously that $\hat{c} = P\hat{R}^{-1}\hat{Q}^*b$, thence $A^+ = P\hat{R}^{-1}\hat{Q}^*$. It is easily verified that the Moore-Penrose conditions are satisfied. Since $\kappa_2(A^*A) = \kappa_2(A)^2$, and we shall argue below that $\kappa_2(A) = \kappa_2(\hat{R})$, the QR factorization is numerically preferable to the normal equations; however, the normal equations involve less arithmetic.

Taking for granted the anticipated extension of earlier results to this case, we obtain

$$\| A \|_2 = \sqrt{\rho(A^*A)}$$

and

$$\| A^+ \|_2 = \sqrt{\rho(A^+(A^+)^*)} = \sqrt{\rho((A^*A)^{-1})},$$

from which we find that

$$\kappa_2(A) = \| A \|_2 \| A^+ \|_2 = \max_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\} / \min_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| x \|_2} \right\}.$$

We again rewrite this as

$$\min_{x \neq 0} \left\{ \frac{\| Ax \|_2}{\| A \|_2 \| x \|_2} \right\} = \kappa_2(A)^{-1} = \min_{\|x\|_2 = 1} \left\{ \frac{\| Ax \|_2}{\| A \|_2} \right\}$$

and identify the reciprocal of $\kappa_2(A)$ as a scale-invariant measure of near linear dependence, for $\kappa_2(A) \gg 1$. Thus, for $\| \cdot \|_2$ norms, we have parallel results for the square nonsingular matrix and the maximal-rank column-rectangular matrix cases.

The earlier arguments extending these results for square nonsingular matrices to other norms fail to extend them for maximal-rank column-rectangular matrices. For the $\| \cdot \|_2$ norms, matrices with nontrivial nullspace can be addressed using the extended Rayleigh Principle by restriction to the orthogonal complement of the nullspace. For other norms, it is not clear how to most usefully extend the notion of condition number, even to maximal-rank column-rectangular matrices. We shall not pursue these matters further.

Again one can identify the reciprocal of $\kappa_2(A)$ as the relative distance between A and the nearest matrix B of lower rank:

$$\| A - B \|_2 / \| A \|_2 = \kappa_2(A)^{-1}.$$

If $A \to B$, we see that $\kappa_2(A) \to \infty$; however, $\kappa_2(B)$ is well-defined, so $\kappa_2(A)$ is not continuous at B. In particular, if A is a maximal-rank column-rectangular matrix with $\kappa_2(A) \gg 1$, then A is nearly rank deficient, so its columns are nearly linearly dependent, and $\hat{c} = A^+ b$ may be unduly sensitive to small perturbations of A or b.

Consider $A \in \mathbb{C}_m^{n \times m}$, $m < n$, and the decomposition/factorization

$$AP = QR = \hat{Q}\hat{R}.$$

From $Q^*Q = I$ and $\hat{Q}^*\hat{Q} = I$, we infer that

$$P^*A^*AP = (AP)^*(AP) = R^*R = \hat{R}^*\hat{R}.$$

A^*A is positive definite, so its eigenvalues are all positive and coincide with those of P^*A^*AP. It follows that

$$\begin{aligned} \| A \|_2 &= \| AP \|_2 = \| R \|_2 = \| \hat{R} \|_2, \\ \| A^+ \|_2 &= \| (AP)^+ \|_2 = \| R^+ \|_2 = \| \hat{R}^{-1} \|_2, \end{aligned}$$

and

$$\kappa_2(A) = \kappa_2(AP) = \kappa_2(R) = \kappa_2(\hat{R}).$$

Therefore, we can focus on $\kappa_2(\hat{R}) = \| \hat{R} \|_2 \| \hat{R}^{-1} \|_2$, the condition number of a regularly upper triangular square matrix.

Let $D \in \mathbb{R}_m^{m\times m}$ be any positive definite diagonal matrix. For $A \in \mathbb{C}_m^{n\times m}$, $2 \le m \le n$, let $S \in \mathbb{R}_m^{m\times m}$ be the positive definite diagonal matrix defined by $e_k^* S e_k = \| Ae_k \|_2$, $1 \le k \le m$. It can be shown that

$$\min_D \kappa_2(D^{-1}A^*AD^{-1}) \le \kappa_2(S^{-1}A^*AS^{-1}) \le m \min_D \kappa_2(D^{-1}A^*AD^{-1}),$$

thence

$$\min_D \kappa_2(AD^{-1}) \le \kappa_2(AS^{-1}) \le \sqrt{m} \min_D \kappa_2(AD^{-1}).$$

This is of particular interest for small to moderate $m \ll n$, and motivates the standard scaling strategy: see Golub and Van Loan [9]. If A has columns of disparate sizes, we can usually expect $\kappa_2(AS^{-1}) \ll \kappa_2(A)$. Recall, however, that $\kappa_2(\alpha A) = \kappa_2(A)$, for $\alpha \neq 0$. Note the implications for the normal equations. The example $A = \begin{bmatrix} S \\ 0 \end{bmatrix}$ is instructive, though not representative, especially in our context. We anticipate that the size of $\kappa_2(AS^{-1})^{-1}$ will provide a more reliable measure of near linear dependence of the columns of A than $\kappa_2(A)^{-1}$, because the latter may be small due only to disparate sizes of these columns. Our primary concern is detecting near linear dependence, rather than the condition number per se.

Two brief digressions are in order at this point, before returning to the main argument. First, recall that during our preliminary remarks about "ill-determination" and "ill-conditioning," we introduced an inequality whose proof was deferred for later consideration, to which we now turn. Let $\hat{c}$ be the unique minimizer of $\| b - Ac \|_2$, for a maximal-rank column-rectangular A, and let $\check{c}$ be a putative approximation thereto, so $\| b - A\check{c} \|_2 \ge \| b - A\hat{c} \|_2$. We see that

$$(b - A\check{c}) = (b - A\hat{c}) + A(\hat{c} - \check{c})$$

thence

$$\| b - A\check{c} \|_2 \le \| b - A\hat{c} \|_2 + \| A(\check{c} - \hat{c}) \|_2 .$$

The compatibility inequality yields

$$\| A(\check{c} - \hat{c}) \|_2 \le \| A \|_2 \| \check{c} - \hat{c} \|_2 .$$

It follows that, for $\check{c} \neq \hat{c}$ and $\hat{c} \neq 0$,

$$\frac{\| A(\check{c} - \hat{c}) \|_2}{\| A \|_2 \| \hat{c} \|_2} \le \left\{ \frac{\| A(\check{c} - \hat{c}) \|_2}{\| A \|_2 \| \check{c} - \hat{c} \|_2} \right\} \left\{ \frac{\| \check{c} - \hat{c} \|_2}{\| \hat{c} \|_2} \right\},$$

and we have the sharp bounds

$$\kappa_2(A)^{-1} \le \frac{\| A(\check{c} - \hat{c}) \|_2}{\| A \|_2 \| \check{c} - \hat{c} \|_2} \le 1.$$

We may therefore write

$$0 \le \frac{[\| b - A\check{c} \|_2 - \| b - A\hat{c} \|_2]}{\| A \|_2 \| \hat{c} \|_2} \le \frac{\| A(\check{c} - \hat{c}) \|_2}{\| A \|_2 \| \hat{c} \|_2} \le \frac{\| \check{c} - \hat{c} \|_2}{\| \hat{c} \|_2},$$

as advertised earlier. If $\kappa_2(A) \gg 1$, the columns of A are nearly linearly dependent, so there are $\check{c}$ such that $\| A(\check{c} - \hat{c}) \|_2 \ll \| A \|_2 \| \check{c} - \hat{c} \|_2$, thence

$$\| A(\check{c} - \hat{c}) \|_2 / \| A \|_2 \| \hat{c} \|_2 \ll \| \check{c} - \hat{c} \|_2 / \| \hat{c} \|_2 .$$

This means that we may well have $\| b - A\check{c} \|_2 \approx \| b - A\hat{c} \|_2$, for moderately large $\| \check{c} - \hat{c} \|_2 / \| \hat{c} \|_2$. There are also $\check{c}$ such that $\| A(\check{c} - \hat{c}) \|_2 \approx \| A \|_2 \| \check{c} - \hat{c} \|_2$, thence

$$\| A(\check{c} - \hat{c}) \|_2 / \| A \|_2 \| \hat{c} \|_2 \approx \| \check{c} - \hat{c} \|_2 / \| \hat{c} \|_2 .$$

This allows for the possibility that $\| b - A\check{c} \|_2 \gg \| b - A\hat{c} \|_2$, though it does not guarantee this.

Second, if we have the QR decomposition/factorization $AP = QR = \hat{Q}\hat{R}$ and take any unitary diagonal matrix

$$U = \begin{bmatrix} \hat{U} & 0 \\ 0 & \check{U} \end{bmatrix},$$

then $AP = (QU)(U^*R) = (\hat{Q}\hat{U})(\hat{U}^*\hat{R})$ is also such a decomposition/factorization. By choosing $\check{U} = I$ and $\hat{U} = Diag(sgn(e_k^*\hat{R}e_k))$, we can arrange that $\hat{U}^*\hat{R}$ has real, positive diagonal elements, for $A \in \mathbb{C}_m^{n\times m}$. We identify

$$P^*(A^*A)P = (\hat{U}^*\hat{R})^*(\hat{U}^*\hat{R})$$

as the Cholesky factorization of the positive definite matrix $P^*(A^*A)P$. By construction, the modified Gram–Schmidt process automatically produces this standard QR factorization. In general, the algorithm detailed in the last section does not produce the standard QR factorization, but could be augmented to do so by incorporating the relevant U at the end. If we are only interested in solving

$$\hat{R}(P^*\tilde{c}) = \hat{Q}^*b = \hat{d},$$

or subsystems thereof, there is no need to do so since,

$$(\hat{U}^*\hat{R})(P^*\tilde{c}) = (\hat{Q}\hat{U})^*b = \hat{U}^*\hat{d}.$$

However, observe that if we chose to actually triangularize the augmented matrix $[A \;\; b]$ using an extra Householder matrix, we could also choose U to obtain

$$\begin{bmatrix} \hat{R} & \hat{d} \\ 0 & \| \check{d} \|_2 \end{bmatrix}$$

The extra Householder matrix and associated U are of no interest when we invoke this observation later.

Consider $A \in \mathbb{C}_m^{n\times m}$, $m < n$, and the QR factorization $AP = \hat{Q}\hat{R}$, so $\hat{R} \in \mathbb{C}_m^{m\times m}$ is regularly upper triangular, as is $\hat{R}^{-1}$. Recall that we have

$$| e_k^*\hat{R}^{-1}e_k | = | e_k^*\hat{R}e_k |^{-1}, 1 \le k \le m.$$

Assume that P has been chosen so that

$$| e_1^*\hat{R}e_1 | \ge | e_2^*\hat{R}e_2 | \ge \cdots \ge | e_m^*\hat{R}e_m | > 0,$$

so

$$| e_m^*\hat{R}e_m |^{-1} \ge | e_{m-1}^*\hat{R}e_{m-1} |^{-1} \ge \cdots \ge | e_1^*\hat{R}e_1 |^{-1} > 0,$$

thence

$$| e_m^*\hat{R}^{-1}e_m | \ge | e_{m-1}^*\hat{R}^{-1}e_{m-1} | \ge \cdots \ge | e_1^*\hat{R}^{-1}e_1 | > 0.$$

We then obtain

$$| e_1^*\hat{R}e_1 | = \| \hat{R}e_1 \|_2 \le \| \hat{R} \|_2$$

and

$$| e_m^* \hat{R}^{-1} e_m | = \| (\hat{R}^{-1})^* e_m \|_2 \leq \| (\hat{R}^{-1})^* \|_2,$$

so

$$| e_m^* \hat{R} e_m |^{-1} \leq \| \hat{R}^{-1} \|_2 .$$

It follows that

$$| e_1^* \hat{R} e_1 | / | e_m^* \hat{R} e_m | \leq \kappa_2(\hat{R}) = \kappa_2(A)$$

or

$$| e_m^* \hat{R} e_m | / | e_1^* \hat{R} e_1 | \geq \kappa_2(\hat{R})^{-1} = \kappa_2(A)^{-1}.$$

Examples have been constructed (see Björck [3] or Golub and Van Loan [9]) for which this lower bound on the condition number is of order one and the condition number is large. Practical experience suggests that such disparity is rare; while this lower bound may not provide a good approximation to a large condition number, it will ordinarily also be large, providing a reasonably reliable indicator of ill-conditioning, but this cannot be guaranteed. A large lower bound on the condition number yields a small upper bound on the reciprocal condition number. We might fail to diagnose near linear dependence, but will not misdiagnose it, for a specified threshold. Scaling, pivoting, and narrow or dual regularization would arrange that

$$| e_m^* \hat{R} e_m | \geq \tau | e_1^* \hat{R} e_1 |,$$

or

$$| e_m^* \hat{R} e_m | / | e_1^* \hat{R} e_1 | \geq \tau,$$

so

$$| e_1^* \hat{R} e_1 | / | e_m^* \hat{R} e_m | \leq \tau^{-1}.$$

Without scaling, pivoting, or regularization, the diagonal elements of $\hat{R}$ may not provide a reliable indicator of near rank deficiency, thence near linear dependence.

At this point, we shall briefly examine connections among redundance, relevance, and conditioning. The operational definition of redundance involves the scaling, pivoting, and regularization strategies employed, and the details thereof. We focus on interpretations of the pivoting strategy; the scaling strategy affects the outcome thereof, and the regularization strategy affects the consequences thereof.

Assume, for the moment, that the standard scaling and pivoting strategies are used, without regularization. At the kth stage, we seek to maximize $| e_k^* \hat{R} e_k |$ among available alternatives. We now identify the alternatives as the norm of the residuals when available columns are approximated using columns chosen at previous stages. One interpretation then is that we seek to minimize the collective relevance of the previous columns in approximating the next one. If all columns were initially scaled to have unit length, these residual norms are the sines of the angles between the candidate columns and the span of the previous ones, so another interpretation is that one is seeking to maximize the corresponding angle. If the columns were initially scaled to have the same length, but not unit length, the residual norms would be proportional to the sines, so the same interpretation is apt. If all columns were not initially scaled to have unit or equal length, we would no longer be maximizing the angle. Columns of disparate sizes may alter the choice for $| e_k^* \hat{R} e_k |$. Smaller columns may

be deemed more redundant than appropriate, and larger columns may be deemed less redundant than appropriate. The alternate scaling strategy given earlier uses this to accommodate aspects of the problems of interest.

A third interpretation is that we seek to minimize a lower bound on the condition number of the submatrix consisting of the columns chosen through the kth stage. This sounds peculiar when so stated. However, if this lower bound is a reasonably reliable indicator of potential ill-conditioning, thence near linear dependence, it sounds more sensible. The first two interpretations are concrete; the third is somewhat more tenuous. All three interpretations share elements of the intuitive significance of the word "redundancy."

Once the kth column has been chosen, regularization may increase $\mid e_k^* \hat{R} e_k \mid$ by redefining the task at hand to include a penalty term intended to reduce the adverse impact of redundancy on the solution.

Adaptive selection of $m^{(\ell)}$ is reasonably straightforward when only scaling and pivoting are involved, and also when regularization is incorporated. Using scaling and pivoting, a threshold τ, with $0 < \tau \ll 1$, determines an effective rank as the largest k such that $1 \leq k \leq \min(\ell, M)$ and $\mid e_k^* \hat{R} e_k \mid \geq \tau \mid e_1^* \hat{R} e_1 \mid$. This effective rank is a measure of redundancy, and not a reliable estimate of rank per se. Recall that τ is an upper bound for the relative distance to the nearest rank-deficient matrix and a reasonably reliable threshold for declaring near linear dependence. A first approach is to take the initial candidate for $m^{(\ell)}$ as this effective rank and determine the basic least squares solution as though $m^{(\ell)}$ was the actual rank. A second approach is to take the initial candidate for $m^{(\ell)}$ as $\min(\ell, M)$ and determine the minimal least squares solution as though the effective rank was the actual rank. The basic solution approach disregards data regarded as redundant; the minimal solution approach retains all available data. Thereafter, the initial candidate for $m^{(\ell)}$ would be reduced as necessary to satisfy the constraint. For strongly nonlinear problems, I favor using small to moderate M and the basic solution approach. Redundant iterant data, especially older data, may be misleading. For weakly nonlinear (or linear) problems, somewhat larger M and the minimal solution approach may be useful.

When dual regularization is incorporated, we identify the threshold τ with the narrow regularization parameter. The broad regularization parameter μ may be assigned or chosen adaptively. The initial candidate for $m^{(\ell)}$ is $\min(\ell, M)$, and this is reduced as necessary to satisfy the constraint. Recall that this bridges between the basic and minimal solutions without specifically invoking an effective rank. Near rank deficiency is accommodated through the penalty term. Observe that the ingredients for a basic or minimal solution approach are at hand for $\mu = 0$: narrow regularization.

4.5 Addendum

The code which evolved during my work with the Extrapolation Algorithm thru the 1970s was based on ideas akin to those outlined above. The ideas to be introduced in the remainder of this section are of more recent vintage. Many of the theoretical results to follow are familiar, but others are less familiar, and may be of independent interest. Later sections are not dependent on this material. I have not had, and will not have, an opportunity to explore their potential practical utility.

We begin by establishing notation and recording well-known inequalities involving the three vector norms of interest. For $x \in \mathbb{C}^n$, define $\mid x \mid \in \mathbb{C}^n$ by $e_i^* \mid x \mid = \mid e_i^* x \mid$ and $e \in \mathbb{C}^n$ by $e_i^* e = 1$. We see that $\| \mid x \mid \|_1 = \| x \|_1$, $\| \mid x \mid \|_\infty = \| x \|_\infty$ and $\| \mid x \mid \|_2 = \| x \|_2$. The triangle inequality (for complex numbers) can be expressed as

$$\mid e^* x \mid \leq e^* \mid x \mid = \| x \|_1 .$$

For $x, y \in \mathbb{C}^n$ the Cauchy–Schwarz inequality can be expressed as

$$\mid x^* y \mid \leq \mid x \mid^* \mid y \mid \leq \| x \|_2 \| y \|_2 .$$

The latter is a special case of the Hölder inequality, which also has the limiting cases

$$\mid x^* y \mid \leq \mid x \mid^* \mid y \mid \leq \| x \|_1 \| y \|_\infty$$

and

$$\mid x^* y \mid \leq \mid x \mid^* \mid y \mid \leq \| x \|_\infty \| y \|_1,$$

which can be argued directly in elementary fashion. It follows that

$$\| x \|_2^2 = \mid x^* x \mid \leq \| x \|_1 \| x \|_\infty,$$

so

$$\| x \|_2 \leq \sqrt{\| x \|_1 \| x \|_\infty}.$$

For $x \in \mathbb{C}^n$, we have the three pairs of inequalities

$$\begin{aligned} \| x \|_2 &\leq \| x \|_1 \leq \sqrt{n} \| x \|_2, \\ \| x \|_\infty &\leq \| x \|_1 \leq n \| x \|_\infty, \end{aligned}$$

and

$$\| x \|_\infty \leq \| x \|_2 \leq \sqrt{n} \| x \|_\infty .$$

These inequalities are all sharp: satisfied as equalities for some x. Their proofs are straightforward; we shall record that for the first pair, which is the one most relevant for our later purposes, to make a subsidiary point. We obtain the upper bound from

$$\| x \|_1 = e^* \mid x \mid \leq \| e \|_2 \| x \|_2 = \sqrt{n} \| x \|_2,$$

which is satisfied as an equality for $\mid x \mid / \| x \|_\infty = e$. We obtain the lower bound from

$$\begin{aligned} \| x \|_1^2 &= \left(\sum_k \mid e_k^* x \mid \right)^2 = \left(\sum_i \mid e_i^* x \mid \right) \left(\sum_j \mid e_j^* x \mid \right), \\ &= \sum_k \mid e_k^* x \mid^2 + \sum_{i \neq j} \mid e_i^* x \mid \mid e_j^* x \mid, \\ &= \| x \|_2^2 + 2 \sum_{i<j} \mid e_i^* x \mid \mid e_j^* x \mid, \end{aligned}$$

so $\| x \|_2^2 \leq \| x \|_1^2$, which is satisfied as an equality for $\mid x \mid / \| x \|_\infty = e_\ell$, $1 \leq \ell \leq n$. For moderate to large n, we observe that the lower bound is loose in the sense that

$\| x \|_1$ is comparable to $\| x \|_2$ only for those special x such that $| x | / \| x \|_\infty \approx e_\ell$, $1 \le \ell \le n$, and that the upper bound is tight in the sense that $\| x \|_1$ is comparable to $\sqrt{n} \| x \|_2$ for most x not such that $| x | / \| x \|_\infty \approx e_\ell$, $1 \le \ell \le n$: that is, not special. The second and third pairs are also sharp for the corresponding x, with the upper bound tight and lower bound loose. We then obtain the complementary pairs of inequalities

$$\frac{1}{\sqrt{n}} \| x \|_1 \le \| x \|_2 \le \| x \|_1,$$
$$\frac{1}{\sqrt{n}} \| x \|_1 \le \| x \|_\infty \le \| x \|_1,$$

and

$$\frac{1}{\sqrt{n}} \| x \|_2 \le \| x \|_\infty \le \| x \|_2,$$

which are sharp with tight lower and loose upper bounds. Observe that we have

$$\| x \|_\infty \le \| x \|_2 \le \sqrt{\| x \|_1 \| x \|_\infty} \le \| x \|_1,$$

and these inequalities are satisfied as equalities for $| x | / \| x \|_\infty = e_\ell$, $1 \le \ell \le n$. By the foregoing, the lower bound on $\| x \|_2$ and the rightmost upper bound are loose; the leftmost upper bound may be more representative.

For $A \in \mathbb{C}^{n \times n}$, the standard argument above that $\| A \|_2 \le \sqrt{\| A \|_1 \| A \|_\infty}$ is simple and elegant, but not very informative. Using the arithmetic-geometric mean inequality, we also have

$$\| A \|_2 \le \sqrt{\| A \|_1 \| A \|_\infty} \le \frac{1}{2} [\| A \|_1 + \| A \|_\infty] \le \max \{\| A \|_1, \| A \|_\infty\}.$$

These inequalities are sharp, satisfied as equalities for $A = I$. The arithmetic and geometric means of $\| A \|_1$ and $\| A \|_\infty$satisfy (1) and (2); the arithmetic mean satisfies (3), but not (4); the geometric mean satisfies (4), but not (3). Thus, neither mean defines a matrix norm. However,

$$\| A \|_{II} := \max \{\| A \|_1, \| A \|_\infty\}$$

satisfies (1)–(4) and (6), thence defining a normalized matrix norm. $\| A \|_{II}$ is compatible with the $\| \cdot \|_1$, $\| \cdot \|_\infty$, and $\| \cdot \|_2$ vector norms, but the compatibility inequalities (5) are not sharp for all A. We shall record an elementary and lengthier, but more informative, proof of the $\| A \|_2 \le \sqrt{\| A \|_1 \| A \|_\infty}$ inequality. We shall then extend the discussion to counterpart lower bounds for $\| A \|_2$. In the course of doing so, we shall encounter two other matrix norms which are readily computable and compatible with the $\| \cdot \|_2$ vector norm, thence providing upper bounds for $\| A \|_2$.

For $A \in \mathbb{C}^{n\times n}$ and $x \in \mathbb{C}^n$, so $Ax \in \mathbb{C}^n$, we obtain

$$\begin{aligned}
\| Ax\|_2^2 &= \sum_i \mid \sum_j (e_i^* A e_j)(e_j^* x) \mid^2, \\
&\leq \sum_i \left\{ \sum_j \mid e_i^* A e_j \mid\mid e_j^* x \mid \right\}^2, \\
&\leq \sum_i \left\{ \sum_j \mid e_i^* A e_j \mid^{\frac{1}{2}} (\mid e_i^* A e_j \mid^{\frac{1}{2}} \mid e_j^* x \mid) \right\}^2, \\
&\leq \sum_i \left\{ \sum_j \mid e_i^* A e_j \mid \sum_k \mid e_i^* A e_k \mid\mid e_k^* x \mid^2 \right\}, \\
&\leq \max_\ell \sum_j \mid e_\ell^* A e_j \mid \sum_i \sum_k \mid e_i^* A e_k \mid\mid e_k^* x \mid^2, \\
&\leq \| A \|_\infty \sum_k \sum_i \mid e_i^* A e_k \mid\mid e_k^* x \mid^2, \\
&\leq \| A \|_\infty \max_\ell \sum_i \mid e_i^* A e_\ell \mid \sum_k \mid e_k^* x \mid^2, \\
&\leq \| A \|_\infty \| A \|_1 \| x\|_2^2,
\end{aligned}$$

thence

$$\| Ax \|_2 \leq \sqrt{\| A \|_1 \| A \|_\infty}\, \| x \|_2$$

and

$$\| A \|_2 = \max_{x\neq 0} (\| Ax \|_2 / \| x \|_2) \leq \sqrt{\| A \|_1 \| A \|_\infty}.$$

In the chain of inequalities bounding $\| Ax\|_2^2$, the first (triangle inequality), third (Cauchy–Schwarz inequality), fourth, and sixth (limiting Hölder inequality) are sharp, but usually increase the upper bound, perhaps substantially. The second, fifth, and seventh do not increase the upper bound. The geometric mean of $\| A \|_1$ and $\| A \|_\infty$ may be significantly larger than $\| A \|_2$.

In the foregoing, all indices in summations and maximizations implicitly range from 1 to n. To extend the argument to rectangular matrices $A \in \mathbb{C}^{n\times m}$, $m \neq n$, it will suffice to adjust the ranges of all indices in obvious ways. We shall focus primarily on the square matrix case $m = n$ hereafter, but will flag one further point regarding the rectangular case $m \neq n$. We introduce the notation

$$\| A \|_F = \left\{ \sum_i \sum_j \mid e_i^* A e_j \mid^2 \right\}^{\frac{1}{2}}$$

and

$$\| A \|_{ii} = \sqrt{m} \max_k \| Ae_k \|_2,$$

for $A \in \mathbb{C}^{n\times m}$, anticipating subsequent verification that these are matrix norms compatible with the $\| \cdot \|_2$ vector norm. For the Frobenius norm $\| A \|_F$, index ranges are accommodated implicitly, and we see that $\| A^* \|_F = \| A \|_F$. For $\| A \|_{ii}$, index ranges enter explicitly, and we see that

$$\| A^* \|_{ii} = \sqrt{n} \max_{\ell} \| A^* e_\ell \|_2 \ .$$

Minor modifications of results to follow, derived for $m = n$, are needed to accommodate $m \neq n$; the task is left as an exercise for the interested reader.

Returning to the square matrix case of primary interest later, and to the earlier result

$$\| Ax \|_2^2 \leq \sum_i \left\{ \sum_j | e_i^* A e_j | \, | e_j^* x | \right\}^2 ,$$

we invoke the Cauchy–Schwarz inequality to establish that

$$\| Ax \|_2^2 \leq \sum_i \sum_j | e_i^* A e_j |^2 \sum_k | e_k^* x |^2 = \| A \|_F^2 \, \| x \|_2^2,$$

thereby obtaining the compatibility inequality (5), $\| Ax \|_2 \leq \| A \|_F \| x \|_2$, thence $\| A \|_2 \leq \| A \|_F$. For A in the linear algebra $\mathbb{C}^{n\times n}$, we observe that

$$A = \sum_i \sum_j (e_i^* A e_j) e_i e_j^*,$$

and identify $\| A \|_F$ as the $\| \cdot \|_2$ vector norm of the coordinate vector of A with respect to the standard basis $\left\{ e_i e_j^* \right\}$ for the linear vector space $\mathbb{C}^{n\times n}$, from which (1)–(3) follow. However, we have $\| I \|_F = \sqrt{n}$, so the normalization condition (6) is not satisfied. We see that

$$\| A \|_F^2 = \sum_j \| Ae_j \|_2^2 = \sum_i \| A^* e_i \|_2^2 = \| A^* \|_F^2,$$

thence

$$\| BA \|_F^2 = \sum_j \| BAe_j \|_2^2 = \sum_i \| A^* B^* e_i \|_2^2 = \| A^* B^* \|_F^2.$$

We observe first that if B, thence also B^*, is unitary then

$$\| BA \|_F^2 = \| A \|_F^2 = \| A^* \|_F^2 = \| A^* B^* \|_F^2,$$

so the Frobenius norm is unitarily invariant for left or right multiplication by a unitary matrix. Using the compatibility inequality, we observe second that

$$\| BA \|_F^2 \leq \| B \|_F^2 \sum_j \| Ae_j \|_2^2 = \| B \|_F^2 \, \| A \|_F^2,$$

thence

$$\| BA \|_F \leq \| B \|_F \| A \|_F,$$

so the consistency condition (4) is satisfied. Ergo, we identify $\| A \|_F$ as an unnormalized matrix norm compatible with the $\| \cdot \|_2$ vector norm.

The foregoing facts are familiar and recorded for use in the discussion to follow. It is also a familiar fact that $\| A \|_F$ is the norm induced on the linear vector space $\mathbb{C}^{n\times n}$ by the Frobenius inner product

$$< A \mid B >= trc(A^*B),$$

so

$$\| A\|_F^2 = trc(A^*A).$$

Therefore, $\| A \|_F$ is the square root of the sum of the eigenvalues of A^*A, thence usually significantly larger than $\| A \|_2$, the square root of the largest eigenvalue, especially for $A \in \mathbb{C}_n^{n\times n}$ with moderate to large n. The inequality $\| A \|_2 \leq \| A \|_F$ is sharp: satisfied as an equality for $A \in \mathbb{C}_1^{n\times n}$. Likewise, for $A \in \mathbb{C}^{n\times n}$, we see that $\| A \|_F \leq \sqrt{n} \| A \|_2$, which is also sharp: satisfied as an equality for $A = I$. We therefore have the counterpart inequalities

$$\frac{1}{\sqrt{n}} \| A \|_F \leq \| A \|_2 \leq \| A \|_F$$

and the complementary inequalities

$$\| A \|_2 \leq \| A \|_F \leq \sqrt{n} \| A \|_2 .$$

We now observe that

$$\| A\|_F^2 = \sum_j \| Ae_j\|_2^2 \leq n \max_k \| Ae_k\|_2^2,$$

thence

$$\| A \|_F \leq \sqrt{n} \max_k \| Ae_k \|_2 = \| A \|_{ii} .$$

We also have

$$\| A \|_F = \| A^* \|_F \leq \| A^* \|_{ii},$$

thence

$$\begin{aligned} \| A \|_F &\leq \min \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\}, \\ &\leq \sqrt{\| A \|_{ii} \| A^* \|_{ii}}, \\ &\leq \frac{1}{2}[\| A \|_{ii} + \| A^* \|_{ii}], \\ &\leq \max \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\}. \end{aligned}$$

In particular, from $\| A \|_F \leq \| A \|_{ii}$, we obtain

$$\| Ax \|_2 \leq \| A \|_{ii} \| x \|_2,$$

the compatibility inequality (5). It is readily apparent that (1) and (2) are satisfied. There is at least one j such that

$$\begin{aligned} \max_k \| (A + B)e_k \|_2 &= \| (A + B)e_j \|_2, \\ &\leq \| Ae_j \|_2 + \| Be_j \|_2, \\ &\leq \max_k \| Ae_k \|_2 + \max_\ell \| Be_\ell \|_2, \end{aligned}$$

so we see that

$$\| A + B \|_{ii} \leq \| A \|_{ii} + \| B \|_{ii},$$

and (3) is satisfied. However, we have $\| I \|_{ii} = \sqrt{n}$, so the normalization condition (6) is not satisfied. Consider

$$\| BA \|_{ii} = \sqrt{n} \max_k \| BAe_k \|_2 .$$

We observe first that if B is unitary then $\| BA \|_{ii} = \| A \|_{ii}$, so $\| \cdot \|_{ii}$ is unitarily invariant for left multiplication by a unitary matrix, but not, in general, invariant for right multiplication. As above, using the compatibility condition, we obtain

$$\max_k \| BAe_k \|_2 = \| BAe_j \|_2 \leq \| B \|_{ii} \| Ae_j \|_2 \leq \| B \|_{ii} \max_\ell \| Ae_\ell \|_2,$$

and we observe second that

$$\| BA \|_{ii} \leq \| B \|_{ii} \| A \|_{ii},$$

so the consistency condition (4) is satisfied. Ergo, we identify $\| A \|_{ii}$ as an unnormalized matrix norm compatible with the $\| \cdot \|_2$vector norm. Observe that we could define another such norm for A using $\| A^* \|_{ii}$ instead of, or in addition to, $\| A \|_{ii}$: for example, $\| A \|_{iv} = \max \{ \| A \|_{ii}, \| A^* \|_{ii} \}$.

With all this formalism in hand, we reach the crux of the matter. We have the upper bounds

$$\| A \|_2 \leq \sqrt{\| A \|_1 \| A \|_\infty} \leq \| A \|_{II}$$

and

$$\| A \|_2 \leq \| A \|_F \leq \| A \|_{ii},$$

thence

$$\| A \|_2 \leq \min \left\{ \sqrt{\| A \|_1 \| A \|_\infty}, \| A \|_F \right\}.$$

For $A = I$, we have $\| A \|_F = \| A \|_{ii} = \sqrt{n}$ and $\| A \|_1 = \| A \|_\infty = \sqrt{\| A \|_1 \| A \|_\infty} = 1$, so we see that $\| A \|_F > \sqrt{\| A \|_1 \| A \|_\infty}$. For $A = ee_1^* + e_1 e^* - e_1 e_1^*$, we have $\| A \|_F = \sqrt{2n-1}$ and $\| A \|_1 = \| A \|_\infty = \sqrt{\| A \|_1 \| A \|_\infty} = \| A \|_{ii} = n$, so we see that $\| A \|_F < \sqrt{\| A \|_1 \| A \|_\infty}$, because $n^2 - 2n + 1 = (n-1)^2 > 0$. For $A = ee^*$, we have $\| A \|_F = \| A \|_1 = \| A \|_\infty = \| A \|_{ii} = \sqrt{\| A \|_1 \| A \|_\infty} = \| A^* \|_{ii} = n$, so we see that $\| A \|_F = \sqrt{\| A \|_1 \| A \|_\infty}$. For the first and third example, $\| A \|_2$ achieves its best upper bound; for the second example, it does not. The issue of whether $\| A \|_F$ or $\sqrt{\| A \|_1 \| A \|_\infty}$ provides a better upper bound for $\| A \|_2$ is separable from that of whether that bound is close to $\| A \|_2$ and from that of whether the bound is tight or loose. From this upper bound perspective, $\| A \|_{II}$ and $\| A \|_{ii}$ are of little interest since as good or better readily computable bounds are available. They are of interest when the fact that they are norms is relevant—and for notational convenience. The point of the present discussion is to obtain counterpart lower bounds of interest especially for small to moderate n.

As a simple example, define $v \in \mathbb{C}^n$ by $e_j^* v = \| Ae_j \|_2$ Recall that

$$\| v \|_\infty \leq \| v \|_2 \leq \sqrt{n} \| v \|_\infty,$$

so

$$\max_k \| Ae_k \|_2 \leq \left\{ \sum_\ell \| Ae_\ell \|_2^2 \right\}^{\frac{1}{2}} \leq \sqrt{n} \max_k \| Ae_k \|_2,$$

thence

$$\frac{1}{\sqrt{n}} \| A \|_{ii} \leq \| A \|_F \leq \| A \|_{ii}$$

Recall further that these inequalities are sharp, and that the upper bound is tight while the lower bound is loose. For the complementary bounds

$$\| A \|_F \leq \| A \|_{ii} \leq \sqrt{n} \| A \|_F,$$

the lower bound is tight and the upper bound is loose. We also have

$$\frac{1}{\sqrt{n}} \| A^* \|_{ii} \leq \| A \|_F \leq \| A^* \|_{ii}$$

thence

$$\frac{1}{\sqrt{n}} \max \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\} \leq \| A \|_F \leq \min \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\},$$

and

$$\frac{1}{\sqrt{n}} \sqrt{\| A \|_{ii} \| A^* \|_{ii}} \leq \| A \|_F \leq \sqrt{\| A \|_{ii} \| A^* \|_{ii}}.$$

Since $\| A \|_F$ and $\| A \|_{ii}$ are equally readily calculable, these bounds per se are of limited interest in practice, but they provide a model for what follows.

By definition, we have

$$\| Ae_j \|_2 \leq \| A \|_2, \ \forall j,$$

and

$$\| A^* e_i \|_2 \leq \| A^* \|_2 = \| A \|_2, \ \forall i.$$

It follows that

$$\frac{1}{\sqrt{n}} \| A \|_{ii} = \max_k \| Ae_k \|_2 \leq \| A \|_2$$

and

$$\frac{1}{\sqrt{n}} \| A^* \|_{ii} = \max_\ell \| A^* e_\ell \|_2 \leq \| A \|_2 .$$

We then obtain from $\| A \|_2 \leq \| A \|_F$ and the foregoing that

$$\frac{1}{\sqrt{n}} \max \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\} \leq \| A \|_2 \leq \min \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\}$$

and

$$\frac{1}{\sqrt{n}} \sqrt{\| A \|_{ii} \| A^* \|_{ii}} \leq \| A \|_2 \leq \sqrt{\| A \|_{ii} \| A^* \|_{ii}}.$$

$\| A \|_2$ is not readily computable, but the bounds are readily computable, and for small to moderate n are comparable to one another.

For $x \in \mathbb{C}^n$, recall the sharp inequalities

$$\frac{1}{\sqrt{n}} \| x \|_1 \leq \| x \|_2 \leq \| x \|_1,$$

with the upper bound loose and lower bound tight. We have, for all j,

$$\frac{1}{\sqrt{n}} \parallel Ae_j \parallel_1 \leq \parallel Ae_j \parallel_2 \leq \parallel Ae_j \parallel_1,$$

$$\frac{1}{\sqrt{n}} \parallel Ae_j \parallel_1 \leq \parallel Ae_j \parallel_2 \leq \max_{\ell} \parallel Ae_\ell \parallel_1,$$

$$\frac{1}{\sqrt{n}} \parallel Ae_j \parallel_1 \leq \max_{k} \parallel Ae_k \parallel_2 \leq \parallel A \parallel_1,$$

$$\frac{1}{\sqrt{n}} \max_{j} \parallel Ae_j \parallel_1 \leq \frac{1}{\sqrt{n}} \parallel A \parallel_{ii} \leq \parallel A \parallel_1,$$

thence

$$\frac{1}{\sqrt{n}} \parallel A \parallel_1 \leq \frac{1}{\sqrt{n}} \parallel A \parallel_{ii} \leq \parallel A \parallel_1,$$

and

$$\parallel A \parallel_1 \leq \parallel A \parallel_{ii} \leq \sqrt{n} \parallel A \parallel_1 .$$

We also find that

$$\frac{1}{\sqrt{n}} \parallel A^* \parallel_1 \leq \frac{1}{\sqrt{n}} \parallel A^* \parallel_{ii} \leq \parallel A^* \parallel_1,$$

so

$$\frac{1}{\sqrt{n}} \parallel A \parallel_\infty \leq \frac{1}{\sqrt{n}} \parallel A^* \parallel_{ii} \leq \parallel A^* \parallel_\infty,$$

and

$$\parallel A \parallel_\infty \leq \parallel A^* \parallel_{ii} \leq \sqrt{n} \parallel A^* \parallel_\infty .$$

We then obtain

$$\frac{1}{\sqrt{n}} \sqrt{\parallel A \parallel_1 \parallel A \parallel_\infty} \leq \frac{1}{\sqrt{n}} \sqrt{\parallel A \parallel_{ii} \parallel A^* \parallel_{ii}} \leq \sqrt{\parallel A \parallel_1 \parallel A \parallel_\infty},$$

which, combined with $\parallel A \parallel_2 \leq \sqrt{\parallel A \parallel_1 \parallel A \parallel_\infty}$ and the foregoing, yields

$$\frac{1}{\sqrt{n}} \sqrt{\parallel A \parallel_1 \parallel A \parallel_\infty} \leq \parallel A \parallel_2 \leq \sqrt{\parallel A \parallel_1 \parallel A \parallel_\infty}.$$

For nonsingular A, we likewise have

$$\frac{1}{\sqrt{n}} \sqrt{\parallel A^{-1} \parallel_1 \parallel A^{-1} \parallel_\infty} \leq \parallel A^{-1} \parallel_2 \leq \sqrt{\parallel A^{-1} \parallel_1 \parallel A^{-1} \parallel_\infty},$$

so we infer that

$$\frac{1}{n} \sqrt{\kappa_1(A) \kappa_\infty(A)} \leq \kappa_2(A) \leq \sqrt{\kappa_1(A) \kappa_\infty(A)}.$$

We also obtain

$$\max \{ \parallel A \parallel_1, \parallel A \parallel_\infty \} \leq \max \left\{ \parallel A \parallel_{ii}, \parallel A^* \parallel_{ii} \right\} \leq \sqrt{n} \max \{ \parallel A \parallel_1, \parallel A \parallel_\infty \},$$

$$\parallel A \parallel_{II} \leq \max \left\{ \parallel A \parallel_{ii}, \parallel A^* \parallel_{ii} \right\} \leq \sqrt{n} \parallel A \parallel_{II},$$

and

$$\frac{1}{\sqrt{n}} \parallel A \parallel_{II} \leq \frac{1}{\sqrt{n}} \max \left\{ \parallel A \parallel_{ii}, \parallel A^* \parallel_{ii} \right\} \leq \parallel A \parallel_{II} .$$

The best available readily computable lower and upper bounds for $\| A \|_2$ derived above yield

$$\frac{1}{\sqrt{n}} \max \left\{ \| A \|_{ii}, \| A^* \|_{ii} \right\} \leq \| A \|_2 \leq \min\{ \| A \|_F, \sqrt{\| A \|_1 \| A \|_\infty} \}.$$

With the corresponding bounds for $\| A^{-1} \|_2$, we obtain our best available readily computable bounds for $\kappa_2(A)$. The other counterpart bounds derived above guarantee that these best bounds increase in tandem as the quantity being bounded increases, and indicate that these lower and upper bounds are comparable to one another for small to moderate n. The inequalities involved are sharp. We anticipate that the lower bounds are tight and the upper bounds are loose, for moderate n. For large n, the gap between counterpart lower and upper bounds increases, but the norms will tend to increase with n. However, dramatic increases in norms of inverses thence condition numbers, due to incipient near linear dependence, are what we seek to detect.

We now focus on the situation of primary interest hereafter. Let $T \in \mathbb{C}^{n \times n}$ be regularly upper triangular: that is, $e_k^* T e_k \neq 0$ and $e_i^* T e_j = 0$, for $i > j$. The ingredients needed to calculate $\| T \|_1, \| T \|_\infty = \| T^* \|_1, \| T \|_F, \| T \|_{ii}$ and $\| T^* \|_{ii}$ are the sums $\sum_{i=1}^{j} | e_i^* T e_j |$ and $\sum_{i=1}^{j} | e_i^* T e_j |^2$, for $1 \leq j \leq n$, and the sums $\sum_{j=i}^{n} | e_i^* T e_j |$ and $\sum_{j=i}^{n} | e_i^* T e_j |^2$, for $1 \leq i \leq n$. Note that it would suffice, for our purposes, to calculate $\frac{1}{\sqrt{n}} \| T \|_{ii}$ and $\frac{1}{\sqrt{n}} \| T^* \|_{ii}$.

Define, for $\tau \neq 0$,

$$T_+ = \begin{bmatrix} T & t \\ 0 & \tau \end{bmatrix},$$

which is also regularly upper triangular. To calculate $\| T_+ \|_1, \| T_+ \|_\infty = \| T_+^* \|_1$, $\| T_+ \|_F, \| T_+ \|_{ii}$ and $\| T_+^* \|_{ii}$, the additional ingredients needed are the sums $\sum_{i=1}^{n} | e_i^* t | + | \tau |$ and $\sum_{i=1}^{n} | e_i^* t |^2 + | \tau |^2$, and the sums $\sum_{j=i}^{n} | e_i^* T e_j | + | e_i^* t |$ and $\sum_{j=i}^{n} | e_i^* T e_j |^2 + | e_i^* t |^2$, for $1 \leq i \leq n$, together with $| \tau |$ and $| \tau |^2$. The foregoing contains the essence of a recursive or sequential algorithm for evaluating the relevant norms for all of the leading principal submatrices of T, thence of T and T_+. Details of the implementation are left to the interested reader.

It is easily verified that

$$T_+^{-1} = \begin{bmatrix} T^{-1} & -\tau^{-1}(T^{-1}t) \\ 0 & \tau^{-1} \end{bmatrix}.$$

Likewise, we have at hand the essence of a recursive or sequential algorithm for evaluating the relevant norms of the leading principal submatrices of T^{-1}, thence of T^{-1} and T_+^{-1}. While one could evaluate $T^{-1}t$, which will prove to be of interest in its own right, using T^{-1}, it would be preferable to obtain $T^{-1}t$ by solving $Tx = t$. If only the relevant norms are of interest, we need not store T^{-1} and can simply store the requisite ingredients for the next step in the process.

The upshot is that we can calculate the relevant best bounds for $\| T \|_2, \| T^{-1} \|_2$, thence $\kappa_2(T) = \| T \|_2 \| T^{-1} \|_2$. We can also obtain, in one fell swoop, these bounds for all of the leading principal submatrices of T, and for T_+. We anticipate that, for

moderate n, the best lower bound is a better estimate than the best upper bound; the geometric mean of the best (or the counterpart) lower and upper bounds might also be a reasonable candidate.

Recall the earlier derivation of the primitive lower bound $\mid e_1^* T e_1 \mid / \mid e_n^* T e_n \mid$ for $\kappa_2(T)$. Clearly, the best lower bound now at hand is at least as large and possibly significantly larger than the primitive lower bound. If the corresponding estimate for $\kappa_2(T)$ is above (below) some specified threshold, one might adaptively increase (decrease) the narrow regularization parameter at the next iteration. This would increase (decrease) the size of the penalty term used to control the impact of ill-conditioning and decrease (increase) the size of the condition number of the operative matrix. A parallel or alternative discussion in terms of the reciprocal condition number is immediate and may be preferable.

We shall now return to the original problem. We begin by recalling that the alternate scaling strategy involves scaling both A and b, and we seek the least squares solution of $\tilde{A}\tilde{c} = \tilde{b}$. Recall also that the standard scaling strategy usually involves scaling only A, but b could also be scaled. We assume hereafter that both A and b have been scaled so $\| \tilde{b} \|_2$ and $\| \tilde{A} e_j \|_2$ are expected to be comparable. Scaling controls the norms of T and T_+, so the norms of T^{-1} and T_+^{-1} primarily determine $\kappa_2(T)$ and $\kappa_2(T_+)$, though these too are mollified by scaling.

Using our previous notation, we first identify $T = \hat{R}, t = \hat{d}$, and $\tau = \| \check{d} \|_2$. Ideally, $\kappa_2(T)$ is moderately small and $\kappa_2(T_+)$ is large, so their ratio $\kappa_2(T_+)/\kappa_2(T)$ is moderately large. This means that $\hat{R}(P^*\tilde{c}) = \hat{d}$ can be solved without adverse impact from ill-conditioning. It also means that $\tilde{b}$ is well approximated by $\tilde{A}\tilde{c}$, so $\| \check{d} \|_2$ is small. Note that $T^{-1}t = P^*\tilde{c}$. If $\kappa_2(T)$ is at or above some specified threshold, we can reduce the initial value of $m^{(\ell)}$ from its nominal value $\min(\ell, M)$ to compensate. For $2 \leq m^{(\ell)} < \min(\ell, M)$, we now identify $T = \hat{R}_{11}, t = \hat{d}_1$ and $\tau = \{\| \check{d} \|_2^2 + \| \hat{d}_2 \|_2^2\}^{\frac{1}{2}}$. We already have bounds and estimates for $\kappa_2(T)$, so we can choose the largest initial $m^{(\ell)}$ such that $\kappa_2(T)$ is below the threshold.

We can also easily calculate bounds and estimates for $\kappa_2(T_+)$. If $\| \hat{d}_2 \|_2$ is small, it may be possible by reducing $m^{(\ell)}$ to decrease $\kappa_2(T)$ below the threshold without significantly decreasing $\kappa_2(T_+)$, so the ratio $\kappa_2(T_+)/\kappa_2(T)$ increases. The ratio incorporates aspects of both redundance and relevance. This suggests choosing the initial $m^{(\ell)}$ to maximize the ratio, subject to $\kappa_2(T)$ being below the threshold. Whether this idea has practical merit worth the extra effort involved remains to be seen.

5 Choosing β and W

The default choice is $\beta = 1$, which is motivated by the tacit assumption that, for the given $x^{(0)}$, the Picard iteration $x^{(\ell+1)} = g(x^{(\ell)})$ converges, $x^{(\ell)} \to \hat{x} = g(\hat{x})$, albeit perhaps uncomfortably slowly. This means that we expect $y^{(\ell)} = g(x^{(\ell)})$ to be closer to $\hat{x}$ than $x^{(\ell)}$, and, by extension, $\hat{v}^{(\ell)}$ to be closer to $\hat{x}$ than $\hat{u}^{(\ell)}$ to the extent that $\hat{v}^{(\ell)}$ approximates $g(\hat{u}^{(\ell)})$, for $m^{(\ell)} > 0$.

The default choice is $W = I$, which is motivated by the tacit assumption that, absent additional problem-dependent information, we have no principled basis for

distinguishing one element of x or of $y = g(x)$ from another. Of course, simplicity and parsimony are also factors, as are others to be discussed below.

These assumptions may not be valid, in whole or in part, depending on the problem context involved. Prospective users of a utility code based on these default options should be made aware that it may well be productive to reconsider these issues based on their knowledge of a particular class of problems. Designers of a utility code should assume the responsibility to educate their prospective users, and to facilitate user response by making relevant options available. These matters will be discussed in more detail as we proceed. We shall initially consider the role of β, and later that of W. These are largely separable issues, but may interact.

5.1 Choice of β

Given $g : \mathbb{R}^N \to \mathbb{R}^N$ and $\beta \in \mathbb{R}$, define

$$\begin{aligned} g(x \mid \beta) &= (1-\beta)x + \beta g(x), \\ &= x + \beta(g(x) - x). \end{aligned}$$

We see that $g(x \mid 1) = g(x)$, $g(x \mid 0) = x$ and $g(x \mid -1) = 2x - g(x)$. For $\beta \neq 0$, $g(x \mid \beta)$ defines a fixed point problem whose fixed points coincide with those of g. For $\beta = 0$, any $x \in \mathbb{R}^N$ is a fixed point of $g(x \mid 0)$, with no direct connection to $g(x)$. Thus, if $g(\hat{x} \mid \beta) = \hat{x}$, for all $\beta \neq 0$, we also have $g(\hat{x} \mid 0) = \hat{x}$. We identify the nondefault Extrapolation Algorithm with $\beta \neq 1$ (and $\beta \neq 0$) applied to the $g(x)$ fixed point problem as the default Extrapolation Algorithm applied to the $g(x \mid \beta)$ fixed point problem. Since the behavior of the default Extrapolation Algorithm depends on the convergence properties of the Picard iteration for the fixed point problem to which it is applied, we are led to consideration of the convergence properties of the Picard iteration: first, for $g(\hat{x}) = \hat{x}$, then for $g(\hat{x} \mid \beta) = \hat{x}$, focusing on the influence of β.

The Picard iteration for g is locally convergent at $\hat{x} = g(\hat{x})$ if there is an $\epsilon > 0$ such that for any $x^{(0)}$ with $\| x^{(0)} - \hat{x} \| < \epsilon$, the Picard iterants $x^{(\ell+1)} = g(x^{(\ell)})$ satisfy $\| x^{(\ell)} - \hat{x} \| < \epsilon$, for $\ell > 0$, and converge to $\hat{x} : x^{(\ell)} \to \hat{x}$. In short, the iteration converges for all initial (and subsequent) iterants sufficiently close to the fixed point in the specified norm. For highly nonlinear g and large N, it will typically be the case that ϵ is small, but that there are many $x^{(0)}$ such that $\| x^{(0)} - \hat{x} \| \geq \epsilon$, even with $\| x^{(0)} - \hat{x} \|$ moderately large compared to ϵ, such that the Picard iteration converges to $\hat{x}$ with this initial iterant. However, identifying such $x^{(0)}$ may be difficult—often the most fraught part of the problem to be solved. Furthermore, note that while convergence per se does not depend on the norm, this characterization of local convergence and ϵ does. If $x^{(\ell)} \to \hat{x}$, then we know that $\| x^{(\ell)} - \hat{x} \| < \epsilon$ for sufficiently large ℓ.

Let $G(x) \in \mathbb{R}^{N \times N}$ be the Jacobian matrix of g at $x \in \mathbb{R}^N$. We shall take for granted the well-known facts that a sufficient condition for the Picard iteration to be locally convergent at $\hat{x} = g(\hat{x})$ is that $\| G(\hat{x}) \| < 1$ for some matrix norm compatible with the vector norm in question and that another sufficient condition is that $\rho(G(\hat{x})) < 1$. Moreover, the asymptotic rate of convergence of the iteration, for

sufficiently large ℓ, is controlled by the size of $\rho(G(\hat{x}))$: the smaller, the faster. For intuitive motivational purposes, observe that, with $G(\hat{x}) \neq 0$ and $\| x^{(\ell)} - \hat{x} \|$ small enough, we have

$$x^{(\ell+1)} = g(x^{(\ell)}) \approx g(\hat{x}) + G(\hat{x})(x^{(\ell)} - \hat{x}) = \hat{x} + G(\hat{x})(x^{(\ell)} - \hat{x}),$$

thence

$$(x^{(\ell+1)} - \hat{x}) \approx G(\hat{x})(x^{(\ell)} - \hat{x})$$

and

$$\| x^{(\ell+1)} - \hat{x} \| \approx \| G(\hat{x})(x^{(\ell)} - \hat{x}) \| \leq \| G(\hat{x}) \| \| x^{(\ell)} - \hat{x} \| .$$

For affine g, the approximation is exact, for any $x^{(\ell)}$. Furthermore, $\rho(G(\hat{x})) > 1$ implies that the Picard iteration is not locally convergent at $\hat{x}$. Recall that $\rho(G(\hat{x})) \leq \| G(\hat{x}) \|$. Therefore, the sufficient condition $\| G(\hat{x}) \| < 1$ implies the sufficient condition $\rho(G(\hat{x})) < 1$. Moreover, $\rho(G(\hat{x})) > 1$ implies that $\| G(\hat{x}) \| > 1$. The case $\rho(G(\hat{x})) = 1$ is equivocal with regard to local convergence at $\hat{x}$ and implies that $\| G(\hat{x}) \| \geq 1$. See Ortega and Rheinboldt [13], pages 299–303, and Ostrowski [14], pages 161–166. (Caution: the Ostrowski book may be somewhat difficult to read because the mathematical style, terminology, and notation were out of step with prevailing customs when published, as comparison with the nearly contemporaneous Ortega/Rheinboldt book illustrates.)

If $\rho(G(\hat{x})) < 1$, so the Picard iteration is locally convergent, there may be some $x^{(0)}$ with $\| x^{(0)} - \hat{x} \| \geq \epsilon$ such that the iterant sequence $\left\{x^{(\ell)}\right\}$ converges to $\hat{x}$. If not, applying the Extrapolation Algorithm may lead to $\| x^{(\ell)} - \hat{x} \| < \epsilon$, for some $\ell > 0$, so convergence to $\hat{x}$ ensues. If $\rho(G(\hat{x})) > 1$, so the Picard iteration is not locally convergent, for any given $x^{(0)} \neq \hat{x}$, convergence to $\hat{x}$ is a possibility but problematic, and as a practical matter very unlikely. Applying the Extrapolation Algorithm may still lead to $\| g(x^{(\ell)}) - x^{(\ell)} \| < \delta$, for some $\ell > 0$ and specified small δ, yielding an approximate fixed point $x^{(\ell)}$. Such computational (as opposed to mathematical) convergence to an approximate fixed point is more plausible for smaller $\rho(G(\hat{x}))$ and larger M, up to a point. In this computational convergence framework, one might even be able to dispense not only with local convergence but also existence of a fixed point, provided there are suitable, reasonably well-determined, approximate fixed points. Finally, it is conceivable that the equistationary Extrapolation Algorithm might generate a mathematically convergent iterative process even if the underlying Picard iteration does not.

We shall focus on the fact that if $g(\hat{x}) = \hat{x}$, then $g(\hat{x} \mid \beta) = \hat{x}$ for all $\beta = \mathbb{R}$. The corresponding Jacobian matrix is

$$G(\hat{x} \mid \beta) = (1 - \beta)I + \beta G(\hat{x}).$$

If the eigenvalues of $G(\hat{x})$ are λ_k, $1 \leq k \leq N$, then the eigenvalues of $G(\hat{x} \mid \beta)$ are $(1 - \beta) + \beta\lambda_k$. At least conceptually, we can examine the dependence of $\rho(G(\hat{x} \mid \beta)) = \max_k \mid (1-\beta)+\beta\lambda_k \mid$ on β. The Picard iteration is of interest only for $\beta \neq 0$; indeed, for $|\beta|$ sufficiently large, but not too large. Observe that $\rho(G(\hat{x} \mid 0)) = 1$. For small $\mid \beta \mid$, the fixed point is ill-determined.

We assume at the outset that $\rho(G(\hat{x})) < 1$, but $\rho(G(\hat{x})) \approx 1$, so the Picard iteration for $g(x) = g(x \mid 1)$ is locally convergent, but the asymptotic rate of convergence

is slow. Observe that $\rho(G(\hat{x} \mid 1)) = \rho(G(\hat{x})) < 1$. Define $\hat{\lambda} = \min_k |\lambda_k| = |\lambda_i|$ and $\check{\lambda} = \max_k |\lambda_k| = |\lambda_j| = \rho(G(\hat{x})) < 1$, for some $1 \leq i, j \leq N$. We have

$$||1 - \beta| - |\beta||\lambda_k|| \leq |(1 - \beta) + \beta\lambda_k| \leq |1 - \beta| + |\beta||\lambda_k|.$$

We shall consider the three cases $\beta < 0$, $\beta > 1$, and $0 < \beta < 1$, which exhaust the remaining possibilities.

For $\beta < 0$, we see that $|1 - \beta| = 1 + |\beta|$, so we obtain, for $1 \leq k \leq N$, from $0 \leq \hat{\lambda} \leq |\lambda_k| \leq \check{\lambda} < 1$,

$$\begin{aligned} ||1 - \beta| - |\beta||\lambda_k|| &= |(1 + |\beta|) - |\beta||\lambda_k||, \\ &= |1 + (1 - |\lambda_k|)|\beta||, \\ &= 1 + (1 - |\lambda_k|)|\beta|, \\ &\geq 1 + (1 - |\check{\lambda}|)|\beta|, \end{aligned}$$

which is a sharp inequality satisfied as an equality for $k = j$. It follows that

$$|(1 - \beta) + \beta\lambda_k| \geq 1 + (1 - \check{\lambda})|\beta|,$$

and we conclude that, for $\beta < 0$ and $\check{\lambda} < 1$, we have

$$\rho(G(\hat{x} \mid \beta)) \geq 1 + (1 - \check{\lambda})|\beta| > 1.$$

We note in passing that the argument remains valid also for $\check{\lambda} = 1$, except that the conclusion is that $\rho(G(\hat{x} \mid \beta)) \geq 1$; the argument is not valid for $\check{\lambda} > 1$. In particular, we see that $g(x \mid -1) = 2x - g(x)$, thence $G(\hat{x} \mid -1) = 2I - G(\hat{x})$. Since all eigenvalues of $G(\hat{x})$ lie inside the unit circle in the complex plane, all eigenvalues of $G(\hat{x} \mid -1)$ lie outside the unit circle.

For $\beta > 1$, we see that $|1 - \beta| = \beta - 1$ and $|\beta| = \beta$, so we obtain, for $1 \leq k \leq N$, from $0 \leq \hat{\lambda} \leq |\lambda_k| \leq \check{\lambda} < 1$,

$$\begin{aligned} ||1 - \beta| - |\beta||\lambda_k|| &= |(\beta - 1) - \beta|\lambda_k||, \\ &= |(1 - |\lambda_k|)\beta - 1|. \end{aligned}$$

For $\beta > 2/(1 - \hat{\lambda})$, we find that

$$||1 - \beta| - |\beta||\lambda_i|| = |(1 - \hat{\lambda})\beta - 1| > 1,$$

thence

$$|(1 - \beta) + \beta\lambda_i| > 1;$$

so we conclude that we have

$$\rho(G(\hat{x} \mid \beta)) > 1.$$

We note in passing that the argument and conclusion remain valid for $\check{\lambda} \geq 1$. For $1 \leq \beta \leq 2/(1 - \hat{\lambda})$, and $\check{\lambda} < 1$, we know that $\rho(G(\hat{x} \mid \beta))$ must increase from $\rho(G(\hat{x} \mid 1)) < 1$ to $\rho(G(\hat{x} \mid 2/(1 - \hat{\lambda}))) \geq 1$. In particular, we see that $\rho(G(\hat{x} \mid \beta)) < 1$ for $\beta > 1$, but β sufficiently small.

For $0 < \beta < 1$, we see that $|1 - \beta| = 1 - \beta$ and $|\beta| = \beta$, so we obtain, for $1 \leq k \leq N$, from $0 \leq \hat{\lambda} \leq |\lambda_k| \leq \check{\lambda} < 1$,

$$\begin{aligned} |1 - \beta| + |\beta||\lambda_k| &= (1 - \beta) + \beta|\lambda_k|, \\ &= 1 + \beta(|\lambda_k| - 1), \\ &\leq 1 + \beta(\check{\lambda} - 1), \end{aligned}$$

which is a sharp inequality satisfied as an equality for $k = j$. It follows that

$$|(1 - \beta) + \beta\lambda_k| \leq 1 + \beta(\check{\lambda} - 1),$$

and we conclude that, for $0 < \beta < 1$, and $\check{\lambda} < 1$, we have

$$\rho(G(\hat{x} \mid \beta)) \leq 1 + \beta(\check{\lambda} - 1) < 1.$$

We note in passing that the argument leading to the bound $\rho(G(\hat{x} \mid \beta)) \leq 1+\beta(\check{\lambda}-1)$ remains valid for $\check{\lambda} \geq 1$; but we have $1 + \beta(\check{\lambda} - 1) \geq 1$, so the bound is not very informative: see below.

5.1.1 Examples

As an illustrative example, consider the case where all eigenvalues of $G(\hat{x})$ are real, positive and labeled so that

$$0 < \lambda_N \leq \lambda_{N-1} \leq \cdots \leq \lambda_1,$$

with $\lambda_2 < \lambda_1 < 1$, so we have $\hat{\lambda} = \lambda_N$ and $\check{\lambda} = \lambda_1 = \rho(G(\hat{x})) < 1$. We see that $(1-\beta)+\beta\lambda_k = 1+\beta(\lambda_k - 1)$, for $1 \leq k \leq N$. We now observe that we can choose an optimal $\hat{\beta} > 1$ minimizing $\rho(G(\hat{x} \mid \hat{\beta}))$ by setting

$$0 < 1 + \hat{\beta}(\lambda_1 - 1) = -(1 + \hat{\beta}(\lambda_N - 1)) < 1$$

We then obtain

$$\hat{\beta} = [1 - (\lambda_1 + \lambda_N)/2]^{-1} > 1$$

and

$$\rho(G(\hat{x} \mid \hat{\beta})) = \{(\lambda_1 - \lambda_N)/2\}/[1 - (\lambda_1 + \lambda_N)/2] < \lambda_1 < 1.$$

(Since $\beta(\lambda_1 - 1) < 0$ and $1 + \beta(\lambda_1 - 1) = \lambda_1$, for $\beta = 1$, we see that $1 + \hat{\beta}(\lambda_1 - 1) < \lambda_1$, for $\hat{\beta} > 1$.) In essence, monotonic convergence of the Picard iteration for $g(x) = g(x \mid 1)$ permits choice of an optimal $\hat{\beta} > 1$, thence a greater asymptotic rate of convergence—but only modestly so for $0 \approx \lambda_N \ll \lambda_1 \approx 1$.

Consider also the case where all eigenvalues of $G(\hat{x})$ are real, negative and labeled so that

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N < 0,$$

with $-1 < \lambda_1 < \lambda_2$, so $\hat{\lambda} =\mid \lambda_N \mid$ and $\check{\lambda} =\mid \lambda_1 \mid= \rho(G(\hat{x})) < 1$. We see that $(1 - \beta) + \beta\lambda_k = 1 - \beta(1 + |\lambda_k|)$, for $1 \leq k \leq N$. We now observe that we can choose an optimal $\hat{\beta}$ such that $0 < \hat{\beta} < 1$ minimizing $\rho(G(\hat{x} \mid \hat{\beta}))$ by setting

$$0 < 1 - \hat{\beta}(1+ \mid \lambda_N \mid) = -(1 - \hat{\beta}(1+ \mid \lambda_1 \mid)) < 1.$$

We then obtain

$$\hat{\beta} = [1 + (|\lambda_1| + |\lambda_N|)/2]^{-1} < 1$$

and

$$\rho(G(\hat{x} \mid \hat{\beta})) = \{(|\lambda_1| - |\lambda_N|)/2\} / [1 + (|\lambda_1| + |\lambda_N|)/2] < |\lambda_1| < 1.$$

(Since $-\beta(1+ \mid \lambda_1 \mid) < 0$ and $-(1 - \beta(1+ \mid \lambda_1 \mid)) = \mid \lambda_1 \mid$ for $\beta = 1$, we see that $-(1 - \hat{\beta}(1 + |\lambda_1|)) < |\lambda_1|$ for $\hat{\beta} < 1$.) In essence, oscillatory convergence of the Picard iteration for $g(x) = g(x \mid 1)$ permits choice of an optimal $\hat{\beta}$ such that $0 < \hat{\beta} < 1$, potentially with a significantly greater asymptotic rate of convergence.

These two examples are simple and contrived. They illustrate that for $\rho(G(\hat{x})) = \check{\lambda} < 1$, so the Picard iteration for $g(x) = g(x \mid 1)$ is locally convergent, it may be possible to increase the asymptotic rate of convergence by using $g(x \mid \beta)$ for some β such that $0 < \beta < 1$ or for $1 < \beta < 2/(1 - \hat{\lambda})$. Since we have $\rho(G(\hat{x} \mid 0)) = 1$, $\rho(G(\hat{x} \mid 1)) < 1$, and $\rho(G(\hat{x} \mid 2/(1 - \hat{\lambda}))) > 1$, this is a reasonable expectation. Lacking information about $\hat{\lambda}$ and $\check{\lambda}$, but anticipating that $\hat{\lambda}$ is small and $\check{\lambda} \approx 1$, one might pragmatically focus on $0 < \beta < 2$. Of course, it could happen that $\hat{\beta} = 1$. There are other observations which they also illustrate, which we shall now explore.

Because of the absolute value and maximization, we know that

$$\rho(G(\hat{x} \mid \beta)) = \max_k |(1 - \beta) + \beta\lambda_k|$$

is a continuous function of β, but possibly only piecewise continuously differentiable. In these examples, the optimal $\hat{\beta}$ occurs at a point at which the derivative is discontinuous, because the eigenvalue whose modulus determines the spectral radius changes there. This could impact the approximate identification of $\hat{\beta}$ in more general problems.

In the oscillatory version of the example, the hypothesis that $\rho(G(\hat{x} \mid 1)) = |\lambda_1| < 1$, so the Picard iteration for $g(x) = g(x \mid 1)$ is locally convergent does not play a role in the determination of $\hat{\beta}$, with $0 < \hat{\beta} < 1$; though it does play a role in the size of $\rho(G(\hat{x} \mid \hat{\beta}))$. For moderate $|\lambda_1| > 1$, so the Picard iteration for $g(x) = g(x \mid 1)$ is not locally convergent, we may have $\rho(G(\hat{x} \mid \hat{\beta})) < 1$, so the Picard iteration for $g(x \mid \hat{\beta})$ is locally convergent. As $|\lambda_1|$ increases, $\hat{\beta}$ decreases and $\rho(G(\hat{x} \mid \hat{\beta}))$ increases. In the monotonic version of the example, the hypothesis that $\rho(G(\hat{x} \mid 1)) = |\lambda_1| < 1$ does play a role in the determination of $\hat{\beta} > 1$. Considering $\beta < 1$ makes more sense than $\beta > 1$ for $\rho(G(\hat{x} \mid 1)) > 1$. For more general problems, one may or may not be able to find a $\hat{\beta}$ such that $\rho(G(\hat{x} \mid \hat{\beta})) < 1$ for $\rho(G(\hat{x} \mid 1)) = \rho(G(\hat{x})) > 1$. Choosing $\beta < 0$ might be considered; examples will be discussed in the next section. Recalling that $\rho(G(\hat{x} \mid 0)) = 1$, I would be more inclined, on pragmatic grounds, to simply use a small, but not too small, positive β, the hope being that if $\rho(G(\hat{x} \mid \hat{\beta}))$ is not too large the Extrapolation Algorithm might succeed in finding an approximate fixed point. Experience suggests that this is a reasonable possibility, but by no means assured.

5.1.2 Algorithm

If the Picard iteration for $g(x) = g(x \mid 1)$ is slowly convergent, we anticipate that there may be a more suitable choice than $\beta = 1$, yielding a larger asymptotic rate of convergence. How might we find such a β ? In practice, β is often chosen for a class

of related problems by experimenting with representative examples for selected values of β. The experiments might measure the number of iterations required to reduce the residual norm to a specified small fraction of its initial value, either with the Picard iteration for $g(x \mid \beta)$ or the default Extrapolation Algorithm applied thereto, which is just the nondefault Extrapolation Algorithm with $\beta \neq 1$. I shall suggest below a conceptual procedure for adaptively choosing $\beta^{(\ell+1)}$ with the caveat that I have not had, and will not have, an opportunity to assess its practical utility. Before doing so, certain implementation issues must be clarified.

We tacitly assumed above that the Extrapolation Algorithm would be implemented as a code whose input is ℓ, M, N, $\mu^{(\ell)}$, $\tau^{(\ell)}$, $\beta^{(\ell)}$, and $(M+1) \times N$ arrays X and Y whose columns contain $x^{(\ell-k)}$ and $y^{(\ell-k)}$, $k = 0, 1, \ldots, \min(\ell, M)$, accessed using a pointer whose value is $(\ell - k)$ modulo $(M+1)$. The code would produce $x^{(\ell+1)}$ as output, plus as yet unspecified accessible byproducts. This means that a separate code generates $y^{(\ell+1)} = g(x^{(\ell+1)})$ and manages the iteration by testing for termination or initiating the next invocation of the Extrapolation Algorithm code. Termination tests will be discussed below. The two codes could be combined, but there are potential advantages to keeping them separate. The iteration management code could be combined with other codes involved in solving the overall problem, but there are potential advantages to keeping them separate—in which case, it may also produce accessible byproducts.

The simplest situation is that in which $\mu^{(\ell)}$, $\tau^{(\ell)}$, and $\beta^{(\ell)}$ are specified, as μ, τ, and β, respectively. We discussed previously how $\mu^{(\ell+1)}$ and/or $\tau^{(\ell+1)}$ might be chosen adaptively within the Extrapolation Algorithm code, so they must be accessible byproducts thereof. Choosing $\beta^{(\ell+1)}$ adaptively requires $y^{(\ell+1)}$, so this must be done in the iteration management code, after the termination tests and before reinitialization of the Extrapolation Algorithm code.

Recall that we have

$$\hat{u}^{(\ell)} = \sum_{k=0}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} x^{(\ell-k)} = x^{(\ell)} + \sum_{k=1}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} (x^{(\ell-k)} - x^{(\ell)})$$

$$\hat{v}^{(\ell)} = \sum_{k=0}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} y^{(\ell-k)} = y^{(\ell)} + \sum_{k=1}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} (y^{(\ell-k)} - y^{(\ell)}),$$

and

$$x^{(\ell+1)} = (1 - \beta^{(\ell)}) \hat{u}^{(\ell)} + \beta^{(\ell)} \hat{v}^{(\ell)}.$$

It is understood that $\hat{\theta}_0^{(\ell)} > 0$ and $\beta^{(\ell)} > 0$ and that if $m^{(\ell)} < \min(\ell, M)$ and if the iterant data pair $x^{(\ell-k)}$ and $y^{(\ell-k)}$ is being disregarded then $\hat{\theta}_k^{(\ell)} = 0$. Recall also that asymptotically, when all $x^{(\ell-k)}$ not being disregarded are close enough to $\hat{x}$, we expect that $\hat{v}^{(\ell)} \approx g(\hat{u}^{(\ell)})$, so $\| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \| \approx \| g(\hat{u}^{(\ell)}) - \hat{u}^{(\ell)} \|$.

Introduce relative and absolute termination tolerances ϵ_r and ϵ_a, with $\epsilon_r \geq 0$, $\epsilon_a \geq 0$ and $\epsilon_r + \epsilon_a > 0$. Also, introduce a maximum number of iterations L, as a fail-safe device. We specify three termination tests, to be executed sequentially. If these tests do not result in termination, we proceed to choose $\beta^{(\ell+1)}$. If we have

$$\| y^{(\ell+1)} - x^{(\ell+1)} \| \leq \epsilon_r \| x^{(\ell+1)} \| + \epsilon_a,$$

terminate reporting success with $x^{(\ell+1)}$ as the approximate fixed point. If we have

$$\| x^{(\ell+1)} - x^{(\ell)} \| \leq \epsilon_r \| x^{(\ell+1)} \| + \epsilon_a,$$

terminate reporting failure due to inadequate progress. If we have $\ell = L$, terminate reporting failure due to excessive iterations.

If the Picard iteration for $g(x \mid \beta^{(\ell)})$, with $\beta^{(\ell)} > 0$, converges, we anticipate that

$$x^{(\ell+1)} = (1 - \beta^{(\ell)})\hat{u}^{(\ell)} + \beta^{(\ell)}\hat{v}^{(\ell)}$$

will be closer to $\hat{x}$ than is $\hat{u}^{(\ell)}$, so we expect asymptotically that

$$\| y^{(\ell+1)} - x^{(\ell+1)} \| < \| g(\hat{u}^{(\ell)}) - \hat{u}^{(\ell)} \| \approx \| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \| .$$

For $\| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \| = 0$, take $\beta^{(\ell+1)} = \beta^{(\ell)}$. For $\| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \| > 0$, we shall take

$$\gamma^{(\ell)} := \| y^{(\ell+1)} - x^{(\ell+1)} \| / \| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \|$$

as a measure of the efficacy of the choice of $\beta^{(\ell)}$, by virtue of the convergence of the Picard iteration for $g(x \mid \beta^{(\ell)})$: smaller $\gamma^{(\ell)}$ corresponding to greater efficacy. For this purpose, $\| \hat{v}^{(\ell)} - \hat{u}^{(\ell)} \|$ should be an accessible byproduct of the Extrapolation Algorithm code. For $\ell > 0$, we abide by the proscription against using more than one g evaluation per iteration.

Before sketching a conceptual algorithm for choosing $\beta^{(\ell+1)}$, it is well to keep several things in mind. Motivating arguments in the foregoing depend on asymptotic properties valid for $x^{(\ell)}$ close enough to $\hat{x}$, which may not be valid. Even if they are asymptotically valid, they may not hold for $x^{(0)}$. Typically, there is a transient phase for small ℓ before the underlying Picard iteration and Extrapolation Algorithm settle into systematic patterns of behavior. Moreover, if the Extrapolation Algorithm proves to be reasonably effective in finding an approximate fixed point, the net gain in using a near-optimal $\beta^{(\ell)}$ rather than just an acceptable $\beta^{(\ell)}$ may have a small overall impact. Consequently, a safe-guarded primitive algorithm for choosing $\beta^{(\ell+1)}$ may suffice for our purposes.

Introduce $\dot{\beta}$ and $\ddot{\beta}$ such that $0 < \dot{\beta} < 1 < \ddot{\beta} < 2$. Specifically, choose a small (but not excessively small) $\dot{\beta}$ and set $\ddot{\beta} = 2 - \dot{\beta}$. Partition the interval $[\dot{\beta}, \ddot{\beta}]$ uniformly into an even number greater than 2 of subintervals, so $\beta^{(0)} = 1$ is the central partition point. The set of all partition points will be our candidates for $\beta^{(\ell+1)}$. Let $\triangle\beta$ be the length of the subintervals. Introduce a direction indicator d taking on values -1, 0, 1, and associated quantities γ_{-1}, γ_0, γ_1 and β_{-1}, β_0, β_1. For $\ell = 0$, initialize and invoke the Extrapolation Algorithm code and, absent termination, calculate $\gamma^{(0)}$. Set $\gamma_0 = \gamma^{(0)}$ and $\beta_0 = \beta^{(0)}$. If $\gamma_0 < 1$, set $d = 1$ and $\beta^{(1)} = \beta_0 + \triangle\beta$. If $\gamma_0 \geq 1$, set $d = -1$ and $\beta^{(1)} = \beta_0 - \triangle\beta$. Increment ℓ by 1.

For $\ell = 1$, reinitialize and invoke the Extrapolation Algorithm code and, absent termination, calculate $\gamma^{(1)}$. If $d = 1$, set $\gamma_1 = \gamma^{(1)}$ and $\beta_1 = \beta^{(1)}$. If $d = -1$ set $\gamma_{-1} = \gamma^{(1)}$ and $\beta_{-1} = \beta^{(1)}$. If $d = 1$ and $\gamma_1 > \gamma_0$, set $t = -1$ and $\beta^{(2)} = \beta_0 - \triangle\beta$. If $d = 1$ and $\gamma_1 \leq \gamma_0$, set $t = 1$ and $\beta^{(2)} = \beta_0 + \triangle\beta$, and transfer γ_0, β_0, γ_1, β_1, to γ_{-1}, β_{-1}, γ_0, β_0, respectively. If $d = -1$ and $\gamma_{-1} > \gamma_0$, set $t = 0$ and $\beta^{(2)} = \beta_0$. If $d = -1$ and $\gamma_{-1} \leq \gamma_0$, set $t = -1$ and $\beta^{(2)} = \beta_0 - \triangle\beta$, and transfer γ_{-1}, β_{-1}, γ_0, β_0, to γ_0, β_0, γ_1, β_1, respectively. Set $d = t$ and increment ℓ by 1.

If $\ell \geq 2$, reinitialize and invoke the Extrapolation Algorithm code and, absent termination, calculate $\gamma^{(\ell)}$. If $d = 0$, set $t = 0$ and $\beta^{(\ell+1)} = \beta_0$. If $d = 1$, set $\gamma_1 = \gamma^{(\ell)}$ and $\beta_1 = \beta^{(\ell)}$. If $d = -1$ set $\gamma_{-1} = \gamma^{(\ell)}$ and $\beta_{-1} = \beta^{(\ell)}$. If $d = 1$ and $\gamma_1 > \gamma_0$, set $t = 0$ and $\beta^{(\ell+1)} = \beta_0$. If $d = 1$, $\gamma_1 \leq \gamma_0$ and $\beta_1 + \triangle\beta > \ddot{\beta}$, set $t = 0$ and $\beta^{(\ell+1)} = \ddot{\beta}$. If $d = 1$, $\gamma_1 \leq \gamma_0$ and $\beta_1 + \triangle\beta \leq \ddot{\beta}$, set $t = 1$ and $\beta^{(\ell+1)} = \beta_1 + \triangle\beta$. If $d = -1$ and $\gamma_{-1} > \gamma_0$, set $t = 0$ and $\beta^{(\ell+1)} = \beta_0$. If $d = -1$, $\gamma_{-1} \leq \gamma_0$ and $\beta_{-1} - \triangle\beta < \dot{\beta}$, set $t = 0$ and $\beta^{(\ell+1)} = \dot{\beta}$. If $d = -1$, $\gamma_{-1} \leq \gamma_0$ and $\beta_{-1} - \triangle\beta \geq \dot{\beta}$, set $t = -1$ and $\beta^{(\ell+1)} = \beta_{-1} - \triangle\beta$. Set $d = t$ and increment ℓ by 1.

A code using a decision tree would be simpler than the foregoing might appear. The net effect is to attempt to identify a candidate for β that will enhance the asymptotic rate of convergence of the Picard iteration, and, failing this, to fix β at a limiting value $\dot{\beta}$ (or, conceivably, $\ddot{\beta}$), hoping that the Extrapolation Algorithm will succeed in finding an approximate fixed point.

There is a point which needs clarification in anticipation of matters to be discussed in the next section, involving connections between fixed point problems and root-finding problems—or, more specifically, zero-finding problems. Our starting point is the fixed point problem for g, which we assume to have a locally convergent Picard iteration at $g(\hat{x}) = \hat{x}$. This fixed point problem may be explicit as the numerical problem whose solution is sought, or implicit in an iterative process for solving another numerical problem; for example, a root-finding problem $f(\hat{x}) = 0$.

The fixed point problem $g(\hat{x}) = \hat{x}$ is naturally associated with the root-finding problem $g(\hat{x}) - \hat{x} = 0$, though many people prefer $\hat{x} = g(\hat{x})$ and $\hat{x} - g(\hat{x}) = 0$. With the root-finding problem $f(\hat{x}) = 0$, we can naturally associate the fixed point problem for $x + f(x)$, though many people prefer $x - f(x)$. With the root-finding problem $g(\hat{x}) - \hat{x} = 0$, we see that $x + f(x) = g(x) = g(x \mid 1)$, and $x - f(x) = 2x - g(x) = g(x \mid -1)$, which we know have radically different convergence properties. Likewise, with $\hat{x} - g(\hat{x}) = 0$, we see that $x + f(x) = g(x \mid -1)$, and $x - f(x) = g(x)$. The root-finding problem $f(\hat{x}) = 0$ is essentially unaltered if replaced by $\alpha f(\hat{x}) = 0$, for $\alpha \neq 0$. Similarly, for $g(\hat{x}) - \hat{x} = 0$, we see that $x + \alpha f(x) = g(x \mid \alpha)$ and $x - \alpha f(x) = g(x \mid -\alpha)$, and for $\hat{x} - g(\hat{x}) = 0$, we see that $x + \alpha f(x) = g(x \mid -\alpha)$ and $x + \alpha f(x) = g(x \mid \alpha)$. For general root-finding problems, there is no correspondence between α and β. It may or may not be possible to choose α to arrange for or accelerate the convergence of the corresponding Picard iteration, and the sign of α may be significant.

Finally, the root-finding problem $f(\hat{x}) = 0$ is also essentially unaltered if replaced by $Bf(\hat{x}) = 0$, for nonsingular B. Let $F(\hat{x})$ be the Jacobian of f at $\hat{x}$ and assume that $F(\hat{x})$ is nonsingular, so $\hat{x}$ is a locally unique solution. An ideal (but infeasible) choice for B with $x \pm Bf(x)$ would be $B = \mp F(\hat{x})^{-1}$, so the Jacobian $I \pm BF(x)$ is 0 at $\hat{x}$. Newton's method approximates $F(\hat{x})$ by $F(x^{(\ell)})$, which is impractical in our context. However, other iterative procedures for root-finding problems can be interpreted from this perspective: see further below.

5.2 Choice of W

We turn now to the choice of W in the Extrapolation Algorithm. Note that a different choice might be appropriate in the termination tests. Recall that $W = Diag(w)$, so

$w = diag(W)$, with $w > 0$. Thus, we can equivalently discuss the choice of w. I shall normalize W by requiring that $\| w \|_2 / \sqrt{N} = 1$, which is the case for $W = I$ where $w = e$. Several preliminary remarks are in order before we proceed. First, for a fixed point problem $g(\hat{x}) = \hat{x}$, there is a natural correspondence between the elements of x and of $g(x)$, thence comparable scaling considerations for x and $g(x)$. For a root-finding problem $f(\hat{x}) = 0$, there is not necessarily any such association between x and f; one can think of the earlier generation of a fixed point problem from a root-finding problem as an attempt to establish such a connection.

Second, we wrote $\beta^{(\ell)}$ in anticipation of choosing β adaptively, but we did not write $W^{(\ell)}$. Adaptive choice of $W^{(\ell)}$ would complicate monitoring the iteration, and feedback might lead to potential instability. We contemplate that W will be chosen at the outset, but could allow the iteration to be restarted episodically or periodically. In particular, recall the observation above that there may be a brief unrepresentative transient phase at the beginning if the initial iterant is inadequate. For both the choice of β and W, it may be helpful to do a small number (2–3) of iterations at the outset with the default values before restarting the iteration with an updated initial iterant, and nondefault values of β and W. If β has been chosen adaptively, it might be fixed in any subsequent restart. Note that changing β and W does not entail discarding prior iterant data, though early data regarded as unrepresentative might well be discarded.

Third, the choice of a nondefault W necessarily involves problem-dependent knowledge allowing us to make cogent distinctions among subsets of the elements of x and $g(x)$. We posit that we can partition the elements of x and $g(x)$ into a relatively small number of subvectors of significant size with relevantly different characteristics. We shall assign all elements of corresponding subvectors of w the same value. The efficacy of the Extrapolation Algorithm hinges on perceiving and predicting pertinent patterns in the iterant data, and all subvectors must contribute appropriately to the inner products and norms centrally involved. There may be complementary and competing considerations requiring careful compromises. Insights from the scientific or engineering context from which the mathematical problem being solved numerically derives may be crucial.

The most straightforward basis for partitioning x and $g(x)$ into subvectors arises when the mathematical problem involves several dependent variables defined over some domain, so subvectors can be associated with the discretized version of each dependent variable. Such initial subvectors might be subdivided further based on geometric or other considerations. For instance, the class of problems that originally motivated the development of the Extrapolation Algorithm constituted a set of three to five coupled singular nonlinear Fredholm integral equations of the second kind, modeling a rarefied gas: for example, argon. The dependent variables involved were a number density, a temperature, and one to three velocity components, for which scientifically natural units would be moles per cubic meter, kelvins, and meters per second. Using such natural units may lead to dependent variables of disparate sizes. Scaling to balance their contributions to inner products and norms may be in order on numerical grounds, but this is a complicated issue which is context-dependent.

I shall briefly discuss four sets of ideas related to W in what follows, which I shall label as adjustment, influence, decimation, and implementation. The intent is not to be definitive, but simply to suggest that the choice of W is worth thinking

about seriously in the framework of a class of related problems, especially if these are challenging problems.

5.2.1 Adjustment

Adjustment is related to but distinguishable from simple-minded rescaling of multiple dependent variables of disparate sizes to make them comparable in size. Consider the nonsingular affine transformation of variables $z = W(x - s)$, thence $x = W^{-1}z + s$. Correspondingly, define

$$h(z) = W(g(W^{-1}z + s) - s),$$

thence

$$g(x) = W^{-1}h(z) + s.$$

Then, the fixed point problems $g(\hat{x}) = \hat{x}$ and $h(\hat{z}) = \hat{z}$ are related by $\hat{z} = W(\hat{x} - s)$, thence $\hat{x} = W^{-1}\hat{z}+s$. We should select the shift vector s so that $e_i^* s$ is a representative value of $e_i^* x$ or $e_i^* g(x)$, $1 \leq i \leq N$, in some neighborhood of $\hat{x}$. The choice $s = \hat{x}$ would be ideal, but infeasible. Consequently, $W^{-1}z$ constitutes the deviation of x from s. Observe that we have

$$\begin{aligned}
&\left\| \sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} \left[h(z^{(\ell-k)}) - z^{(\ell-k)} \right] \right\|_2^2 \\
&= \left\| \sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} \left[W(g(x^{(\ell-k)}) - s) - W(x^{(\ell-k)} - s) \right] \right\|_2^2, \\
&= \left\| \sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} \left[Wg(x^{(\ell-k)}) - Wx^{(\ell-k)} \right] \right\|_2^2, \\
&= \left\| W \sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} \left[g(x^{(\ell-k)}) - x^{(\ell-k)} \right] \right\|_2^2.
\end{aligned}$$

Therefore, the $\hat{\theta}_k^{(\ell)}$, $0 \leq k \leq \min(\ell, M)$, depend on W but do not depend on s, except insofar as the choice of s affects that of W. From this perspective, the choice of W should be made to roughly equilibrate $z = W(W^{-1}z)$ rather than $W^{-1}z$. Units affect the deviations as well as the representative values, but ordinarily more moderately if the latter are disparate in size. We are rescaling the residual $g(x) - x$ rather than x and $g(x)$. For example, if x and $g(x)$ have initially been partitioned into subvectors corresponding to different dependent variables, possibly further subdivided, one might choose the elements of the corresponding subvectors of w as a multiple of the reciprocal of the standard deviation (assumed nonzero) of the set of elements of the corresponding subvector of $g(x^{(0)})$ and $x^{(0)}$. The multiplier should be chosen so that $\| w \|_2 / \sqrt{N} = 1$. Among other things, this would adjust for differences in units. In the unusual event of a zero, or excessively small, standard deviation, one might

temporarily assign a zero value to the elements of that subvector, and choose the multiplier so that $\| w \|_2 / \sqrt{n} = 1$, where n is the number of nonzero elements of w. If we then reassign the $N - n$ temporarily zero elements of w the value one, we will obtain $\| w \|_2 / \sqrt{N} = 1$. This initial w might be modified based on considerations discussed hereafter.

5.2.2 Influence

In addition to issues related to disparate size, there are potential issues of disparate influence which are worth looking for in example calculations and anticipating or rationalizing in the problem context. For illustration, we shall dichotomize, but there might be intermediate categories. Suppose that there is a category of volatile variables which depend sensitively on a category of nonvolatile variables which ultimately determine the values of both. The volatile variables may be ill-determined even when the values of the nonvolatile variables have stabilized. Turbulent behavior of volatile variables may obscure systematic behavior of nonvolatile variables. One may profit by downweighting the volatile variables and letting them dominate only after the nonvolatile variables have stabilized. Suppose that there is a category of stolid variables which are largely determined (for example, by boundary or asymptotic conditions) for the particular problem at hand. If there are a significant number of stolid variables, one may profit from upweighting the more active nonstolid variables. Adjustment also responds to volatility and stolidity. In complementary fashion, sensitive variables for which small changes can cause much larger changes in other variables might be upweighted to inhibit excessive variation, and insensitive variables for which moderately large changes are required to significantly affect other variables might be downweighted. This is a surrogate, based on qualitative knowledge of the problem context (if available and unequivocal), for quantitative information about off-diagonal elements of the Jacobian. In the same vein, there may be localized regions within the domain which play a key role and should be focused upon during the iteration. These are examples of disparate influence of subvectors which, if anticipated, may be worth incorporating into the choice of W. Downweighting or upweighting might be applied to an initial w chosen along the lines laid out above. Downweighting or upweighting would be followed by renormalizing so that $\| w \|_2 / \sqrt{N} = 1$. Consequently, downweighting (upweighting) one subvector would be accompanied by upweighting (downweighting) the other subvector.

5.2.3 Decimation

Before discussing decimation, I shall briefly sketch two more familiar ideas which should not be confused with it. Let N be the number of degrees of freedom to be determined in a discretization of a continuous problem like a differential or integral equation. For ease in exposition, we focus on a single dependent variable, but extension to several is straightforward. For challenging nonlinear problems of this sort requiring large N, I take it for granted that potential use of a continuation procedure in N will be on the agenda. This involves solving a family of problems with increasing N, taking an approximate solution for one N as the initial iterant for the next

larger N, beginning with an N large enough to capture the essence of the problem but not large enough to yield the requisite accuracy. In our context, we imagine solving a fixed point problem for a given N and initial iterant, whose Picard iteration is increasingly costly and slowly convergent as N becomes larger. Consequently, the overall cost of solving a family of such problems may be less than that of solving the problem with maximal N alone.

Consider fixed point problems whose Picard iteration preserves and enhances smoothness of the iterants. Discretizations of Fredholm integral equations of the second kind naturally lead to such problems because integration is a global averaging, thence global smoothing, process. Discretizations of appropriate differential equations using elementary iterative methods based on a local averaging, thence preferential local smoothing, process may also yield problems of this sort. This is familiar in the context of multigrid, or multilevel, methods, also involving a family of discretized problems akin to those in the aforementioned continuation procedure. By systematically cycling among different family members, one seeks to damp out smaller scale errors before moving to the next member.

Neither of the foregoing ideas is of current interest here. However, one can imagine situations in which continuation or multigrid iterations might be used to define the fixed point problem to which the Extrapolation Algorithm is applied, or in which the Extrapolation Algorithm is applied within stages of the continuation or multigrid iterations.

Recall that N-vectors enter the Extrapolation Algorithm first, and foremost, in the evaluation of inner products and norms involved in the calculation of the affine combination coefficients $\hat{\theta}_k^{(\ell)}$, $0 \leq k \leq \min(\ell, M)$, and subsequently in the calculation of the affine combinations. The use of large N to achieve desired numerical accuracy may lead to a form of redundancy, which we can attempt to alleviate. We shall identify two relevant situations. The first situation arises when elements of x represent local approximations to values of the dependent variable in the vicinity of points within the domain. Elements of x associated with nearby points must be nearly equal, the moreso as N increases with refinement of the discretization. The second situation arises when elements of x represent coefficients in a linear combination of basis functions approximating the dependent variable globally throughout the domain, and when they can be ordered so as to decrease rapidly in magnitude as N increases with refinement of the discretization. Finite difference, finite volume, and finite element methods using piecewise polynomial nodal basis functions of small support yield the first situation. Finite orthogonal expansions in trigonometric functions, orthogonal polynomials or other special functions, and finite element methods with hierarchical basis functions yield the second situation. The basis functions which are more highly oscillatory or have smaller support resolve finer details and their coefficients become small for smooth dependent variables.

In the integral equation problems motivating the development of the Extrapolation Algorithm, a dual representation of the dependent variables was employed, using both values at specially selected grid points and coefficients of finite expansions in Chebyshev polynomials of the first kind, connected via the well-known discrete orthogonality conditions. The problems were small enough so that decimation applied to grid point values was not a plausible tactic, but could possibly have

been applied to expansion coefficients because of the smoothness of the solutions, though incentive to do so was absent. Small expansion coefficients attributable to smoothness may also be regarded as stolid.

Having identified interesting situations arising from two classes of discretizations, we now note that they could be combined. If there are several dependent variables, different modes of discretization appropriate to each could be utilized. If there are several independent variables, different modes of discretization appropriate to each could be utilized. This approach is not uncommon in practice. We shall focus below simply on the two situations identified above.

I prefer to think about decimation in the framework of choosing W, but this is not essential. We shall formally relax the constraint $w > 0$ to $w \geq 0$ and assume that w is normalized so that $\| w \|_2 / \sqrt{n} = 1$, where n is the number of nonzero elements of w, with $1 \ll n \ll N$. It is understood from the outset that we shall not simply apply the foregoing algorithms with the corresponding W.

The elements of x and $g(x)$ corresponding to the nonzero elements of w will be called the representative subset, and those corresponding to the zero elements the complementary subset. In practice, the representative subset will be chosen, followed by w, as discussed further below. If the complementary subset of x is held fixed, g defines a fixed point problem and Picard iteration for the representative subset, to which the Extrapolation Algorithm could be applied. One can envision an analog of the block Jacobi or block Gauss-Seidel iteration in which x is partitioned into subsets which are identified successively with the representative subset. A small number of Picard or Extrapolation Algorithm iterations could be applied for each representative subset, and the Picard iteration or Extrapolation Algorithm could then be applied to the overall iterative process. As a practical matter, this requires the capability to evaluate subvectors of g independently, which may not be feasible. Again, this approach is not of current interest here and is mentioned only to distinguish it from the decimation idea to follow.

Recall that in the Extrapolation Algorithm, we essentially use a single Picard iteration to generate $y^{(\ell)} = g(x^{(\ell)})$ from $x^{(\ell)}$, but, for $\ell > 0$, $x^{(\ell)}$ is not itself ordinarily the direct product of a Picard iteration. It is iterant data pairs $x^{(\ell-k)}$ and $y^{(\ell-k)}$, $0 \leq k \leq \min(\ell, M)$, that enter the determination of $x^{(\ell+1)}$ by the Extrapolation Algorithm. This involves calculating the affine combination coefficients $\hat{\theta}_k^{(\ell)}$, $0 \leq k \leq \min(\ell, M)$, thence the affine combinations $\hat{u}^{(\ell)} = \sum_{k=0}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} x^{(\ell-k)}$ and $\hat{v}^{(\ell)} = \sum_{k=0}^{\min(\ell,M)} \hat{\theta}_k^{(\ell)} y^{(\ell-k)}$, and finally $x^{(\ell+1)} = (1 - \beta^{(\ell)})\hat{u}^{(\ell)} + \beta^{(\ell)}\hat{v}^{(\ell)}$. In the decimation approach, the idea is to use the n-vector representative subsets of $x^{(\ell-k)}$ and $y^{(\ell-k)}$, $0 \leq k \leq \min(\ell, M)$, to calculate the affine combination coefficients $\hat{\theta}_k^{(\ell)}$, $0 \leq k \leq \min(\ell, M)$, and then use these to calculate the N-vectors $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$, thence $x^{(\ell+1)}$. Whether it makes sense to approximate the affine combination coefficients calculated using N-vectors by those calculated using n-vectors depends on the nature of the original fixed point problem and on the selection of the representative subsets involved.

The key assumption is that the convergent Picard iteration for g preserves and enhances smoothness. Forming affine combinations does likewise. In the first situation outlined above, elements of x in the representative subset may be chosen as

proxies for neighboring elements with nearly equal values; one must take n large enough to adequately sample all relevant neighborhoods. The nonzero elements of w should ideally be proportional to the number of members of the complementary subset for which that element of the representative subset serves as a proxy. In the second situation outlined above, representative elements must include all coefficients of significant magnitude. The process might be facilitated by recognizing variables contained in a less refined discretization within a more refined discretization. With ample computational resources at their disposal, scientists and engineers commonly seek high enough numerical accuracy that n might usefully be taken much smaller than N, especially when multiple dependent and independent variables are involved.

5.2.4 Implementation

We turn now to implementation issues. I reiterate that code providers should educate prospective users about potential advantages of choosing $W \neq I$ and facilitate this to the extent feasible. Above (and below), I have chosen to incorporate W into the Extrapolation Algorithm calculations when forming the AB array, containing the ingredients A and b of the least squares problem to be solved, from the arrays X and Y containing the input iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)} = g(x^{(\ell-k)})$, $0 \leq k \leq \min(\ell, M)$. There are two other approaches that could be considered.

The most elegant, but least attractive, approach would involve reworking the constructions detailed above, using the standard Euclidean inner product and norm, in terms of the weighted inner product and norm introduced to define the minimization problem determining the optimal affine combination coefficients. The N^{-1} factor in the inner product and the $N^{-\frac{1}{2}}$ factor in the norm could be accommodated using the implicit scaling strategy sketched previously.

More interesting in practice is the observation that we could use an algorithm based on the default choice $W = I$, but replace the input iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)}$ by $Wx^{(\ell-k)}$ and $Wy^{(\ell-k)}$, $0 \leq k \leq \min(\ell, M)$. Furthermore, we could use an algorithm based on the default choice $\beta^{(\ell)} = \beta = 1$, but replace the input iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)}$ by $Wx^{(\ell-k)}$ and $(1 - \beta^{(\ell)})Wx^{(\ell-k)} + \beta^{(\ell)}Wy^{(\ell-k)}$, $0 \leq k \leq \min(\ell, M)$. One can envision an interface subprogram that accepts the original input iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)}$, $0 \leq k \leq \min(\ell, M)$, and produces the modified input iterant data for use by the default algorithm, and vice versa.

In my experience, scientists and engineers are skilled and comfortable with the use of scaling and other transformations in the formulation of mathematical models, to identify relevant dimensionless parameters and suitable approximations. They are often reluctant to accept the desirability of further scaling and other transformations for numerical purposes, and recalcitrant about producing input or being presented with intermediate or final output results in other than the variables and units natural to the problem context. An interface subprogram could be provided by a user interested in exploring the potential advantages of nondefault options. Alternatively, an interface subprogram could be provided by a member of the project team who already appreciates such advantages and can deploy them to meet the unfelt needs of the user. Note that the termination criterion has then been altered correspondingly.

I tacitly assumed above that the Extrapolation Algorithm code would recognize three cases with regard to β. The first case is the default option $\beta = 1$, which means that $x^{(\ell+1)} = \hat{v}^{(\ell)}$, so $\hat{u}^{(\ell)}$ need not be evaluated. The second case involves a specified $\beta \neq 1$, which means that $x^{(\ell+1)} = (1-\beta)\hat{u}^{(\ell)} + \beta\hat{v}^{(\ell)}$, so both $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$ are required. The third case is $x^{(\ell+1)} = (1-\beta^{(\ell)})\hat{u}^{(\ell)} + \beta^{(\ell)}\hat{v}^{(\ell)}$ and $\beta^{(\ell+1)}$ is to be chosen adaptively. It is much more important that the code recognize three cases with regard to W. The first case is the default option $w = e$, which means that $[A \;\; b]$ can be evaluated column-by-column, as presented above, omitting all vacuous multiplications by $W = I$. The second case involves a specified $w > 0$. Again $[A \;\; b]$ can be evaluated column-by-column, as presented above. We may be able to exploit Fortran array products, which are element-by-element Hadamard products, using w rather than formal use of W. The third case involves a specified $w \geq 0$, with $n \ll N$. We now evaluate $[A \;\; b]$ row-by-row, ignoring the $N-n$ rows associated with zero elements of w and multiplying corresponding rows by the nonzero elements, obtaining n as a byproduct.

The user must provide nondefault w. A stand-alone code or subroutine along the following lines might be of assistance. The basic input would be an N-vector which might become w upon output. The elements of the input vector would be nonzero integers whose magnitudes designate which subvector of x and $g(x)$ they are associated with, and whose signs designate whether they are members of the representative subset (positive) or the complementary subset (negative). Also, part of the input would be a list of all subvectors which are to be downweighted or upweighted, and a positive weighting factor less than, equal to, or greater than one. Initially, the output vector would be defined by the adjustment procedure described above. For this purpose, the code must also be supplied with $x^{(0)}$ and $y^{(0)} = g(x^{(0)})$. Then, if the weighting factor is not one, the listed subvectors would be multiplied by the weighting factor. Finally, all elements of the output w vector in the complementary subset would be set equal to zero, n would be determined, and the output vector normalized so that we obtain $\| w \|_2 / \sqrt{n} = 1$. For ease in exposition, we have discussed the simplest version using all of the foregoing, which could be elaborated upon.

6 Remarks on relevant literature

It is not my purpose here to comprehensively review the extensive literature pertaining to Anderson Acceleration, Anderson Mixing, and equivalent or related methods. Rather, I shall focus on selected aspects of noteworthy items and on two themes, one computational and the other conceptual.

Later in this section, I shall pay particular attention to the already influential Walker and Ni [16] paper, which introduced the Anderson Acceleration terminology, and the next section will be devoted to detailed discussion of related implementation issues. Comparing and contrasting computational considerations in the literature to those laid out above is the recurrent first theme. I believe that portions of the literature might well lead readers astray.

In the classic Ortega and Rheinboldt [13] magnum opus on root-finding problems, there was a note (pages 204–205) on the Anderson [1] paper. No mention was made

of the fact that the Extrapolation Algorithm was motivated by and intended for fixed point problems with slowly converging Picard iterations. Rather, it was recast as a method for root-finding problems, in a form which made it appear to be a silly idea. Repeated conflicts between thinking in terms of fixed point or root-finding problems, with consequent confusion, is the recurrent second theme. I have already indicated how I believe this conflict ought to be resolved, and will explain why and in what sense.

6.1 Broyden

There is a large literature stemming from the Broyden [4] paper, with alternative and competing terminology and characterizations. A sketch for orientation hereafter will suffice for our purposes, and I shall adopt my own language in aid of clarity and conciseness. The goal is to connect the Extrapolation Algorithm with, but distinguish it from , this body of material, which is framed in the context of the root-finding problem $f(\hat{x}) = 0$.

As above, denote the Jacobian of $f(x)$ by $F(x)$, assumed nonsingular. Let x_0 and x_1 be two distinct nearby points and assume that $f(x_0)$ and $f(x_1)$ are also distinct nearby points. We have the direct approximation

$$f(x_1) - f(x_0) \approx F(x_0)(x_1 - x_0)$$

and the inverse approximation

$$x_1 - x_0 \approx F(x_0)^{-1}(f(x_1) - f(x_0)).$$

Taking $x_1 = \hat{x}$, so $f(x_1) = 0$, we obtain

$$\hat{x} \approx x_0 - F(x_0)^{-1} f(x_0)$$

or

$$F(x_0)(\hat{x} - x_0) \approx -f(x_0).$$

This is the approximation underlying the Newton method: for $\ell = 0, 1, \ldots$,

$$x^{(\ell+1)} = x^{(\ell)} - F(x^{(\ell)})^{-1} f(x^{(\ell)})$$

or

$$F(x^{(\ell)})(x^{(\ell+1)} - x^{(\ell)}) = -f(x^{(\ell)}).$$

In the original Broyden direct secant method, we initially possess $x^{(\ell-1)}$, $f(x^{(\ell-1)})$ and an approximation $J^{(\ell-1)}$ to $F(x^{(\ell-1)})$. We solve $J^{(\ell-1)}(x^{(\ell)} - x^{(\ell-1)}) = -f(x^{(\ell-1)})$ to obtain $x^{(\ell)} = x^{(\ell-1)} - (J^{(\ell-1)})^{-1} f(x^{(\ell-1)})$ and evaluate $f(x^{(\ell)})$. We then obtain a companion $J^{(\ell)}$ by satisfying the direct secant condition

$$J^{(\ell)}(x^{(\ell-1)} - x^{(\ell)}) = (f(x^{(\ell-1)}) - f(x^{(\ell)})),$$

or equivalently,

$$J^{(\ell)}(x^{(\ell)} - x^{(\ell-1)}) = (f(x^{(\ell)}) - f(x^{(\ell-1)})),$$

and also the restriction that $J^{(\ell)}d = J^{(\ell-1)}d$, for all nonzero $d \perp (x^{(\ell)} - x^{(\ell-1)})$. It can be shown that this is equivalent to minimizing $\| J^{(\ell)} - J^{(\ell-1)} \|_F$ subject to the direct secant condition constraint, but this will play no direct role in what follows.

In the original Broyden inverse secant method, we initially possess $x^{(\ell-1)}$, $f(x^{(\ell-1)})$, and an approximation $K^{(\ell-1)}$ to $F(x^{(\ell-1)})^{-1}$. We obtain $x^{(\ell)} = x^{(\ell-1)} - K^{(\ell-1)} f(x^{(\ell-1)})$ and evaluate $f(x^{(\ell)})$. We then obtain a companion $K^{(\ell)}$ by satisfying the inverse secant condition

$$K^{(\ell)}(f(x^{(\ell-1)}) - f(x^{(\ell)})) = (x^{(\ell-1)} - x^{(\ell)}),$$

or equivalently,

$$K^{(\ell)}(f(x^{(\ell)}) - f(x^{(\ell-1)})) = (x^{(\ell)} - x^{(\ell-1)}),$$

and also the restriction that $K^{(\ell)}d = K^{(\ell-1)}d$, for all nonzero $d \perp (f(x^{(\ell)}) - f(x^{(\ell-1)}))$. It can be shown that this is equivalent to minimizing $\| K^{(\ell)} - K^{(\ell-1)} \|_F$ subject to the inverse secant condition constraint.

Before discussing these methods further, it will prove convenient to slightly reorganize the calculations involved, with no change in substance.

In the reorganized original Broyden direct secant method, we initially possess $x^{(\ell-1)}$, $f(x^{(\ell-1)})$, $x^{(\ell)}$, $f(x^{(\ell)})$, and an approximation $J^{(\ell-1)}$ to $F(x^{(\ell-1)})$, with $x^{(\ell)}$ obtained as above. We obtain a companion $J^{(\ell)}$ by satisfying the direct secant condition

$$J^{(\ell)}(x^{(\ell-1)} - x^{(\ell)}) = (f(x^{(\ell-1)}) - f(x^{(\ell)})),$$

or equivalently,

$$J^{(\ell)}(x^{(\ell)} - x^{(\ell-1)}) = (f(x^{(\ell)}) - f(x^{(\ell-1)})),$$

and also the restriction that $J^{(\ell)}d = J^{(\ell-1)}d$, for all nonzero $d \perp (x^{(\ell)} - x^{(\ell-1)})$. We then solve $J^{(\ell)}(x^{(\ell+1)} - x^{(\ell)}) = -f(x^{(\ell)})$ to obtain $x^{(\ell+1)} = x^{(\ell)} - (J^{(\ell)})^{-1} f(x^{(\ell)})$ and evaluate $f(x^{(\ell+1)})$.

In the reorganized original Broyden inverse secant method, we initially possess $x^{(\ell-1)}$, $f(x^{(\ell-1)})$, $x^{(\ell)}$, $f(x^{(\ell)})$ and an approximation $K^{(\ell-1)}$ to $F(x^{(\ell-1)})^{-1}$. We obtain a companion $K^{(\ell)}$ by satisfying the inverse secant condition

$$K^{(\ell)}(f(x^{(\ell-1)}) - f(x^{(\ell)})) = (x^{(\ell-1)} - x^{(\ell)}),$$

or equivalently,

$$K^{(\ell)}(f(x^{(\ell)}) - f(x^{(\ell-1)})) = (x^{(\ell)} - x^{(\ell-1)}),$$

and also the restriction that $K^{(\ell)}d = K^{(\ell-1)}d$, for all nonzero $d \perp (f(x^{(\ell)}) - f(x^{(\ell-1)}))$. We then obtain $x^{(\ell+1)} = x^{(\ell)} - K^{(\ell)} f(x^{(\ell)})$ and evaluate $f(x^{(\ell+1)})$.

We shall extend these secant methods to counterpart multisecant methods. Suppose that we have $x^{(\ell-k)}$ and $f(x^{(\ell-k)})$, for $0 \le k \le m$, with $1 \le m \le M$ and $\ell \ge m$. We focus on $m > 1$ for multisecant methods, but would reduce to secant methods for $m = 1$. There are two natural ways to formulate direct and inverse multisecant methods. In the end, they will prove to be essentially equivalent. The literature considers primarily the second approach.

Since we think of $J^{(\ell)}$ as an approximation to $F(x^{(\ell)})$, we are led to the centered direct multisecant conditions

$$J^{(\ell)}(x^{(\ell-k)} - x^{(\ell)}) = (f(x^{(\ell-k)}) - f(x^{(\ell)})),$$

for $1 \leq k \leq m$. In order to avoid redundancy or inconsistency, we need $\{x^{(\ell-k)} - x^{(\ell)}\}_{k=1}^{m}$ to be linearly independent; in order that this be compatible with nonsingularity of $J^{(\ell)}$, we also need $\{f(x^{(\ell-k)}) - f(x^{(\ell)})\}_{k=1}^{m}$ to be linearly independent. If the labeling of $x^{(\ell-k)}$, $0 \leq k \leq m$, reflects an underlying ordering, we are led to the sequential direct multisecant conditions

$$J^{(\ell)}(x^{(\ell-k+1)} - x^{(\ell-k)}) = (f(x^{(\ell-k+1)}) - f(x^{(\ell-k)})),$$

for $1 \leq k \leq m$. Again, we need $\{x^{(\ell-k+1)} - x^{(\ell-k)}\}_{k=1}^{m}$ and $\{f(x^{(\ell-k+1)}) - f(x^{(\ell-k)})\}_{k=1}^{m}$ linearly independent. By earlier work, this means that $\{x^{(\ell-k)} - x^{(\ell)}\}_{k=1}^{m}$ and $\{x^{(\ell-k+1)} - x^{(\ell-k)}\}_{k=1}^{m}$ are deviation and difference bases for the same subspace, and $\{f(x^{(\ell-k)}) - f(x^{(\ell)})\}_{k=1}^{m}$ and $\{f(x^{(\ell-k+1)}) - f(x^{(\ell-k)})\}_{k=1}^{m}$ are deviation and difference bases for the same subspace. This also means that $\{x^{(\ell-k)}\}_{k=0}^{m}$ and $\{f(x^{(\ell-k)})\}_{k=0}^{m}$ are affinely independent.

Since we think of $K^{(\ell)}$ as an approximation to $F(x^{(\ell)})^{-1}$, we are led to the centered inverse multisecant conditions

$$K^{(\ell)}(f(x^{(\ell-k)}) - f(x^{(\ell)})) = (x^{(\ell-k)} - x^{(\ell)}),$$

for $1 \leq k \leq m$, and to the sequential inverse multisecant conditions

$$K^{(\ell)}(f(x^{(\ell-k+1)}) - f(x^{(\ell-k)})) = (x^{(\ell-k+1)} - x^{(\ell-k)}),$$

for $1 \leq k \leq m$, if the labeling reflects an underlying ordering. We also need the same linear and affine independence properties, and consequences thereof.

It will prove convenient to combine further discussion of the centered and sequential multisecant conditions, thence the deviation and difference bases, by the notational device of introducing $N \times m$ matrices $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$. For the centered multisecant conditions, define $\Delta X^{(\ell)} e_k = x^{(\ell-k)} - x^{(\ell)}$ and $\Delta F^{(\ell)} e_k = (f(x^{(\ell-k)}) - f(x^{(\ell)}))$, for $1 \leq k \leq m$. For the sequential multisecant conditions, define $\Delta X^{(\ell)} e_k = x^{(\ell-k+1)} - x^{(\ell-k)}$ and $\Delta F^{(\ell)} e_k = (f(x^{(\ell-k+1)}) - f(x^{(\ell-k)}))$, for $1 \leq k \leq m$. Note that by my indexing conventions the columns of $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$ are ordered by increasing age (decreasing $\ell - k$). Probably for historical reasons (but possibly as an artifact of indexing preferences), the sequential multisecant conditions, thence $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$, are commonly ordered by decreasing age. This has potentially adverse numerical consequences.

The direct multisecant conditions now take the form

$$J^{(\ell)} \Delta X^{(\ell)} = \Delta F^{(\ell)},$$

and the inverse multisecant conditions now take the form

$$K^{(\ell)} \Delta F^{(\ell)} = \Delta X^{(\ell)},$$

for both the centered and sequential versions. By the foregoing, we know that $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$ have maximal rank and that the ranges $R\{\Delta X^{(\ell)}\}$ and $R\{\Delta F^{(\ell)}\}$ are the same for both versions because the columns of $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$ are the deviation and difference bases thereof.

We shall obtain $J^{(\ell)}$ from $J^{(\ell-m)}$ by satisfying the direct multisecant conditions $J^{(\ell)}\Delta X^{(\ell)} = \Delta F^{(\ell)}$ and also the restriction that $J^{(\ell)}d = J^{(\ell-m)}d$, for all nonzero $d \perp R\{\Delta X^{(\ell)}\}$, or equivalently, $(\Delta X^{(\ell)})^* d = 0$. If we write

$$J^{(\ell)} = J^{(\ell-m)} + U\left[(\Delta X^{(\ell)})^*\Delta X^{(\ell)}\right]^{-1}(\Delta X^{(\ell)})^*,$$

then the restriction will be satisfied for any $N \times m$ matrix U. From

$$J^{(\ell)}\Delta X^{(\ell)} = J^{(\ell-m)}\Delta X^{(\ell)} + U,$$

we infer that the direct multisecant conditions $J^{(\ell)}\Delta X^{(\ell)} = \Delta F^{(\ell)}$ will be satisfied for

$$U = \Delta F^{(\ell)} - J^{(\ell-m)}\Delta X^{(\ell)}.$$

We shall obtain $K^{(\ell)}$ from $K^{(\ell-m)}$ by satisfying the inverse multisecant conditions $K^{(\ell)}\Delta F^{(\ell)} = \Delta X^{(\ell)}$ and also the restriction that $K^{(\ell)}d = K^{(\ell-m)}d$ for all nonzero $d \perp R\{\Delta F^{(\ell)}\}$, or equivalently, $(\Delta F^{(\ell)})^* d = 0$. If we write

$$K^{(\ell)} = K^{(\ell-m)} + U\left[(\Delta F^{(\ell)})^*\Delta F^{(\ell)}\right]^{-1}(\Delta F^{(\ell)})^*$$

then the restriction will be satisfied for any $N \times m$ matrix U. From

$$K^{(\ell)}\Delta F^{(\ell)} = K^{(\ell-m)}\Delta F^{(\ell)} + U,$$

we infer that the direct multisecant conditions $K^{(\ell)}\Delta F^{(\ell)} = \Delta X^{(\ell)}$ will be satisfied for

$$U = \Delta X^{(\ell)} - K^{(\ell-m)}\Delta F^{(\ell)}.$$

At this point, I shall introduce the simplified direct and inverse multisecant methods. In the simplified direct multisecant method, we systematically replace $J^{(\ell-m)}$ in the foregoing expressions by $-(\beta^{(\ell)})^{-1}I$; in the simplified inverse multisecant method, we systematically replace $K^{(\ell-m)}$ in the foregoing expressions by $-\beta^{(\ell)}I$. I use the word "replace" advisedly, since we have no basis for regarding $-(\beta^{(\ell)})^{-1}I$ as an approximation to $F(x^{(\ell-m)})$, or $-\beta^{(\ell)}I$ as an approximation to $F(x^{(\ell-m)})^{-1}$. The intent is simplification, not approximation. Nevertheless, we proceed on the hope and expectation that incorporation of information from the multisecant conditions will make $J^{(\ell)}$ and $K^{(\ell)}$ useful approximations to $F(x^{(\ell)})$ and $F(x^{(\ell)})^{-1}$, respectively—which may or may not be the case. In particular, I shall introduce stationary simplified multisecant methods, incorporating quasistationary and equistationary components. We shall be primarily concerned with stationary simplified inverse multisecant methods, so we shall focus on these and make brief remarks later about stationary simplified direct multisecant methods. In simplified form, we have

$$K^{(\ell)} = -\beta^{(\ell)}I + \left\{\Delta X^{(\ell)} + \beta^{(\ell)}\Delta F^{(\ell)}\right\}\left[(\Delta F^{(\ell)})^*\Delta F^{(\ell)}\right]^{-1}(\Delta F^{(\ell)})^*.$$

Define

$$\hat{c}^{(\ell)} := \left[(\Delta F^{(\ell)})^*\Delta F^{(\ell)}\right]^{-1}(\Delta F^{(\ell)})^* f(x^{(\ell)}),$$

so we have

$$\left[(\Delta F^{(\ell)})^*\Delta F^{(\ell)}\right]\hat{c}^{(\ell)} = (\Delta F^{(\ell)})^* f(x^{(\ell)}),$$

which we recognize as the normal equations for the least squares problem $\Delta F^{(\ell)} c^{(\ell)} = f(x^{(\ell)})$. From $x^{(\ell+1)} = x^{(\ell)} - K^{(\ell)} f(x^{(\ell)})$, we obtain

$$x^{(\ell+1)} = \left\{ x^{(\ell)} - \Delta X^{(\ell)} \hat{c}^{(\ell)} \right\} + \beta^{(\ell)} \left\{ f(x^{(\ell)}) - \Delta F^{(\ell)} \hat{c}^{(\ell)} \right\}.$$

We now take $f(x) = g(x) - x$, so we have

$$f(x^{(\ell)}) = g(x^{(\ell)}) - x^{(\ell)} = y^{(\ell)} - x^{(\ell)} = r^{(\ell)}.$$

We also take $\beta^{(\ell)} > 0$, and $W = I$.

For the centered inverse secant conditions and corresponding deviation basis for $R\left\{\Delta F^{(\ell)}\right\}$, we recognize that $e_k^* \hat{c}^{(\ell)} = -\hat{\theta}_k^{(\ell)}$, $1 \le k \le m$. It follows that

$$\left\{ x^{(\ell)} - \Delta X^{(\ell)} \hat{c}^{(\ell)} \right\} = x^{(\ell)} + \sum_{k=1}^{m} \hat{\theta}_k^{(\ell)} (x^{(\ell-k)} - x^{(\ell)}) = \hat{u}^{(\ell)}$$

and

$$\left\{ f(x^{(\ell)}) - \Delta F^{(\ell)} \hat{c}^{(\ell)} \right\} = r^{(\ell)} + \sum_{k=1}^{m} \hat{\theta}_k^{(\ell)} (r^{(\ell-k)} - r^{(\ell)}) = \hat{v}^{(\ell)} - \hat{u}^{(\ell)},$$

so

$$x^{(\ell+1)} = \hat{u}^{(\ell)} + \beta^{(\ell)} (\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = (1 - \beta^{(\ell)}) \hat{u}^{(\ell)} + \beta^{(\ell)} \hat{v}^{(\ell)},$$

where

$$\hat{v}^{(\ell)} = y^{(\ell)} + \sum_{k=1}^{m} \hat{\theta}_k^{(\ell)} (y^{(\ell-k)} - y^{(\ell)}).$$

We see that the simplified inverse multisecant method applied to $f(x) = g(x) - x$ yields the same $x^{(\ell+1)}$ as the Extrapolation Algorithm applied to $g(x) = x + f(x)$, for the same iterant data.

For the sequential inverse secant conditions and corresponding difference basis for $R\left\{\Delta F^{(\ell)}\right\}$, we recognize that $e_j^* \hat{c}^{(\ell)} = \hat{\xi}_j^{(\ell)}$, $1 \le j \le m$. It follows that

$$\left\{ x^{(\ell)} - \Delta X^{(\ell)} \hat{c}^{(\ell)} \right\} = x^{(\ell)} - \sum_{j=1}^{m} \hat{\xi}_j^{(\ell)} (x^{(\ell-j+1)} - x^{(\ell-j)}) = \hat{u}^{(\ell)}$$

and

$$\left\{ f(x^{(\ell)}) - \Delta F^{(\ell)} \hat{c}^{(\ell)} \right\} = r^{(\ell)} - \sum_{j=1}^{m} \hat{\xi}_j^{(\ell)} (r^{(\ell-j+1)} - r^{(\ell-j)}) = \hat{v}^{(\ell)} - \hat{u}^{(\ell)},$$

so

$$x^{(\ell+1)} = \hat{u}^{(\ell)} + \beta^{(\ell)} (\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = (1 - \beta^{(\ell)}) \hat{u}^{(\ell)} + \beta^{(\ell)} \hat{v}^{(\ell)},$$

where

$$\hat{v}^{(\ell)} = y^{(\ell)} - \sum_{j=1}^{m} \hat{\xi}_j^{(\ell)} (y^{(\ell-j+1)} - y^{(\ell-j)}).$$

The same conclusion follows.

The quasistationary simplified inverse multisecant method corresponds to $1 \le \ell = m \le M$ and the quasistationary version of the Extrapolation Algorithm. The

equistationary simplified inverse multisecant method corresponds to $m = M < \ell$ and the equistationary version of the Extrapolation Algorithm. The stationary simplified inverse multisecant method combines these components as with the stationary version of the Extrapolation Algorithm. Recall that, in formulating the inverse multisecant conditions, we required that $\{f(x^{(\ell-k)})\}_{k=0}^{m} = \{r^{(\ell-k)}\}_{k=0}^{m}$ be affine independent and also that $\{x^{(\ell-k)}\}_{k=0}^{m}$ be affine independent, in the expectation that $K^{(\ell)}$ will be a nonsingular approximation to $F(x^{(\ell)})^{-1}$. Only the affine independence of $\{r^{(\ell-k)}\}_{k=0}^{m}$ plays an explicit role in the Extrapolation Algorithm.

I categorically and emphatically reject the facile assertion that the foregoing extends the stationary Extrapolation Algorithm from fixed point problems to root-finding problems or subsumes it within the stationary simplified inverse multisecant method. Rather, I would argue, anyone considering application of the stationary simplified inverse multisecant method to the root-finding problem $f(x) = 0$ should take cognizance of the fact that this is equivalent to applying the stationary Extrapolation Algorithm to the implicit fixed point problem for $g(x) = x + f(x)$. The convergence properties of the Picard iteration for g have implications for the efficacy of both methods. There are any number of choices for f yielding the same zero $\hat{x}$. Some choices will yield a cogent g; many others will not. As a simple example anticipated above and discussed further below, replacing f by $-f$ requires also replacing $\beta^{(\ell)}$ by $-\beta^{(\ell)}$ to generate the same $x^{(\ell+1)}$. This makes sense in that replacing f by $-f$ replaces F by $-F$. Failure to replace $\beta^{(\ell)} > 0$ by $\beta^{(\ell)} < 0$ will convert a locally convergent Picard iteration into one that is not locally convergent. This is a trap for the unwary, so I forthrightly and steadfastly resist all attempts to recast the stationary Extrapolation Algorithm as a method for solving root-finding rather than fixed point problems. There are other instances where this temptation arises. The fact that $r^{(\ell)}$ is involved in the discussion of the stationary Extrapolation Algorithm (as an abbreviation!) does not mean that it can simply be replaced by $f(x^{(\ell)})$.

There is an isomorphism between the initial presentations above of the direct and inverse multisecant methods, involving interchange of the roles of $\Delta X^{(\ell)}$ and $\Delta F^{(\ell)}$. However, for the direct version, we are not actually interested in $J^{(\ell)}$ because the solution of $J^{(\ell)}(x^{(\ell+1)} - x^{(\ell)}) = -f(x^{(\ell)})$ to obtain $x^{(\ell+1)}$ is prohibitively costly. Rather, we are interested in $(J^{(\ell)})^{-1}$, so $x^{(\ell+1)} = x^{(\ell)} - (J^{(\ell)})^{-1} f(x^{(\ell)})$. An advantage of the simplified (but also the unsimplified) direct multisecant method is that the well-known Sherman-Morrison-Woodbury formula can be used to derive an expression for $(J^{(\ell)})^{-1}$ from that for $J^{(\ell)}$. For $A \in \mathbb{C}_n^{n\times n}$; $U, V \in \mathbb{C}^{n\times m}$; $S, T \in \mathbb{C}_m^{m\times m}$, with $m < n$ and $T := S^{-1} \pm V^* A^{-1} U$, we have the Sherman-Morrison-Woodbury formula $(A \pm USV^*)^{-1} = A^{-1} \mp (A^{-1}U)T^{-1}(V^*A^{-1})$. For $m = 1$, we obtain the Sherman-Morrison formula by replacing U, V by $u, v \in \mathbb{C}^n$, and replacing S, T by $\sigma, \tau \in \mathbb{C}$, assumed nonzero. There is again a linear equation to be solved to obtain $x^{(\ell+1)}$ but it no longer takes the form of the normal equations for a least squares problem. We shall not pursue the details here; they may be found in papers discussed below.

These papers will be considered in two groups whose members can usefully be compared and contrasted on matters related to our themes. The first group consists of Eyert [7], Marks and Luke [11], Fang and Saad [8], and Calef, Fichtl, Warsa and

Carlson [5], which will be abbreviated hereafter as Calef et al. [5]. The second group consists of Walker and Ni [16], Ni [12], and Toth and Kelley [15], plus Calef et al. [5].

It should be understood from the outset that issues raised in the context of a particular paper may arise there and be of interest because of their implications should they be taken as a model for later work by others. However, these issues may arise in earlier papers and/or be included in many other related papers; their discussion in this context simply reflects the choice to consider this paper.

It used to be commonplace, and still is in some contexts, to solve positive definite linear equations without scaling, pivoting, or regularization, since numerical stability can be established. Many utility codes for this purpose were so constructed, ordinarily using the Cholesky or Turing factorization. For authors using the normal equations, I shall assume that scaling, pivoting, or regularization were not used if no mention was made that they were used. When scaling or pivoting is mentioned without further specification, I shall assume that the corresponding standard strategy was employed either when a QR decomposition or factorization approach or the normal equations approach was involved.

6.2 Eyert

Eyert [7] introduces the fixed point context of the discussion without specifically introducing g (and later uses $G^{(\ell)}$ where I have used $K^{(\ell)}$). However, Eyert adopts my $x^{(\ell)}$ and $y^{(\ell)} = g(x^{(\ell)})$, and the equivalent of $x^{(\ell+1)} = (1-\beta^{(\ell)})\hat{u}^{(\ell)} + \beta^{(\ell)}\hat{v}^{(\ell)}$, with $\beta^{(\ell)} > 0$, though with $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$ replaced by $\bar{x}^{(\ell)}$ and $\bar{y}^{(\ell)}$, respectively. He also adopts the abbreviation $r^{(\ell)} = y^{(\ell)} - x^{(\ell)}$, though with the residual $r^{(\ell)}$ replaced by $F^{(\ell)}$ and $\hat{v}^{(\ell)} - \hat{u}^{(\ell)}$ by $\bar{F}^{(\ell)}$, so $x^{(\ell+1)} = \bar{x}^{(\ell)} + \beta^{(\ell)}\bar{F}^{(\ell)}$. The upshot is that he is thinking throughout in terms of the fixed point problem for g and the associated zero-finding problem for $g(x) - x$. This matters later.

There is some terminological confusion in the literature. Eyert reviews what I have called the stationary Extrapolation Algorithm under the label Anderson Mixing. The case $M = 0$, so $x^{(\ell+1)} = (1-\beta^{(\ell)})x^{(\ell)} + \beta^{(\ell)}y^{(\ell)}$, is called simple mixing, by Eyert and others, and $\beta^{(\ell)}$ is correspondingly called the mixing parameter. For $M > 0$, Eyert appears to think of $x^{(\ell+1)} = (1-\beta^{(\ell)})\bar{x}^{(\ell)} + \beta^{(\ell)}\bar{y}^{(\ell)}$, as mixing $\bar{x}^{(\ell)}$ and $\bar{y}^{(\ell)}$. In the physics community, the original impetus to introduce simple mixing, typically with an empirical $\beta^{(\ell)} = \beta \sim \frac{1}{2}$, was to damp out oscillatory behavior from one Picard iterant to the next. Other authors, especially those taking $\beta^{(\ell)} = \beta = 1$, so there is no mixing per se, think of the affine combination coefficients, or the equivalent thereof, as the mixing coefficients; if $\beta^{(\ell)} = \beta \neq 1$, is used, this is thought of as redefining the fixed point problem in an attempt to ensure or enhance the convergence of the associated Picard iteration.

Eyert also reviews secant and multisecant methods focusing finally on what I have called the stationary simplified inverse multisecant method. Much of the paper is devoted to sorting out issues related to variants in the physics literature deriving from the minimization characterization of multisecant methods, but we shall not pursue these matters here. Subsequently, Eyert demonstrates that Anderson Mixing (that is, the stationary Extrapolation Algorithm) is isomorphic to the stationary simplified inverse multisecant method.

Recall that it is customary for historical reasons to formulate multisecant methods using the sequential secant conditions, and to order the resulting difference basis by decreasing age of the iterant data; Eyert followed these customs in his review. I used the deviation basis ordered by increasing age of the iterant data in the Extrapolation Algorithm; Eyert followed these customs in his review. I used the centered and sequential secant conditions and corresponding deviation and difference bases ordered by increasing age of the iterant data in my presentation of multisecant methods above and showed earlier that the two bases have the same span. Eyert's construction of the correspondence between the two bases differs from mine in that the ordering of our difference bases is reversed. Eyert pursues computational aspects of Anderson Mixing using the difference basis ordered by decreasing age rather than the deviation basis ordered by increasing age. This matters later. Eyert uses the normal equations without scaling or pivoting, but he does use regularization, with the equivalent of D given by $e_k^* De_k = \mu \parallel Ae_k \parallel_2$, $1 \leq k \leq \min(\ell, M)$, and a relatively large $\mu = 10^{-2}$. With the standard scaling, $\parallel Ae_k \parallel_2 = 1$, this would reduce to broad regularization, as defined above. Without scaling, this approach is a common, though not universal, surrogate.

The computational examples in Eyert [7] use a small ($N = 5$), simple ($F(x)$ diagonal and negative definite) and weakly nonlinear test problem, $f(x) = 0$, with $\hat{x} = 0$ and $x^{(0)} = e$. We see that $G(\hat{x}) = I + F(\hat{x})$, thence $G(\hat{x} \mid \beta) = I + \beta F(\hat{x})$. This problem falls within the framework used for studying the examples discussed above, though with $\rho(G(\hat{x} \mid 1)) = 2$. Thus, the Picard iteration for g, corresponding to $\beta = 1$, is not locally convergent. Nevertheless, there is an optimal choice $\hat{\beta} = 4/7 \approx 0.571$ with $\rho(G(\hat{x} \mid \hat{\beta})) = 5/7 \approx 0.714$. Thus, the Picard iteration for $g(x \mid \hat{\beta})$ is locally convergent. We also have $\rho(G(\hat{x} \mid 1/2)) = 3/4$ and $\rho(G(\hat{x} \mid 1/5)) = 9/10$, so the Picard iterations for $g(x \mid 1/2)$ and $g(x \mid 1/5)$ are locally convergent. Figures 5, 7, and 8 in Eyert [7] portray results for a range of M with $\beta = 1, 1/2$, and $1/5$, respectively. The results for $M = 0$ correspond to the Picard iteration for $g(x \mid \beta)$ and are consistent with the local convergence properties noted above: divergence for $\beta = 1$, convergence for $\beta = 1/2$ and $\beta = 1/5$, with more rapid convergence for $\beta = 1/2$. For $1 \leq M \leq 4$, the results portray convergence, somewhat erratic for $\beta = 1$, smoother and more rapid for $\beta = 1/2$ and $\beta = 1/5$, the moreso for $\beta = 1/2$ and for increasing M. For $M = N$, the equistationary Extrapolation Algorithm is equivalent to the Wolfe formulation of the classical secant method: see Ortega and Rheinboldt [13]. One would expect the results for $M = 5$ to be unusually and unrepresentatively rapidly convergent, because $M = N$, and this proves to be the case. Unexpectedly, the results for $M = 4$ essentially coincide with those for $M = 5$ in Figures 5, 6, and 7, though not in Figure 8. The meaning of the results portrayed for $M = 6$ (and essentially coinciding with those for $M = 5$) is unclear, since the method is well-defined only for $0 \leq M \leq N$. This is related to the regularization and to matters we have chosen above not to discuss and may safely be ignored for our purposes. While the example is trivial when compared to the challenging problems with $1 \ll M \ll N$ motivating our discussion above, the dependence on Picard iteration convergence mirrors that encountered more broadly.

Figure 6 in Eyert [7] portrays counterpart results for the stationary simplified direct multisecant method to those in Figure 5 for Anderson Mixing, thence the stationary simplified inverse multisecant method. The performance of the inverse and direct methods is very similar, with the inverse method slightly more rapidly convergent for $1 \leq M \leq 3$. For this small test problem, the approximate Jacobian in the direct method was simply inverted (or the equivalent) when formed, presumably paralleling the formation of the approximate inverse Jacobian.

6.3 Conceptual issues

We shall now proceed to discussion of the rest of the first group of papers: Marks and Luke [11], Fang and Saad [8], and Calef et al. [5]. The work reported in the first two papers is contemporaneous but disjoint. Marks/Luke report that an anonymous referee brought a 2007 Fang/Saad technical report to their attention, which they reference, but they apparently did not have access to this document. There is no indication that Fang/Saad were aware of the Marks/Luke work, which has largely been ignored by the Applied Mathematics community. (I have no information about its reception in the Physics community.) Both papers share an undetected sign error, whose genesis differs but whose consequences are equivalent. The Calef et al. paper is included in this group because it clearly exhibits the potentially adverse impact of that sign error, which may not be readily apparent in all problems. In different ways, the work of Eyert is relevant to all three papers.

6.3.1 Marks/Luke

Surprisingly, Marks/Luke appear to be unaware of the Eyert paper; had a referee brought this reference to their attention, I believe that their paper would have been significantly improved. Marks/Luke consider the fixed point problem $g(x) = x$ and the associated root-finding problem $g(x) - x = 0$. Ironically, if they had considered the root-finding problem $x - g(x) = 0$ instead, no sign error would have arisen. The scientific context is formulated in terms of fixed point problems; the mathematical discussions are essentially phrased in terms of root-finding problems, since the special properties of fixed point problems play no role. Basically, they use the centered secant conditions to derive stationary simplified direct and inverse multisecant methods, with $M = 8$ and $\beta^{(\ell)}$ replaced by $-\sigma^{(\ell)} < 0$. They do not connect the resulting inverse algorithm to Anderson Mixing using the deviation basis. The sign error arises from the fact that if the Picard iteration for $g(x)$ is locally convergent then the Picard iteration for $(1 - \beta^{(\ell)})x + \beta^{(\ell)} g(x) = (1 + \sigma^{(\ell)})x - \sigma^{(\ell)} g(x)$ is not locally convergent, the moreso as $\sigma^{(\ell)} > 0$ increases.

The Marks/Luke discussion of relevant features of the motivating electronic structure problems is excellent. However, I find some more mathematical Marks/Luke arguments unpersuasive. In particular, based on the geometry of the situation, they argue that $\sigma^{(\ell)}$ must be nonzero but should be small. Absent the sign error, we know that $\beta^{(\ell)} \sim 1$ may be optimal if the Picard iteration for g is locally convergent.

A smaller positive $\beta^{(\ell)}$ may cope with a Picard iteration for g that is not locally convergent by producing a Picard iteration for $(1 - \beta^{(\ell)})x + \beta^{(\ell)}g(x)$ which is locally convergent or may just mute divergence sufficiently to enable the Extrapolation Algorithm to generate an approximate fixed point. Whether the latter constitutes convergence in a mathematical sense is an open question.

6.3.2 Fang/Saad

The Fang and Saad [8] paper can usefully be divided into three parts: motivating introduction (Section 1), analytical developments (Sections 2 and 3), and computational considerations (Sections 4 and 5), plus a final summary (Section 6). Fairly or not, I choose to attribute the first part to the second author and the remainder to the first author, there being an apparent disjunction. Fang/Saad consider the fixed point problem $g(x) = x$ and the associated root-finding problem $x - g(x) = 0$. The first part provides a nice survey of some of the salient issues related to the solution of large, nonlinear fixed point problems arising in electronic structure calculations, with which the second author has been involved. Essentially, the root-finding problem serves as an abbreviation. This survey also draws upon Bierlaire and Crittin [2] where counterpart problems are encountered in transportation systems. (Marks/Luke reference an earlier conference proceeding contribution by the same authors.) This work uses the centered secant conditions in a radically different fashion unsuited for our purposes so we shall not go into further detail. We note, however, that some issues may be more (or less) salient in one problem context than in another. The second and third parts focus almost exclusively on the root-finding problem $f(x) = 0$, with no substantive mention of fixed point problems.

In Section 2, direct and inverse secant and multisecant methods are reviewed, and a related Nonlinear Eirola-Nevalinna-like method (which we shall not go into) and a preliminary version of Anderson Mixing are introduced. In Section 3, these methods are extended in various ways, leading to a large collection of potential methods of different types, families, and classes. Selected members of this collection were implemented and tested, as reported in part 3. We shall focus here only on their final version of Anderson Mixing.

Their preliminary version of Anderson Mixing is a restatement of the equistationary version as reformulated by Eyert in terms of the difference basis ordered by decreasing age. This is characterized as "a procedure for solving a large nonlinear system of equations $f(x) = 0$ by an iterative process." Eyert is credited with establishing the equivalence of the stationary version of Anderson Mixing and the stationary simplified inverse multisecant method (in my language). Their final version of Anderson Mixing is the quasistationary version, formally obtained by taking $M = \infty$. Since the method is undefined for $M > N$, there is a tacit assumption that the iteration will terminate for some $\ell \leq M = N$.

Eyert was quite clear that the stationary version with relatively small M was his intended approach and that solving the fixed point problem for $g(x)$ was the goal, using the root-finding problem $g(x) - x = 0$ just as a convenient abbreviation. Fang/Saad use the root-finding problem $x - g(x) = 0$. They later mention related invariance issues, but apparently fail to recognize that reversing the sign of the

residual will not leave the algorithm invariant unless the sign of $\beta^{(\ell)}$ is also reversed. In part 3, they continue to use $\beta^{(\ell)} > 0$, so they are applying Anderson Mixing to the fixed point problem for $(1 + \beta^{(\ell)})x - \beta^{(\ell)}g(x)$, not $(1 - \beta^{(\ell)})x + \beta^{(\ell)}g(x)$. If the Picard iteration for $g(x)$ is locally convergent, that for $(1 + \beta^{(\ell)})x - \beta^{(\ell)}g(x)$, with $\beta^{(\ell)} > 0$, is not locally convergent. This is the aforementioned sign error. The use of the quasistationary version throughout is also problematic, especially for large ℓ.

6.3.3 Calef et al.

We turn to Calef et al. [5] for insight on the consequences of the sign error in Marks and Luke [11] and in Fang and Saad [8] . We shall return to computational aspects of these three papers subsequently.

I intend this discussion to be taken as emblematic of the imperative need to think of Anderson Acceleration or mixing as a method for fixed point problems rather than root-finding problems. When stationary simplified inverse multisecant methods are applied to the root-finding problem $f(x) = 0$, it should be fully appreciated that this is equivalent to applying the stationary Extrapolation Algorithm to the implicit fixed point problem for $x + f(x)$; due attention should be paid to the fact that convergence properties of the Picard iteration for $x + f(x)$ are a relevant consideration.

Calef et al. consider the fixed point problem for $g(x)$ and the associated root-finding problem $f(x) = x - g(x) = 0$. The motivating assumption (valid by design for all three test cases) is that the Picard iteration for $g(x)$ is convergent. However, most of the discussion is carried on in the root-finding framework. The scientific context is neutron transport theory for nuclear reactors.

The nonlinear Krylov acceleration method is described in Carlson and Miller [6], but has been in use since 1990. The original motivating assumption was that $f(x)$ had been preconditioned so that $F(x) \approx I$, so $G(x) = I - F(x)$ is small. In my language, as used above, this is a stationary simplified inverse multisecant method, with $\beta^{(\ell)} = 1$. The sequential secant conditions are used, but are negated: backward differences rather than forward differences. Therefore, the negated difference basis is used, ordered by increasing age rather than the more conventional decreasing age. Anderson Mixing is introduced with the deviation basis, but again negated. The fact that the negated difference basis and the negated deviation basis are bases for the same subspace is simply asserted. The stationary simplified direct multisecant method, with $\beta^{(\ell)} = 1$, using the negated difference basis ordered by increasing age, is associated with Broyden. The first item on the Calef et al. agenda is to demonstrate that nonlinear Krylov acceleration is mathematically equivalent to Anderson Mixing with $\beta^{(\ell)} = 1$. The second item on the Calef et al. agenda is discussion of an issue arising in the Walker/Ni paper. We shall postpone consideration of this item until we review the second group of publications. We shall also consider computational matters related to Calef et al. [5] together with those for Marks and Luke [11] and Fang and Saad [8].

The third item on the Calef et al. agenda and the computational results related thereto are of immediate conceptual interest for our purposes. Recall that both Fang/Saad and Calef et al. use the root-finding problem $x - g(x) = 0$ to study the fixed point problem for $g(x)$. Calef et al. review the formulation of Anderson Mixing

in Fang and Saad [8], translated into the Calef et al. notation, detect the aforementioned sign error, and note that Fang/Saad use a positive $\beta^{(\ell)}$ rather than a negative $\beta^{(\ell)}$. In a footnote, they indicated that they believe that Fang/Saad implicitly modified the formula given for $x^{(\ell+1)}$ to correct the sign error. I am inclined to believe otherwise.

Calef et al. heroically set out to illustrate what happens when a $\beta^{(\ell)}$ of the "wrong" sign as chosen by running all of their test cases using their original nonlinear Krylov acceleration method (equivalent to Anderson Mixing with $\beta^{(\ell)} = 1$) and a modified version (equivalent to Anderson Mixing with $\beta^{(\ell)} = -1$). I say"heroically" because their test cases are computationally challenging: $M = 0, 5, 10, 20$, and 30 for $N = 1.2 \times 10^6$, 8.0×10^6, and 6.5×10^8. The contrast to the Eyert test case is stark, but one can learn from both. The Picard iteration for $g(x)$, corresponding to $\beta^{(\ell)} = 1$ is locally convergent at the fixed point $\hat{x}$. The Picard iteration for $2x - g(x)$, corresponding to $\beta^{(\ell)} = -1$, is therefore (as we have seen above) not locally convergent at $\hat{x}$; in fact, all eigenvalues of the associated Jacobian have modulus greater than one and positive real parts.

Based on previous experience, for the $\beta^{(\ell)} = 1$ case, one would expect to see smooth, steady reduction in the residual norm, at a rate increasing with M, with a significant rate increase for smaller M, but "plateauing" with a slower increase for larger M. Though the costs per iteration are dominated by that of evaluating $g(x^{(\ell)})$, thence $f(x^{(\ell)})$, the acceleration has costs that increase with M, for a given N, and there may be a best choice for M. More precisely, some acceleration costs, for a given N, will increase with M^2 or M^3, with N as an overall multiplicative factor. The results reported are generally consistent with these expectations.

For the $\beta^{(\ell)} = -1$ case, we have a competition between the Picard iteration pushing $x^{(\ell)}$ and $y^{(\ell)}$ apart, and the acceleration process pulling $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$ together. I would anticipate erratic oscillatory behavior of the residual norm as the iteration proceeds, with an amplitude roughly proportional to a local moving average of the residual norm values, and decreasing as M increases. I note that on the figures, the residual norms are plotted only for every second, or in some cases every sixth, iteration; this might mute signs of oscillation. The Eyert results in Figure 5, though primitive, may be indicative. See also Marks/Luke, Figure 2(e) and Fang/Saad, Figure 6. The Calef et al. figures suggest that a standoff may arise, with the rate of decrease of the residual norms becoming smaller, so the residuals tend to level off near a value which is smaller for larger M. In the most challenging problem, convergence is reported for no value of M considered; in the less challenging problems, convergence is reported after many iterations for some of the larger values of M. I suspect that this is a computational convergence in the sense that the residual norm is decreased enough to pass the termination test, yielding an approximate fixed point, but does not represent mathematical convergence, except by accident. The upshot is that the convergence properties of the Picard iteration for the explicit or implicit fixed point problem involved matter, at least for nonlinear problems. However, even if this Picard iteration is not convergent, using a small but nonzero $\beta^{(\ell)}$, of either sign, might allow the acceleration process to produce an approximate fixed point, using a reasonable value of M.

The Broyden method results were uniformly poorer than the Anderson Mixing with $\beta^{(\ell)} = 1$ results, though in some cases comparable for larger M. The behavior of the residual norms was erratic and oscillatory, especially for smaller M, and often diverged. This is consistent with other observations that inverse multisecant methods are preferable to direct multisecant methods, though direct secant methods were preferred to inverse secant methods originally.

6.4 Computational procedures

We turn now to discussion of computational procedures and results for Marks and Luke [11], Fang and Saad [8], and Calef et al. [5]. All three consider both direct and inverse multisecant methods. We shall focus on the inverse methods and on ideas worth noting in the light of our earlier discussion, and not on details best studied in the papers themselves. We shall begin with Calef et al., because we have already discussed most of the computational results for their conceptual value, and because our discussion of computational procedures is brief.

6.4.1 Calef et al.

In addition to the methods noted above, Calef et al. also consider the well-known Jacobian-Free Newton Krylov method, which we shall not go into here. In many contexts, this is regarded as the method to beat. In the Calef et al. context, it is beaten by nonlinear Krylov acceleration (that is, Anderson Mixing with $\beta^{(\ell)} = 1$).

Calef et al. use the normal equations, $\tilde{A}^*\tilde{A}\tilde{c} = \tilde{A}^*\tilde{b}$, with the counterpart of the standard scaling strategy, $\| \tilde{A}e_k \|_2 = 1$, solved by Cholesky factorization without pivoting: $\tilde{A}^*\tilde{A} = C^*C$, $\tilde{c} = C^{-1}(C^*)^{-1}d$, where $d = \tilde{A}^*\tilde{b}$. Note that it is tacitly assumed that $Ae_k \neq 0$, for $1 \leq k \leq \min(\ell, M)$. Recall that they are using the negated difference basis ordered by increasing age. The description of their approach to coping with potential ill-conditioning (near-linear-dependence) is concise and cryptic. To explain and discuss it, I shall present it as I would implement it, which may not be the way they did. Recall that if $\tilde{A} = \hat{Q}\hat{R}$ is the standard QR factorization of a maximal rank $\tilde{A}$ then $C = \hat{R}$. We identify $e_1^*Ce_1 = 1$. For $1 < k \leq \min(\ell, M)$, we identify $e_k^*Ce_k$ as the magnitude of the sine of the angle between $\tilde{A}e_k$ and $spn\left\{\tilde{A}e_1, \tilde{A}e_2, \ldots, \tilde{A}e_{k-1}\right\} = spn\left\{\hat{Q}e_1, \hat{Q}e_2, \ldots, \hat{Q}e_{k-1}\right\}$. If $e_k^*Ce_k$ falls below a specified small positive threshold, then $\tilde{A}e_k$ is declared to be sufficiently nearly linearly dependent on $\tilde{A}e_j$, $1 \leq j \leq k-1$, to be disregarded or deleted. The question is how to do so conveniently, recognizing that several such k might arise as the process proceeds. Note that $\tilde{A}e_1$ is always included.

One possibility is to modify $\tilde{A}^*\tilde{A}$ and $\tilde{A}^*\tilde{b}$ by removing $\tilde{A}^*\tilde{A}e_k$, $e_k^*\tilde{A}^*\tilde{A}$, and $e_k^*\tilde{A}^*\tilde{b}$ and continuing with the smaller problem with $e_k^*\tilde{c}$ removed. This would also require keeping track of all k for which this situation arose to interpret the results. I prefer to proceed as follows: modify $\tilde{A}^*\tilde{A}$ and $\tilde{A}^*b$ by replacing $\tilde{A}^*\tilde{A}e_k$ by e_k, replacing $e_k^*\tilde{A}^*\tilde{A}$ by e_k^*, and replacing $e_k^*\tilde{A}^*b$ by 0. Clearly, the solution of the modified linear equation will have $e_k^*\tilde{c} = 0$. Correspondingly, we will have a modified C and

d with $Ce_k = e_k$, $e_k^* C = e_k^*$, and $e_k^* d = 0$. We continue to work with matrices and vectors of the same size without needing to keep track of all k or move data around in computer storage. The result is to find a basic solution of the underlying least squares problem, determined by a particular strategy to avoid near-linear-dependence (ill-conditioning). One should probably use the column-oriented algorithm that forms C column-by-column, essentially processing the leading principal submatrices in order. There is no indication whether this approach actually played a role in connection with the computational results presented.

If $\tilde{c}$ is a basic solution, the elements (or variables) which are necessarily zero are termed nonbasic, and those which are ordinarily expected to be nonzero, but which need not necessarily be nonzero, are termed basic. The number of basic variables is the effective rank. The basic solution generated by the foregoing and that generated using the standard pivoting strategy may have different basic variables, thence nonbasic variables, and even different numbers thereof. Because the approach above is somewhat more restrictive about declaring variables nonbasic than the pivoting approach, one would expect the number of basic variables to be at least as large. On the other hand, the approach above, for the same reason, will tend to privilege the choice of variables corresponding to older data as nonbasic more than the pivoting approach. One of the putative advantages of the difference basis (or its negation) in comparison with the deviation basis is that the differences do not need to be recomputed from one iteration to the next so long as they remain part of the basis. However, the difference basis may be more nearly linear dependent than the deviation basis, especially in the later stages of monotonic convergence. Observe that if a variable is declared nonbasic in the approach above, it will be declared nonbasic in subsequent iterations until it ages out of further consideration.

Recall that, except for the youngest and oldest ones, declaring the coefficient associated with a difference basis vector nonbasic does not correspond to disregarding a particular residual, as would be the case for the deviation basis. For the negated difference basis ordered by increasing age, we have before scaling

$$Ae_k = r^{(\ell-k)} - r^{(\ell-k+1)}$$

and

$$Ae_{k+1} = r^{(\ell-k-1)} - r^{(\ell-k)},$$

for $1 < k < \min(\ell, M)$. Observe that

$$Ae_k + Ae_{k+1} = r^{(\ell-k-1)} - r^{(\ell-k+1)}.$$

If the scale factors $\| Ae_k \|_2$ and $\| Ae_{k+1} \|_2$ by which Ae_k and Ae_{k+1} are divided to obtain $\tilde{A}e_k$ and $\tilde{A}e_{k+1}$ are available, a linear combination of the latter pair will produce the sum of the former pair. If, in addition to modifying $\tilde{A}^*\tilde{A}e_k$, $e_k^*\tilde{A}^*\tilde{A}$ and $e_k^*\tilde{A}^*\tilde{b}$, we also relatively straightforwardly modify $\tilde{A}^*\tilde{A}e_{k+1}$, $e_{k+1}^*\tilde{A}^*\tilde{A}$ and $e_{k+1}^*\tilde{A}^*\tilde{b}$, we could alter the system to correspond to deletion of $r^{(\ell-k)}$.

6.4.2 Marks/Luke

We turn now to computational aspects of Marks and Luke [11]. They present results for a version of the direct and inverse secant methods, and for the stationary simplified direct and inverse multisecant methods, with $M = 8$. For the multisecant methods, they are using the deviation basis; during the quasistationary phase, as a matter of convenience, this is presumably ordered by increasing or decreasing age. However, during the equistationary phase, data is discarded not based on age but on proximity to the most recent iterant, discarding the most distant. They use the normal equations without pivoting, employing broad regularization with the equivalent of $\mu = 10^{-2}$, invoking several motivations. They begin by incorporating the standard scaling strategy, observing that this is needed for sensible use of broad regularization (and of the standard pivoting strategy had they employed it). However, they go on thereafter to dynamically choose a $W^{(\ell)}$ based on a measure of the relative size of the norms of two designated subvectors of the residuals associated with different modes of discretization. There appear to be both scaling and volatility issues involved. In our discussion above, a static W was introduced at the outset, before scaling. This reflects the different views of the role of W or $W^{(\ell)}$ which are outlined previously.

Marks/Luke also introduce an adaptive approach to choosing $\sigma^{(\ell)} = -\beta^{(\ell)} > 0$: the sign error. Recall that they have conceptual reasons for believing that $\sigma^{(\ell)}$ ought to be small (which are independent of the sign error). They set upper limits on $\sigma^{(\ell)}$ based on geometric considerations, and in absolute terms ($\sim 0.1 - 0.2$). Adaptive adjustments subject to these limits are multiplicative and based on $\| r^{(\ell-1)} \|_2 / \| r^{(\ell)} \|_2$; this is more aggressive than the arithmetic adaptive adjustments of $\beta^{(\ell)}$ described above. Whether their satisfactory use of small $\sigma^{(\ell)}$ is due to the sign error or to the nature of their fixed point problems is unclear. We know that if g has a locally convergent Picard iteration then $\beta^{(\ell)} \sim 1$ often makes good sense, and that $\beta^{(\ell)}$ too small does not.

Marks/Luke present results for five examples of increasing complexity and difficulty. In the simplest example, all four methods succeed with the direct methods performing better than the inverse methods, though not by much. In the three intermediate examples, the multisecant methods dominate the secant methods, and the inverse methods dominate the direct methods. For the most challenging example, only the inverse multisecant method succeeds. Especially in this last case where more iterations are involved, the anticipated erratic oscillatory behavior of the residual norms is in evidence, but $M = 8$ suffices to produce convergence with small $\sigma^{(\ell)}$, for the inverse multisecant method: that is, Anderson Mixing with small negative $\beta^{(\ell)}$.

6.4.3 Digression

At this point, I shall digress briefly to highlight certain issues lying behind a cryptic remark of Marks/Luke which has wider import. Many mathematical models of scientific problems involve constraints, either explicitly or implicitly. By implicitly, I mean that the solution of the model problem will automatically satisfy the constraint because the constraint was incorporated in formulating the model. By explicitly, I mean that satisfying the constraint is part of the task of finding the solution of the

model problem. Typical examples of constraints are symmetries and conservation laws (another form of symmetry). Symmetries are usually built into the model and thence implicit. Conservation laws require that functionals of the state variable(s) characterizing the solution must have a specified value; commonly, there are families of conservation law constraints parameterized by the assigned value. Conservation law constraints may be explicit or implicit with regard to the model problem and subproblems thereof, but may be explicit (implicit) for the model problem and implicit (explicit) for the subproblem. For our purposes, we focus on fixed point subproblems.

The state variable of the Marks/Luke problem is an electronic charge density. The integral of the charge density over the domain of interest is the total charge; in units of the electron charge, the total charge must be equal to the number of electrons involved in the problem, so this constraint is explicitly part of the model problem. Within the algorithm which generates the value of $g(x)$, for any given x, a univariant root-finding problem is solved for a parameter in the putative charge density $g(x)$ which arranges that the total charge has the specified value. Thus, the conservation law constraint is implicit with respect to the fixed point subproblem, and must be satisfied by any fixed point. I shall call $g(x)$ strongly preservative in the sense that $g(x)$ satisfies the constraint whether or not x does. I would term $g(x)$ weakly preservative if $g(x)$ satisfies the same conservation law constraint as is satisfied by x, which might correspond to a parameter different from the assigned value. Observe that if the initial iterant $x^{(0)}$ satisfies the correct conservation law constraint, then all of the Picard iterants generated by $x^{(\ell+1)} = g(x^{(\ell)})$, for $\ell = 0, 1, \ldots$, will continue to do so, provided $g(x)$ is either strongly or weakly preservative.

Our interest here focusses on the behavior of the accelerated iteration generated by applying the Extrapolation Algorithm. In the Marks/Luke problem, the conservation of total charge involves a linear functional of the charge density, so the constraint equation is affine. If all of the $y^{(\ell-k)} = g(x^{(\ell-k)}), 0 \leq k \leq m^{(\ell)}$ satisfy the constraint equation, then $\hat{v}^{(\ell)}$ as an affine combination thereof will do likewise. If all of the $x^{(\ell-k)}, 0 \leq k \leq m^{(\ell)}$ satisfy the constraint equation, then $\hat{u}^{(\ell)}$ as an affine combination thereof will do likewise. Then $x^{(\ell+1)}$ as an affine combination of $\hat{u}^{(\ell)}$ and $\hat{v}^{(\ell)}$ for $\beta^{(\ell)} \neq 1$, and equal to $\hat{v}^{(\ell)}$ for $\beta^{(\ell)} = 1$, will also do likewise. This is what underlies the Marks/Luke passing remark that total charge is conserved.

The point of going into this is that conservation laws in other problem contexts often involve quadratic or more general nonlinear functionals, which will not remain valid when affine combinations of iterants are evaluated. Ad hoc devices to restore conservation may be appropriate, even if g is strongly preservative, and should be seriously considered if $g(x)$ is weakly preservative, or neither.

The foregoing has been presented in the framework of the continuous model problem, a discretized conservation law constraint may be explicit or implicit within the discretized problem, and similar considerations apply. However, there will be additional numerical errors when the discretized problem is solved computationally.

6.4.4 Fang/Saad

We turn now to computational considerations in Part 3 of Fang and Saad [8]: Sections 4 and 5. Section 4 is problematic in a number of respects; we focus primarily

on aspects pertinent to their version of Anderson Mixing. They say that "regularized Householder QR factorization with complete pivoting" was used. They are using the $AP = \hat{Q}\hat{R}$ factorization, derived from the $AP = QR$ decomposition generated using Householder matrices. (Note that they follow the common custom of using the terms "factorization" and "decomposition" synonymously and ambiguously, whereas I do not.) I interpret "complete pivoting," later recharacterized as column pivoting, to mean the standard pivoting strategy, without scaling. They do not mean "regularization" in the senses used above; rather, they mean that if all elements of $\hat{R}_{22}$ have magnitude smaller than the unit roundoff error times $\mid e_1^* \hat{R} e_1 \mid$, they are treated as indistinguishable from zero to determine an effective rank.

Finally, recall three features of their characterization of Anderson Mixing discussed above: The first feature is the sign error associated with using $\beta^{(\ell)} = \beta > 0$ together with residuals based on the root-finding problem $x - g(x) = 0$, or the equivalent. The second feature is the use of the difference basis ordered by decreasing age, in forming A. The third feature is identifying Anderson Mixing with the quasistationary Extrapolation Algorithm by effectively taking $M = N$.

In Section 4, the anticipation of encountering near linear dependence or ill-conditioning in Anderson Mixing, detected as an effective rank less than ℓ, leads to a proposed response which we shall examine before discussing these issues more broadly. The algorithm proposed calculates the basic least squares solution associated with the scaling and pivoting strategy and choice of the nonmaximal effective rank, but calls it the minimal solution of the least squares problem. Moreover, the subsequent discussion uses the language of Moore-Penrose pseudoinverses, which would be appropriate only for the minimal solution. We argued above that the basic and minimal least squares solutions associated with a given scaling and pivoting strategy and choice of nonmaximal effective rank coincided only in one unlikely circumstance. As we have seen previously, the basic least squares solution may, for our purposes, be as interesting as the minimal least squares solution, or even moreso. Extending this approach to singular or nearly singular linear equations, as in direct rather than inverse multisecant methods, is much less sensible.

We turn now to the broader picture, seeking lessons relevant to a wider range of problems. Recall that for nonlinear, especially strongly nonlinear, problems, we wish to privilege younger over older iterant data because information pertinent locally near the fixed point is more relevant and representative. This may be less important for affine fixed point problems. Allowing for oscillations, if the iteration is converging, we typically expect the columns of A corresponding to younger iterant data to have smaller norms than those corresponding to older iterant data; the moreso for the difference rather than the deviation basis, and for rapid convergence. When A has more columns, we expect more likelihood of encountering near linear dependence; the moreso in the later stages of a convergent iteration. The stringent test for declaring nonmaximal effective rank leaves room for near linear dependence to emerge.

Mathematically, if A has maximal rank, then scaling does not affect the unique least squares solution $\hat{c}$ of $Ac = b$. If A is well-conditioned, so the solution is well-determined, then scaling and pivoting affect the numerical solution only marginally. If A has nearly linearly dependent columns, then scaling and pivoting affect the ill-determined approximate numerical solution selected. Pivoting without scaling and

with columns ordered by decreasing age and roughly by decreasing norm will tend to privilege older iterant data; pivoting with scaling and with columns ordered by increasing age will tend to privilege younger iterant data. This is of maximal impact for a basic least squares solution when iterant data is disregarded; for a minimal least squares solution, all iterant data would contribute, but older data may do so excessively.

I believe that using a moderately small M makes more sense and would avoid some of the issues related to near linear dependence. So far as I am aware, the experience of others is consistent with this opinion.

Fang/Saad include a provision to restart the iteration if the new residual norm is substantially larger than the current one. This amounts to discarding all iterant data except the current $(x^{(\ell)}, y^{(\ell)})$ pair, or rather the current $(x^{(\ell)}, r^{(\ell)})$ pair in their formulation. However, moderate increases in the residual norm are tolerated. Restarting was invoked when $\| r^{(\ell)} \|_2 < \eta \| r^{(\ell+1)} \|_2$, with $\eta \sim 0.1 - 0.3$. They report that the choice of η often played a key role in convergence, especially for more challenging problems (for which η was increased). However, in the results presented, there is no record of the number or location of instances of restarting, which makes it hard to interpret the results in terms of their nominal version of Anderson Mixing. For frequent restarting, nominal and restarted versions are quite different.

Brief remarks on Section 6 will suffice for our purposes. There are three pairs of examples. The first pair derive from the discretization by finite differences of a variant of a familiar nonlinear partial differential equation model problem on uniform grids on the unit square with homogeneous Dirichlet boundary conditions. This is treated simply as a root-finding problem, so the implicit fixed point problem was involved, with the number of iterations, L, needed to achieve a sufficiently small residual norm tabulated for a range of methods considered in Sections 2 and 3. As usual, we focus on their version of Anderson Mixing. The two examples involved $M = N = 400$ and $M = N = 10000$ and yielded $L = 65$ and $L = 273$, which therefore nominally involved rather large least squares problems with ample room for near linear dependence. Other methods involved comparable numbers of iterations, but Anderson Mixing was equal to the best of these. More crucially from my perspective, they found it necessary to take $\beta = 5 \times 10^{-5}$ for the smaller example and $\beta = 2 \times 10^{-5}$ for the larger example. I do not find results for such small values of β plausible and do not believe that much insight can be gained from these examples.

The second pair of examples involved RSDFT, a MATLAB code package implementing a density function theory approach to atomic electronic structure calculations. The two examples involved fixed point problems with $M = N = 157464$ and $M = N = 79507$, with $\beta = 1.0$ and $L \sim 40$ for the larger example and with $\beta = 0.5$ and $L \sim 25$ for the smaller example. Residual norms were plotted against the number of iterations for selected methods. In these examples, but not the other examples, results for "simple mixing" with $\beta = 0.5$ for the larger and $\beta = 0.3$ for the smaller were also plotted and converged smoothly but somewhat more slowly than most of the other methods. Anderson Mixing converged with some oscillations at a rate comparable to simple mixing : more slowly for the larger example and more rapidly for the smaller example.

The sign error hypothesis would lead us to expect that simple mixing would not converge. Experience of multiple authors would lead us to expect Anderson Mixing to converge significantly more rapidly than simple mixing. We face a quandary that may not be resolvable given the limited information available. Noting the use of different values of β for simple mixing and for the other methods, I wonder whether RSDFT provided a simple mixing option which was availed of for convenience; if so, presumably it would not contain the sign error, which is clearly contained in the simple mixing method as stated by Fang/Saad.

The third pair of examples involved PARSEC, a sophisticated Fortran 90 code package for electronic structure calculations in whose development over a decade the second author has participated. The two examples involved fixed point problems with $M = N = 118238$ and $M = N = 220490$, with $\beta = 0.1$ and $L = 23$ for the smaller example and with $\beta = 0.1$ and $L = 60$ for the larger example. Residual norms were plotted against the number of iterations for selected methods. All methods exhibited erratic oscillation, in some cases of large amplitude. Anderson Mixing was clearly the most effective method.

6.5 Walker/Ni

We shall now proceed to discussion of the second group of publications: Walker and Ni [16], Ni [12], and Toth and Kelley [15]. We shall also return briefly to Calef et al. [5]. The Walker and Ni [16] paper is seminal and influential; it introduced the Anderson Acceleration terminology and contained interesting theoretical results and impressive examples. As such it merits close attention. The original Ni [12] thesis has much less to recommend it. The Toth and Kelley [15] paper builds upon Walker and Ni [16], primarily theoretically. For reasons that will emerge, detailed discussion of some implementation issues related to Walker and Ni [16] will be postponed until the next section. There is now a burgeoning literature illustrating that Anderson Acceleration, and variants and extensions thereof, can be productively applied in a wide range of computational science and engineering contexts, but we shall not pursue these matters here.

Walker/Ni are clear that Anderson Acceleration is intended as a means of increasing the rate of convergence of the Picard iteration for the fixed point problem $g(x) = x$. They use the associated root-finding problem $g(x) - x = 0$ to introduce a residual, as a convenient abbreviation. In essence, they are using the stationary Extrapolation Algorithm, with $W = I$ and $\beta^{(\ell)} = 1$. Strictly speaking, the Walker/Ni Anderson Acceleration algorithm has a minor defect; it tacitly assumes that there will always be unique optimal affine combination coefficients, which need not be the case. While there will always be a unique vector in the affine subspace closest to zero, there may be nonunique affine combination coefficients characterizing that minimizing vector, which would result in the mathematical algorithm as stated not being well-defined. We know that a necessary and sufficient condition that there be unique affine combination coefficients is that the $r^{(\ell-k)}$, $0 \le k \le \min(\ell, M)$, be affine independent and that a sufficient, but not a necessary, condition is that $r^{(\ell-k)}$, $0 \le k \le \min(\ell, M)$, be linearly independent. This will impact theoretical considerations later. As an interim

measure, I shall adopt the understanding that should the mathematical algorithm become ill-defined at some stage, the process will terminate declaring failure.

This glitch will be resolved in the course of translating a conceptual mathematical algorithm into a numerically robust implementation thereof. Indeed, immediately after stating their Anderson Acceleration algorithm, Walker/Ni indicate that, in practice, they will monitor the condition number of the matrix $F^{(\ell)}$ with columns $r^{(\ell-k)}$, $0 \leq k \leq m^{(\ell)}$, and reduce $m^{(\ell)}$ accordingly, but then postpone further discussion. In the end, they actually consider a somewhat different approach. These matters will be sorted out subsequently.

A central contribution of the Ni [12] thesis, reworked and extended in Walker and Ni [16], is establishing a connection between Anderson Acceleration applied to the affine fixed point problem with $g(x) = Gx + h$ and GMRES (Generalized Minimal Residual Method) applied, with the same initial iterant, to the linear equation $(I - G)x = h$. Many classical iterative procedures for solving suitable linear equations $Ax = b$ correspond to the Picard iteration for an associated affine fixed point problem. If $I - G$ is nonsingular, g has a unique fixed point $\hat{x} = (I - G)^{-1}h$. If G is nonsingular, g is invertible: if $y = Gx + h = g(x)$, then $x = G^{-1}y - G^{-1}h = g^{-1}(y)$. We have $G(x) = G$. For any matrix norm induced by a vector norm (or simply compatible with some vector norm), a sufficient, but not a necessary, condition for convergence of the Picard iteration for g, for any $x^{(0)}$, is that $\| G \| < 1$ for the chosen matrix norm.

It is essential to the aforementioned connection between Anderson Acceleration and GMRES that we consider the quasistationary form of Anderson Acceleration by taking $M = N$. There are also counterparts in the GMRES literature to the stationary form of Anderson Acceleration.

We shall not delve into the details of the formulation, arguments, and results presented in Section 2 of Walker and Ni [16]. We shall simply sketch the nature of the aforementioned connection. In essence, what emerges is that the $\{\hat{u}^{(\ell)}\}$ sequence can be identified with the sequence of GMRES iterants. Because g is affine, we have $\hat{v}^{(\ell)} = g(\hat{u}^{(\ell)})$. Consequently, if the $\{\hat{u}^{(\ell)}\}$ sequence converges, the $\{\hat{v}^{(\ell)}\}$ sequence must also converge, both of them to $\hat{x}$. Because $\beta^{(\ell)} = 1$, we have $x^{(\ell+1)} = \hat{v}^{(\ell)} = g(\hat{u}^{(\ell)})$; thus, the Anderson Acceleration iterant $x^{(\ell+1)}$ and the GMRES iterant $\hat{u}^{(\ell)}$ are different, but related. If $\| G \| < 1$, we expect the $x^{(\ell+1)}$ iterant to have smaller error than the $\hat{u}^{(\ell)}$ iterant. Preconditioning, which plays a key role in the efficacy of GMRES, also enhances the convergence of the Picard iteration, thence Anderson Acceleration.

There is a known degeneracy possible in GMRES which has a counterpart in the quasistationary form of Anderson Acceleration. This corresponds to the situation in which the vector closest to zero from the affine span of $r^{(\ell-k)}$, $k = 1, 2, \ldots, \ell$, is the same as that from the affine span of $r^{(\ell-k)}$, $k = 0, 1, \ldots, \ell$, despite the increase in dimension if both sets are affine independent so the method is well-defined. This results in $\hat{u}^{(\ell)} = \hat{u}^{(\ell-1)}$, thence $\hat{v}^{(\ell)} = \hat{v}^{(\ell-1)}$, and consequently $x^{(\ell+1)} = x^{(\ell)}$, thence $y^{(\ell+1)} = y^{(\ell)}$. The upshot is that we will encounter affine dependence, so the Anderson Acceleration process will not be well-defined when seeking $x^{(\ell+2)}$, and by our interim assumption above will terminate declaring failure. GMRES may be

able to recover. This degenerate situation is mathematically possible, but extremely improbable for $\ell \ll N$. This observation adds to the already ample incentives to use a stationary or nonstationary form of Anderson Acceleration. The nonstationary version of the Extrapolation Algorithm detailed above should cope satisfactorily.

Section 3 of Walker and Ni [16] reviews the correspondence between Anderson Acceleration and the stationary simplified inverse multisecant method, following Fang and Saad [8] but without the sign error because Walker/Ni use $\beta^{(\ell)} = 1$ and $g(x) - x = 0$. It also reviews the stationary simplified direct multisecant method and the corresponding algorithm that is the counterpart of Anderson Acceleration. It then reprises Section 2 establishing the counterpart connections between applying the latter algorithm to the affine fixed point problem with $g(x) = Gx + h$ and applying the Arnoldi method rather than GMRES to the linear equation $(I - G)x = h$, with the same initial iterant.

Recall that during our examination of Calef et al. [5], discussion of one topic was postponed until conceptual aspects of Walker and Ni [16] had been addressed. Since Calef et al. establish the mathematical equivalence of nonlinear Krylov acceleration and Anderson Acceleration with $\beta^{(\ell)} = 1$, as studied by Walker/Ni, they can rephrase Walker/Ni results related to GMRES to apply to GMRES and nonlinear Krylov acceleration. Calef et al. also discuss the degenerate situation described above, and they introduce a systematic perturbation of their method to surmount this rare obstacle. Their approach to solving the least squares problem would fail, and presumably terminate, should the situation described above actually arise, because iterant data is processed strictly in order of increasing age, so $Ae_1 = 0$.

I regard undue attention to this unlikely and simple situation as misplaced. This is just one of many ways one can encounter affine dependence, or near affine dependence. Coping with the larger issue seems to me more to the point. I grant, however, that for theoretical purposes, one wants to be able to make general statements, as general as possible while still being correct.

We turn now to computational considerations related to Walker and Ni [16] and Ni [12]. Two preliminary points regarding references should be noted. For obvious reasons, Walker and Ni [16] and Ni [12] reference the third edition of Golub and Van Loan [9] published in 1996, whereas I am referencing the definitive fourth edition. It does not appear that this raises any serious issues in what follows. Ni [12] references Björck [3], but appears to have focussed only on one chapter, and to have failed to appreciate the relevance and significance of material in other chapters. Walker and Ni [16] does not reference Björck [3]. This has potential consequences for others seeking to build upon Walker and Ni [16].

The Ni [12] thesis introduces Anderson Acceleration as a root-finding method for $f(x) = 0$ by quoting the formulation of Anderson Mixing in Fang and Saad [8]. Recall that this is the Eyert [7] reformulation replacing the deviation basis ordered by increasing age by the difference basis ordered by decreasing age. The motivating context for the thesis is self-consistent field electronic structure calculations, so attention is focussed on the fixed point problem for $g(x)$, with $f(x) = g(x) - x$, or $g(x) = x + f(x)$. The choices $\beta^{(\ell)} = \beta > 0$, and eventually $\beta = 1$, are introduced, so the sign error in Fang and Saad [8] is avoided. In the end, the Walker and

Ni [16] paper adopts this formulation for computational purposes. The Ni [12] thesis also explores other options. Recall that Fang/Saad use an $AP = QR$ decomposition constructed employing Householder matrices and the standard pivoting strategy. Ni (explicitly) and Walker/Ni (apparently) use an $A = \hat{Q}\hat{R}$ factorization constructed without pivoting and employing the Gram–Schmidt process: presumably the modified Gram–Schmidt process.

We shall digress briefly before pursuing these matters further.

In formulating the Extrapolation Algorithm above, for $m \leq \min(\ell, M)$, we first introduced the iterant data $x^{(\ell-k)}$ and $y^{(\ell-k)} = g(x^{(\ell-k)})$, for $0 \leq k \leq m$; and, as a convenient abbreviation, the residuals $r^{(\ell-k)} = y^{(\ell-k)} - x^{(\ell-k)}$, for $0 \leq k \leq m$. We introduced the affine combinations $u^{(\ell)} = \sum_{k=0}^{m} \theta_k^{(\ell)} x^{(\ell-k)}$ and $v^{(\ell)} = \sum_{k=0}^{m} \theta_k^{(\ell)} y^{(\ell-k)}$, with $\sum_{k=0}^{m} \theta_k^{(\ell)} = 1$. We now take $W = I$ and seek optimal affine combination coefficients $\hat{\theta}_k^{(\ell)}$, with $\sum_{k=0}^{m} \hat{\theta}_k^{(\ell)} = 1$, minimizing $\| v^{(\ell)} - u^{(\ell)} \|_2 = \| \sum_{k=0}^{m} \theta_k^{(\ell)} (y^{(\ell-k)} - x^{(\ell-k)}) \|_2$, and the associated optimal $\hat{u}^{(\ell)} = \sum_{k=0}^{m} \hat{\theta}_k^{(\ell)} x^{(\ell-k)}$ and $\hat{v}^{(\ell)} = \sum_{k=0}^{m} \hat{\theta}_k^{(\ell)} y^{(\ell-k)}$. We can rephrase this as the task of minimizing $\| \sum_{k=0}^{m} \theta_k^{(\ell)} r^{(\ell-k)} \|_2^2$ subject to the equality constraint $\sum_{k=0}^{m} \theta_k^{(\ell)} = 1$.

Define the $N \times (m+1)$ matrix $F^{(\ell)}$ by $F^{(\ell)} e_k = r^{(\ell-k)}$, for $0 \leq k \leq m$, and the $(m+1)$-vector $\theta^{(\ell)}$ by $e_k^* \theta^{(\ell)} = \theta_k^{(\ell)}$, for $0 \leq k \leq m$, and similarly $\hat{\theta}^{(\ell)}$. Our task then is to minimize $\| F^{(\ell)} \theta^{(\ell)} \|_2^2$ subject to $e^* \theta^{(\ell)} = 1$. Introducing a Lagrange multiplier $\lambda^{(\ell)}$ and the Lagrangian

$$\phi(\theta^{(\ell)}, \lambda^{(\ell)}) = \frac{1}{2} \| F^{(\ell)} \theta^{(\ell)} \|_2^2 - \lambda^{(\ell)} (e^* \theta^{(\ell)} - 1),$$

we obtain the stationarity conditions

$$\frac{\partial \phi}{\partial \theta^{(\ell)}} = \left[(F^{(\ell)})^* F^{(\ell)} \right] \hat{\theta}^{(\ell)} - \hat{\lambda}^{(\ell)} e = 0$$

and

$$\frac{\partial \phi}{\partial \lambda^{(\ell)}} = -e^* \hat{\theta}^{(\ell)} + 1 = 0.$$

If we now assume that $F^{(\ell)}$ has maximal rank, so $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ is linearly independent, we know that $\left[(F^{(\ell)})^* F^{(\ell)} \right]$ is positive definite, thence nonsingular, and that $\left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1}$ is also positive definite. We obtain from the first stationarity condition

$$\hat{\theta}^{(\ell)} = \left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1} (\hat{\lambda}^{(\ell)} e),$$

and from the second stationarity condition

$$1 = e^* \hat{\theta}^{(\ell)} = \hat{\lambda}^{(\ell)} \left\{ e^* \left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1} e \right\},$$

thence

$$\hat{\lambda}^{(\ell)} = \left\{ e^* \left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1} e \right\}^{-1}$$

and

$$\hat{\theta}^{(\ell)} = \left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1} e / \left\{ e^* \left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1} e \right\}.$$

We see that if $F^{(\ell)}$ has maximal rank then there is a unique $\hat{\theta}^{(\ell)}$ given by the foregoing. Methods employed in various areas of physics and chemistry traditionally use this assumption and formulation to find $\hat{\theta}^{(\ell)}$. As we have discussed previously, there will always be a unique vector $\hat{v}^{(\ell)} - \hat{u}^{(\ell)}$ in the affine span of $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ which is closest to 0, in the sense of minimizing $\| v^{(\ell)} - u^{(\ell)} \|_2$, or making $v^{(\ell)}$ closest to $u^{(\ell)}$. A necessary and sufficient condition that there be a unique associated $\hat{\theta}^{(\ell)}$ is that $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ is affinely independent, and if $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ is affinely dependent, then there will be nonunique associated $\hat{\theta}^{(\ell)}$. Moreover, linear independence of $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$, or equivalently $F^{(\ell)}$ having maximal rank, is a sufficient, but not a necessary, condition for $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ to be affinely independent. Furthermore, linear dependence of $\left\{ r^{(\ell-k)} \right\}$, or equivalently $F^{(\ell)}$ being rank deficient, is a necessary, but not a sufficient, condition for 0 to lie in the affine span of $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$, so $\hat{v}^{(\ell)} = \hat{u}^{(\ell)}$. We anticipate that if the acceleration process is successful, then $\left\{ r^{(\ell-k)} \right\}_{k=0}^{m}$ will tend to become nearly linearly dependent, or equivalently $F^{(\ell)}$ will tend to become nearly rank deficient (ill-conditioned), as ℓ increases.

If we had a QR factorization $F^{(\ell)} = \hat{Q}\hat{R}$, with $\hat{Q}$ orthonormal, so $\hat{Q}^*\hat{Q} = I$, and $\hat{R}$ regularly upper triangular, then we could replace $\left[(F^{(\ell)})^* F^{(\ell)} \right]$ by $\hat{R}^*\hat{R}$, thence $\left[(F^{(\ell)})^* F^{(\ell)} \right]^{-1}$ by $\hat{R}^{-1}(\hat{R}^*)^{-1}$, thereby simplifying the calculation of $\hat{\theta}^{(\ell)}$. Recall also, that we showed above that $\kappa_2(F^{(\ell)}) = \kappa_2(\hat{R}) = \kappa_2(\hat{R}^*)$ and that $\kappa_2(\left[(F^{(\ell)})^* F^{(\ell)} \right]) = \kappa_2(F^{(\ell)})^2$. If the factorization is calculated using Householder matrices, $\hat{Q}$ will be nearly orthonormal, and these results are approximately valid. If the factorization is calculated using the modified Gram–Schmidt process, or other algorithms which may produce less nearly orthonormal $\hat{Q}$, then the quality of the approximations will deteriorate.

In defining $F^{(\ell)}$ above, I have followed my usual custom of ordering the residual columns by increasing age. Likewise, when defining $\triangle F^{(\ell)}$ in the inverse multisecant method context, I ordered the deviation or difference basis vector columns by increasing age. In both cases, I regard this choice to be the computationally appropriate one, for my purposes. However, I noted that the usual custom when using the difference basis in this multisecant context is to order the columns by decreasing age, as in Fang and Saad [8] thence Walker and Ni [16]. With this revised understanding, there is no harm in my continuing to use the $\triangle F^{(\ell)}$ notation hereafter, though the reordering requires reparameterization and has computational consequences. Likewise, the counterpart to $F^{(\ell)}$ in Ni [12], there called D , and in Walker and Ni [16], there called F_k, order the residual columns by decreasing age. Again, with this revised understanding, there is likewise no harm in my continuing to use the $F^{(\ell)}$ notation hereafter.

Ni considers the stationarity conditions derived above and the simplified version thereof under the assumption that $F^{(\ell)}$ has maximal rank. In this context, the $F^{(\ell)} = \hat{Q}\hat{R}$ factorization plays a role. Ni also considers the Fang/Saad formulation,

and in this context, the $\triangle F^{(\ell)} = \hat{Q}\hat{R}$ factorization plays a role. Finally, Ni devised an original method based on ideas in Björck [3], involving a different $\hat{Q}\hat{R}$ factorization, but we shall not go into detail. The four were compared based on the average of the condition numbers of the linear equations to be solved for a suite of test cases. The Walker/Ni choice of the Fang/Saad formulation was based on this comparison, though the Ni method was not far behind. As one might expect, the methods using $F^{(\ell)}$ were not competitive by this criterion; they are comparable to using the normal equations to solve the least squares problem associated with $\triangle F^{(\ell)}$. It appears that the discussion of $F^{(\ell)}$ in Walker and Ni [16] is an artifact from the early parts of the Ni [12] thesis and can safely be ignored.

The important point for further consideration here is the task of updating the $\hat{Q}\hat{R}$ factorization when moving from $\triangle F^{(\ell-1)}$ to $\triangle F^{(\ell)}$, and then arranging that $\triangle F^{(\ell)}$ have an acceptably small condition number to be regarded as numerically of maximal rank. Similar considerations are involved in moving from $F^{(\ell-1)}$ to $F^{(\ell)}$, and Ni introduces them in this context. There are four sets of issues to be dealt with. We focus on $\triangle F^{(\ell-1)}$ and $\triangle F^{(\ell)}$.

The first, and simplest, issue is updating the $\hat{Q}\hat{R}$ factorization of $\triangle F^{(\ell-1)}$ to that for $\triangle F^{(\ell)}$ when a new last column is adjoined. When using the (modified) Gram–Schmidt process, this is straightforward as the next step of the column-oriented version of the process. It would be equally easy to update the QR decomposition by applying the next Householder matrix for this purpose. Note that it would be even easier to obtain the $\hat{Q}\hat{R}$ factorization of $\triangle F^{(\ell-1)}$ from that of $\triangle F^{(\ell)}$: simply delete the last columns of $\hat{Q}$ and $\hat{R}$, the new ones created when moving from $\triangle F^{(\ell-1)}$ to $\triangle F^{(\ell)}$. For the QR decomposition an analogous approach suffices.

The second issue arises when $\ell > M$. In order to keep $m \leq M$, we must delete the first column of $\triangle F^{(\ell-1)}$, update the $\hat{Q}\hat{R}$ factorization of this intermediate matrix, and then adjoin a new last column, updating the factorization to that of $\triangle F^{(\ell)}$. Both Ni [12] and Walker and Ni [16] refer the reader to Golub and Van Loan [9]—actually the 1996 edition—for details on how to update the $\hat{Q}\hat{R}$ factorization of the intermediate matrix. I believe that the reader ought to have been referred to Björck [3] instead and that this is significant.

Golub and Van Loan [9] is a magnificent classic, but it does have an agenda. They tend to focus on the QR decomposition rather than the $\hat{Q}\hat{R}$ factorization, and on the use of Householder and Givens matrices. Indeed, they explain how to update a QR decomposition of the intermediate matrix, obtained using Householder matrices, by astute use of Givens matrices, with the resulting Q encoded as the original product of Householder matrices and the new product of Givens matrices. This approach is ideal for a situation in which one has solved a nominal least squares problem and wishes to explore the effects of changes therein, for sensitivity or exploratory data analysis, returning to the nominal problem before making additional changes. In our present context, we contemplate making potentially long chains of successive changes, which is much more manageable when working the $\hat{Q}\hat{R}$ factorization using the modified Gram–Schmidt process. Björck carefully explains both approaches, and there are conceptual and practical differences.

The third issue arises, for $\ell > 1$, when we need to choose $m < \min(\ell, M)$ in order to work with a $\triangle F^{(\ell)}$ of small enough condition number. The immediate issue is

how the condition number is to be approximated. Ni simply talks about the condition number of $F^{(\ell)}$, and later $\triangle F^{(\ell)}$, but there is a hint that MATLAB facilities were availed of. Walker/Ni note that we have an approximate $\hat{Q}\hat{R}$ factorization for each of the matrices whose condition number is sought, so one can focus on $\kappa_2(\hat{R})$, but no further indication is given as to how the latter is to be estimated. Setting this matter aside, the general idea is to remove the first columns of intermediate matrices with too large a condition number, one-after-another using the updating discussed above, until an acceptable $\triangle F^{(\ell)}$ is found. This must result in $m \geq 1$, since the condition number would be 1 for $m = 1$.

The fourth issue concerns the determination of the least squares solution for the $\triangle F^{(\ell)}$ case, and the solution of the simplified stationarity conditions for the $F^{(\ell)}$ case. Both Ni [12] and Walker and Ni [16] indicate unawareness of relevant material in Björck [3] in this connection, which is simply mentioned in passing in Golub and Van Loan [9].

Because Walker and Ni [16] is an important paper, I shall devote the next section to explaining relevant material from Björck [3], in order to be able to comment thereon in more detail in the context at hand.

The computational examples in Walker and Ni [16] speak for themselves. They are many, varied, and impressive. The examples in Ni [12] are different.

6.6 Toth/Kelley

Since Toth and Kelley [15] is primarily theoretical, our discussion will be brief and informal and focused on what can be learned about computational issues therefrom. They are concerned, for the most part, with the stationary Anderson Acceleration algorithm as formulated in Walker and Ni [16]. The case $M = 1$ receives special attention. The underlying assumption basically is that the Picard iteration for g converges to the fixed point $\hat{x}$, for any initial iterant $x^{(0)}$ if g is affine and for $x^{(0)}$ close enough to $\hat{x}$ if g is nonlinear. The detailed hypotheses, arguments, and conclusions are intricate; we shall not go into them here and refer the interested reader to the paper itself. For nonlinear g, achieving such results is a significant accomplishment. On the other hand, in broad brush terms, the outcome is that the convergence of the accelerated iteration is no worse than that of the Picard iteration. It would be more surprising if this were not true than that it is true. This in no way detracts from the merits of the paper per se; the available mathematical tools simply have limited purchase. It does mean that these results do not in themselves form a basis for assessing or improving the efficacy of Anderson Acceleration, but they do focus attention on the role of the convergence of the Picard iteration.

Examples in Walker and Ni [16] and other contexts commonly demonstrate that the convergence of the accelerated iteration is much more rapid than that of the Picard iteration. Moreover, the acceleration may initiate convergence (or at least find an approximate fixed point) for initial iterants for which the Picard iteration does not converge. While the analysis follows Walker and Ni [16] in taking $\beta^{(\ell)} = 1$ throughout, we observe (on the basis of previous discussion) that it applies equally well for $\beta^{(\ell)} = \beta \neq 1$, potentially to advantage in the rate of convergence of the corresponding Picard iteration. Finally, some flexibility noted in formulating hypotheses offers

the prospect that the results obtained and overall performance may be insensitive to modest variations in the acceleration process.

By the triangle inequality, we have

$$1 = \left| \sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} \right| \leq \sum_{k=0}^{\min(\ell,M)} \left| \theta_k^{(\ell)} \right|.$$

The Toth/Kelley theory involves an upper bound, uniform in ℓ, on $\sum_{k=0}^{\min(\ell,M)} \mid \theta_k^{(\ell)} \mid$. The stationary Extrapolation Algorithm cannot guarantee even the existence of such a bound. Toth/Kelley suggest three approaches to imposing a reasonable such bound. The nonstationary Extrapolation Algorithm could combine this constraint with the $\theta_0^{(\ell)} > 0$ constraint, by virtue of its use of the deviation basis. Iterant data is disregarded by setting corresponding affine combination coefficient(s) to zero, and minimizing with respect to the others, subject to the $\sum_{k=0}^{\min(\ell,M)} \theta_k^{(\ell)} = 1$ constraint. Regularization can also be accommodated.

In the main computational example presented, Toth/Kelley explore the possibility of replacing $\| \cdot \|_2$ in the minimization problem determining the optimal affine combination coefficients by $\| \cdot \|_1$ or $\| \cdot \|_\infty$. Doing so replaces solution of a least squares problem by solution of a special linear programming problem, which is computationally more expensive (though good algorithms exist). It would also complicate moving from a stationary to a nonstationary version of the method. Moreover, it opens up new possibilities for nonuniqueness and failure of the algorithm to be well defined which are not posed by the strictly convex $\| \cdot \|_2$. Because $\| \cdot \|_1$ and $\| \cdot \|_\infty$ are not strictly convex, they yield a point in the affine subspace closest to zero, but not necessarily a unique one: a convex set thereof. In the particular example presented, the performance of all three norms is comparable, and severe ill-conditioning appears to curtail the effectiveness of modestly increasing M.

A final remark, tangentially related to Toth and Kelley [15], is occasioned by the first paragraph of the introduction thereto and the extensive references cited therein. There are a number of root-finding algorithms in the literature which are related to the Extrapolation Algorithm in similar fashion to that laid out above in the multisecant context. When suitably applied to $g(x) - x = 0$, they generate iterants corresponding to some version of the Extrapolation Algorithm applied to the fixed point problem for $g(x)$. Consequently, when applied to the root-finding problem $f(x) = 0$, they are generating iterants corresponding to the application of that version of the Extrapolation Algorithm to the implicit fixed point problem for $x + f(x)$. Attention to the convergence of the Picard iteration for this implicit fixed point problem is a relevant consideration, and the Toth/Kelley results may be applicable.

7 Implementation: Walker/Ni

We shall consider several implementation issues arising when the difference basis is used as in the Walker/Ni paper. Nominally, we have $m^{(\ell)} = \min(\ell, M)$. The special case $M = 1$ is best dealt with in its own terms; the least squares problem involved can

be solved directly. Therefore, we assume hereafter that $M > 1$. The case $m^{(1)} = 1$ is also special, but can be incorporated into the general formulation for later purposes.

The least squares problem $A^{(\ell)}c^{(\ell)} = b^{(\ell)}$ posed has $b^{(\ell)} = r^{(\ell)}$ and orders $A^{(\ell)}e_i$ and $e_i^* c^{(\ell)}$ by decreasing age, so $e_i^* c^{(\ell)} = \xi_j^{(\ell)}$ and $A^{(\ell)}e_i = r^{(\ell-j+1)} - r^{(\ell-j)}$, with $j = m^{(\ell)} + 1 - i$, for $i = 1, 2, \ldots, m^{(\ell)}$. This ordering has the cost advantage that the columns of $A^{(\ell)}$ and its $A^{(\ell)} = Q^{(\ell)}R^{(\ell)}$ factorization can be computed incrementally. (No decompositions are involved so we can simplify the notation for factorizations in this section.) This approach has the disadvantages that the problem may be poorly scaled and may privilege older rather than younger iterants; moreover, pivoting is not an option.

We shall look first at the stationary case where $\beta^{(\ell)} = \beta > 0$ and $m^{(\ell)}$ is monotone nondecreasing: $0 < m^{(\ell)} = \min(\ell, M)$. (The Walker/Ni paper takes $\beta = 1$.) We shall consider the quasistationary phase, $0 < m^{(\ell)} = \ell \leq M$, and then the equistationary phase, $0 < m^{(\ell)} = M < \ell$. This will lead us to two canonical constructions through which the computation can be implemented. We shall look second at the nonstationary case where $m^{(\ell)}$ is permitted to decrease, building on the previous discussion.

In the quasistationary phase, for $\ell > 1$, $A^{(\ell)}$ is $A^{(\ell-1)}$ with

$$(r^{(\ell)} - r^{(\ell-1)}) = (y^{(\ell)} + x^{(\ell-1)}) - (x^{(\ell)} + y^{(\ell-1)})$$

adjoined as a new last column. For $\ell = 1$, take $Q^{(1)}e_1 = A^{(1)}e_1 / \parallel A^{(1)}e_1 \parallel_2$ and $e_1^* R^{(1)} e_1 = \parallel A^{(1)}e_1 \parallel_2$, so $A^{(1)} = Q^{(1)}R^{(1)}$. For $\ell > 1$, our basic task is to update the $A^{(\ell-1)} = Q^{(\ell-1)}R^{(\ell-1)}$ factorization to $A^{(\ell)} = Q^{(\ell)}R^{(\ell)}$, by constructing new last columns for $Q^{(\ell)}$ and $R^{(\ell)}$. We shall use (see below) the column-oriented form of the modified Gram–Schmidt process, with or without reorthogonalization, following Björck [3]. This first construction will also be used to solve the least squares problem $A^{(\ell)}c^{(\ell)} = b^{(\ell)}$ given the factorization $A^{(\ell)} = Q^{(\ell)}R^{(\ell)}$.

In the equistationary phase, our first task is to obtain $\check{A}^{(\ell-1)}$ by deleting the first column of $A^{(\ell-1)}$ and updating the $A^{(\ell-1)} = Q^{(\ell-1)}R^{(\ell-1)}$ factorization to $\check{A}^{(\ell-1)} = \check{Q}^{(\ell-1)}\check{R}^{(\ell-1)}$. We shall use the approach outlined by Björck for this second construction, which is also used in the nonstationary case. Our second task is to obtain $A^{(\ell)}$ by adjoining $(r^{(\ell)} - r^{(\ell-1)})$ to $\check{A}^{(\ell-1)}$ as a new last column and updating the $\check{A}^{(\ell-1)} = \check{Q}^{(\ell-1)}\check{R}^{(\ell-1)}$ factorization to $A^{(\ell)} = Q^{(\ell)}R^{(\ell)}$ using the first construction. These two tasks should be done separately and in this order.

7.1 First construction

We shall use generic notation in describing the first construction. Suppose that we have the factorization $A = QR$, with $A, Q \in \mathbb{R}^{n\times m}$ and $R \in \mathbb{R}^{m\times m}$, where A has maximal rank, Q is orthonormal and R is upper triangular and nonsingular. We seek the corresponding factorization

$$\begin{bmatrix} A & a \end{bmatrix} = \begin{bmatrix} Q & q \end{bmatrix} \begin{bmatrix} R & r \\ 0 & \rho \end{bmatrix},$$

with $a, q \in \mathbb{R}^n, r \in \mathbb{R}^m, m < n$, and $\rho \in \mathbb{R}$. Set $d^{(0)} = a$. For $k = 1, 2, \ldots, m$, calculate $\phi^{(k)} = (Qe_k)^* d^{(k-1)}$, $d^{(k)} = d^{(k-1)} - \phi^{(k)}(Qe_k)$, and $e_k^* r = \phi^{(k)}$. Set $\rho = \{d^{(m)*} d^{(m)}\}^{1/2}$ and $q = d^{(m)}/\rho$. We obtain $[A \;\; a] = [QR \;\; Qr + \rho q]$.

In particular, for $a = b$ we see that

$$Ac - b = [A \;\; b] \begin{bmatrix} c \\ -1 \end{bmatrix} = QRc - (Qr + \rho q) = Q(Rc - r) - \rho q.$$

By construction, we have $q \perp Qe_k, 1 \le k \le m$, thence $q \perp Q(Rc - r)$. By the Pythagorean Law, we find that

$$\| Ac - b \|_2^2 = \| Q(Rc - r) \|_2^2 + \rho^2.$$

We then identify $R^{-1}r$ and ρ as the minimizer of $\| Ac - b \|_2$ and the minimum value thereof. Ideally, Q is an orthonormal matrix. Observe, however, that the foregoing depends on the normalization $\| Qe_k \|_2 = 1$ but not specifically on the orthogonality $Qe_i \perp Qe_j, i \ne j$. Without accurate normalization, we could replace $\phi^{(k)} = (Qe_k)^* d^{(k-1)}$ in the foregoing by $\phi^{(k)} = (Qe_k)^* d^{(k-1)}/(Qe_k)^*(Qe_k)$.

The algorithm can be enhanced by invoking reorthogonalization to strengthen the $q \perp Qe_k, 1 \le k \le m$, condition. It may suffice to omit reorthogonalization in computing Q since the $Qe_i \perp Qe_j, i \ne j$, condition is less crucial. There are more elaborate reorthogonalization schemes in the literature, but the following simple modification of the foregoing will illustrate the point. Set $d^{(0)} = a$. For $k = 1, 2, \ldots, m$, calculate $\phi_0^{(k)} = (Qe_k)^* d^{(k-1)}$, $d^{(k-1/2)} = d^{(k-1)} - \phi_0^{(k)}(Qe_k)$, $\phi_1^{(k)} = (Qe_k)^* d^{(k-1/2)}$, $d^{(k)} = d^{(k-1/2)} - \phi_1^{(k)}(Qe_k)$, and $e_k^* r = \phi_0^{(k)} + \phi_1^{(k)}$. Set $\rho = \{d^{(m)*} d^{(m)}\}^{1/2}$ and $q = d^{(m)}/\rho$. The costs involved essentially double. Without reorthogonalization, we need two invocations of the first construction at each stage. We essentially need the equivalent of three if reorthogonalization is used only for q, and four if it is used for both Q and q. Once $x^{(\ell+1)}$ is chosen, q, r and ρ can be deleted to proceed to the next iteration.

We summarize some known (see Björck) properties of this modified Gram–Schmidt process for seeking the least squares solution $\hat{c}$ of $Ac = b$ by constructing the factorization

$$\begin{bmatrix} A & b \end{bmatrix} = \begin{bmatrix} Q & q \end{bmatrix} \begin{bmatrix} R & r \\ 0 & \rho \end{bmatrix}.$$

It can be shown that

(i) $\| A - QR \|_2 \sim \mathbf{u} \| A \|_2$,
(ii) $\| I - Q^* Q \|_2 \sim \mathbf{u} \kappa_2(A)$,
(iii) $\| diag(I - Q^* Q) \|_2 \sim \mathbf{u} \| diag(I) \|_2$,
(iv) $\| \hat{c} - R^{-1} Q^* b \|_2 \sim \mathbf{u} \kappa_2(A)^2$,

where $\mathbf{u}$ is the unit roundoff error and $\kappa_2(A)$ is the condition number of A. The first property indicates that the range of Q is a good approximation to that of A, so approximately orthogonalizing q to the range of Q does likewise to that of A. This implies that ρq approximates $b - A\hat{c}$ well. It can also be shown that there is an orthonormal $\tilde{Q}$ such that $\| A - \tilde{Q}R \|_2 = \mathbf{u} \| A \|_2$, so we anticipate that $\kappa_2(R) \approx \kappa_2(A)$. The second and third properties indicate that Q is not ideally orthogonal, but is close to ideally normalized. The $Qe_k, 1 \le k \le m$, should still provide a well-conditioned

basis for the range of Q, thence approximately for that of A, unless $\kappa_2(A)$ is large. The fourth property indicates that ignoring the fact that Q is not ideally orthonormal may well result in an approximation $R^{-1}Q^*b$ to $\hat{c}$ no better than that generated by solving the normal equations $A^*A\hat{c} = A^*b$ using the Cholesky factorization of A^*A. However, $R^{-1}r$ is a satisfactory approximation to $\hat{c}$; in fact, at least as satisfactory as that generated using Householder (or Givens) matrix triangularization. It appears from the Ni thesis that this point was not appreciated; however, the condition numbers tabulated for the test cases studied are moderately small. (Golub and Van Loan [9] mentions (i) and (ii) on pages 255–256 and (iv) on page 265; (iii) is my addition.)

7.2 Second construction

We turn now to the second construction. We are given the $A = QR$ factorization. $\check{A}$ is obtained by deleting the first column of A, and we seek the $\check{A} = \check{Q}\check{R}$ factorization. Let $P = P^*_{1:m}$ be the permutation matrix effecting a left circular shift so that the first column of A becomes the last column of AP; thus, $\check{A}$ is AP with its last column deleted. We have $AP = QRP$ and recognize that RP is upper Hessenberg: $e_i^*RPe_j = 0, i > j+1$. Let G be a product of Givens matrices, thence unitary, such that G^*RP is upper triangular. The formation of G is a standard topic in the numerical linear algebra literature such as Björck [3] or Golub and Van Loan [9]. I shall record my favorite version since it accommodates both the real and complex cases gracefully and will yield the standard QR factorization in this context, as does the modified Gram–Schmidt process: the diagonal elements of R are real and positive for maximal rank matrices. To avoid a lengthy digression at this point, I shall postpone this discussion until the end of this section. Issues related to the formation of QG are more salient here. We now have $AP = (QG)(G^*RP)$. We thereby obtain $\check{Q}$ by deleting the last column of QG and $\check{R}$ by deleting the last row and column of G^*RP. The last columns to be deleted need not be computed; however, the last column of G^*RP might be worth examining in the light of the use of the first construction to solve the least squares problem.

If Q is orthonormal, then so is QG, thence also $\check{Q}$. However, if Q is not ideally orthogonal, we must expect that QG and thence $\check{Q}$ will be even less so. For $M = 2$, the orthogonality of QG should be comparable to that of Q, because Givens matrices are unitary. In principle, for $M > 2$, if the two-dimensional subspace of the range of Q spanned by the two columns thereof on which a particular Givens matrix acts is exactly orthogonal to the rest of the columns, this will continue to be the case; otherwise, the cosines of the angles between the two new columns and some or all of the other columns will tend to be larger in magnitude than for the old columns. In practice, Q is not ideally orthogonal, so increasing entropy presages deteriorating orthogonality. Normalization should be better preserved, because Givens matrices are unitary, but will erode for large N.

Even though the orthogonality can be expected to deteriorate, the foregoing discussion indicates that this may not pose a serious obstacle. If Q is nearly ideally normalized, the normalization will deteriorate slowly; however, this is potentially more serious. One may wish to renormalize by dividing each column of the putative $\check{Q}$ by its norm, and multiplying the corresponding row of $\check{R}$ by that norm. The

alternative would be to accept and deal with less than ideal normalization as noted previously, which also requires evaluation of these norms. The deterioration will increase with N and M. We still expect the range of $\breve{Q}$ to be a reasonably good approximation to the range of $\breve{A}$, but this connection will also deteriorate, especially for larger numbers of columns of $\breve{Q}$ and $\breve{A}$. Inclusion of the newest iterant data and deletion of the oldest iterant data limits deterioration accumulation. This encourages moderately small M.

The tools needed for the stationary case are now in hand. Thorny issues arise if we turn to the nonstationary case. After the youngest difference basis vector has been adjoined as a new last column and the factorization has been updated, the Walker/Ni paper (and Ni thesis) suggests the possibility of deleting one or more of the older ones in order to control the condition number. Without scaling and pivoting, detecting near (or actual) linear dependence of the columns is more problematic. It would be advantageous, in my opinion, to incorporate the standard scaling strategy by dividing each prospective new last column by its norm before adjoining it, so $\| A^{(\ell)} e_k \|_2 = 1$ for $1 \leq k \leq m^{(\ell)}$. The threshold determining declaration of near (or actual) linear dependence should reflect tolerance for ill-determination rather than data uncertainty.

The Walker/Ni proposal is to estimate the condition number of $R^{(\ell)}$ as a proxy for that of $A^{(\ell)}$. If the estimate exceeds a specified threshold, the second construction would be used to remove the first column of $A^{(\ell)}$ and update the factorization, thereby reducing $m^{(\ell)}$. The estimation and removal cycle would be repeated until the threshold is no longer exceeded. By (ii), the orthogonality of the columns of $\hat{Q}$ may already have been damaged by a large condition number, and this is not repaired, but rather exacerbated, as $m^{(\ell)}$ is reduced to control the condition number. Without scaling and pivoting, the simple condition number estimate provided by the diagonal elements of $R^{(\ell)}$ may be too unreliable to be useful. The more robust estimates developed above might be considered. The Walker/Ni paper (and the Ni thesis) appears to require resort to a condition number estimation code. This also ignores the fact that less than ideal and deteriorating orthonormality of $Q^{(\ell)}$ weakens the connection between the condition numbers of $A^{(\ell)}$ and $R^{(\ell)}$; however, this may be adequate for the diagnostic purposes intended. The process could become expensive for large M and especially for large N, though perhaps not prohibitively so for expensive g evaluations. What was actually done remains unclear to me; the reference cited does not address the issue. I note their remark that they used a relatively large M and usually converged rapidly enough to keep $\ell \leq M$, for $M \ll N$. Thus, they remained in the quasistationary phase.

In short, there is a tradeoff to be made between efficacy and efficiency in incremental processing of the iterant data and in adjusting $m^{(\ell)}$ to cope with the impact of anticipated ill-conditioning on the determination of $x^{(\ell+1)}$.

7.3 Givens matrices

Finally, we return to the choice of Givens matrices used to form G. There are two distinct classes of Givens matrices: rotators and reflectors. Either class could be used. I prefer the reflectors, but the rotators are more commonly encountered in the literature. Consequently, I shall discuss both. The essence of the matter is contained

in the prototype 2×2 matrix case. I shall formulate the complex version and point out the minor simplifications in the real version. I shall then consider general Givens matrices whose product would be used to form G and to calculate G^*RP and QG.

The Givens rotator and reflector prototypes are $\begin{bmatrix} c & \bar{s} \\ -s & \bar{c} \end{bmatrix}$ and $\begin{bmatrix} c & \bar{s} \\ s & -\bar{c} \end{bmatrix}$, respectively, with $| \ c \ |^2 \ + \ | \ s \ |^2 = 1$. Givens matrices are unitary; the rotators have determinant 1, and the reflectors have determinant -1, this being their distinguishing characteristic. These properties are easily verified for the prototypes, and by extension later for their general counterparts.

For $d = \sqrt{| \ a \ |^2 + | \ b \ |^2} > 0$, we seek c and s such that $| \ c \ |^2 + | \ s \ |^2 = 1$ and

$$\begin{bmatrix} c & \bar{s} \\ -s & \bar{c} \end{bmatrix}^* \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \bar{c}a - \bar{s}b \\ sa + cb \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}.$$

We see that $c = a/d$ and $s = -b/d$. We likewise seek c and s such that

$$\begin{bmatrix} c & \bar{s} \\ s & -\bar{c} \end{bmatrix}^* \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \bar{c}a + \bar{s}b \\ sa - cb \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}.$$

We see that $c = a/d$ and $s = b/d$. The key point is that we can determine d, c, and s from a and b so as to transform $\begin{bmatrix} a \\ b \end{bmatrix}$ into the target $\begin{bmatrix} d \\ 0 \end{bmatrix}$. There are familiar protocols for evaluating norms such as d. For the complex case, division by d is facilitated by the fact that it is real. From this perspective, it may be advantageous to arrange that c or s be real. For a unimodular ϕ, $| \ \phi \ | = 1$, multiplying c and s, as defined above, by $\bar{\phi}$ will yield the new target $\begin{bmatrix} \phi d \\ 0 \end{bmatrix}$, the most general target for a 2×2 unitary matrix generating the requisite zero. Choosing $\phi = sgn(a)$ will make the new c real, and choosing $\phi = sgn(b)$ will make the new s real.

In the problem at hand, the modified Gram–Schmidt process generates the standard QR factorization with the diagonal elements of R real and positive, for maximal rank matrices. The original choice for c and s above preserves this property. This implies that b will always be real and positive, so s will automatically be real : $sgn(b) = 1$. The customary choice in the literature is to make c real. Observe that the reflector is Hermitian if c is real.

For a pair of vectors f and g, we find that

$$\begin{bmatrix} f & g \end{bmatrix} \begin{bmatrix} c & \bar{s} \\ -s & \bar{c} \end{bmatrix} = \begin{bmatrix} cf - sg & \bar{s}f + \bar{c}g \end{bmatrix}$$

and

$$\begin{bmatrix} f & g \end{bmatrix} \begin{bmatrix} c & \bar{s} \\ s & -\bar{c} \end{bmatrix} = \begin{bmatrix} cf + sg & \bar{s}f - \bar{c}g \end{bmatrix}.$$

It follows that we also have

$$\begin{bmatrix} c & \bar{s} \\ -s & \bar{c} \end{bmatrix}^* \begin{bmatrix} f^* \\ g^* \end{bmatrix} = \begin{bmatrix} \bar{c}f^* - \bar{s}g^* \\ sf^* + cg^* \end{bmatrix}$$

and

$$\begin{bmatrix} c & \bar{s} \\ s & -\bar{c} \end{bmatrix}^* \begin{bmatrix} f^* \\ g^* \end{bmatrix} = \begin{bmatrix} \bar{c} f^* + \bar{s} g^* \\ s f^* - c g^* \end{bmatrix}.$$

More generally, for $j < k$, the Givens rotator takes the form

$$\begin{aligned} G_{jk}(a,b) \;=\; & I - (e_j e_j^* + e_k e_k^*) \\ & + (c e_j - s e_k) e_j^* \\ & + (\bar{s} e_j + \bar{c} e_k) e_k^*, \end{aligned}$$

and the Givens reflector takes the form

$$\begin{aligned} G_{jk}(a,b) \;=\; & I - (e_j e_j^* + e_k e_k^*) \\ & + (c e_j + s e_k) e_j^* \\ & + (\bar{s} e_j - \bar{c} e_k) e_k^*. \end{aligned}$$

Clearly, postmultiplication of a matrix by $G_{jk}(a,b)$ alters only the jth and kth columns, and premultiplication of a matrix by $G_{jk}^*(a,b)$ alters only the jth and kth rows. The prototype results above therefore suffice to specify the requisite calculations. There is no need to form $G_{jk}(a,b)$ or $G_{jk}^*(a,b)$. It is convenient to parameterize $G_{jk}(a,b)$ by a and b, with the understanding that d, c, and s are defined in terms of these as previously indicated.

For the real case, we need only make superficial changes: elide all vacuous conjugations in the foregoing, both those explicit in $\bar{c}$ and $\bar{s}$ and those implicit in the asterisk superscript denoting conjugate transposition, which becomes simply transposition. If so desired, $|\ a\ |^2$, $|\ b\ |^2$, $|\ c\ |^2$, and $|\ s\ |^2$ could be replaced by a^2, b^2, c^2 and s^2, to advantage.

We can now sketch the formation of G^*RP and QG. The first step is to identify a and b with the diagonal and subdiagonal elements of the first column of the upper Hessenberg matrix RP and calculate the corresponding d, c, and s. Then process the first and second rows of RP and the first and second columns of Q using the foregoing, thereby making the corresponding diagonal element d and subdiagonal element 0. The second step is to repeat this pattern starting with the resulting diagonal and subdiagonal elements in the second column and processing the resulting second and third rows and columns. Continue in this pattern until the upper triangular matrix G^*RP and QG have been formed. G is the product of the Givens rotators or reflectors used, in the order of their formation, though this product need not be formed. There is also no need to form AP; forming RP might be helpful, though avoidable.

Appendix

We shall discuss here supplementary matters not strictly required within the main text, but related and potentially relevant thereto. We adopt the notation and terminology previously introduced. We rely upon essential results argued above. We provide arguments for nonessential results previously stated without proof and choices tacitly made without explicit justification.

Affine subspaces and affine independence/dependence are essentially geometric concepts, though it is convenient to describe and manipulate them in algebraic terms. The labeling of $x^{(\ell-k)}$, $y^{(\ell-k)} = g(x^{(\ell-k)})$ and $r^{(\ell-k)} = y^{(\ell-k)} - x^{(\ell-k)}$, $0 \le k \le m$, derives from the iterative process context of interest, the ordering reflecting the "age" of the iterants. The affine span of the affine independent defining set $\{r^{(\ell-k)}\}_{k=0}^{m}$ is an affine subspace of maximal dimension, m. We have chosen above to describe this algebraically using the shift vector $r^{(\ell)}$ and the linear subspace with the deviation basis $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$. From a geometric perspective, the labeling and ordering of the defining set is irrelevant; any member and the associated deviation basis could equally well have been used. Moreover, any nonzero member of the affine span and the associated deviation basis could be used. It is the questions of how the latter might be accomplished algebraically, and to what advantage, that originally motivated inclusion of this appendix. For the moment, we shall continue with our previous choice of shift vector $r^{(\ell)}$ and deviation basis $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$, examining this choice and alternatives later.

The affine combination $\sum_{k=0}^{m}\theta_k^{(\ell)} r^{(\ell-k)}$, with $\sum_{k=0}^{m}\theta_k^{(\ell)} = 1$, can be written in the form $r^{(\ell)} + \sum_{k=1}^{m}\theta_k^{(\ell)}(r^{(\ell-k)} - r^{(\ell)})$, with $\theta_0^{(\ell)} = 1 - \sum_{k=1}^{m}\theta_k^{(\ell)}$. If we use shift vector $r^{(\ell)}$ in representing the affine span of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, we identify $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ as a spanning set for the associated linear subspace. The dimension is maximal, m, if $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ is linearly independent, thence a basis for the linear subspace. Thus, $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely independent if $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ is linearly independent, and $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely dependent if $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ is linearly dependent.

Recall the pair of assertions above that linear independence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is a sufficient, but not a necessary, condition for affine independence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, and that linear dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is a necessary, but not a sufficient, condition for 0 to be a member of the affine span of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$. We now provide the requisite proofs, in reverse order.

If 0 is an affine combination of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, there is a nontrivial linear combination of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ which is 0, so $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is linearly dependent. Therefore, linear dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is a necessary condition for 0 to be a member of the affine span of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$. If $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is linearly dependent, but all nontrivial linear combinations of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ which are 0 have the property that the sum of their linear combination coefficients is zero, then there is no affine combination of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ that is 0. Therefore, linear dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is not a sufficient condition for 0 to be a member of the affine span of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$.

Assume that $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely dependent, so $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ is linearly dependent and there is a nontrival linear combination of $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ which is 0. If the sum of the linear combination coefficients is nonzero, then we can express $r^{(\ell)}$ as a linear combination of $\left\{r^{(\ell-k)}\right\}_{k=1}^{m}$, so $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is linearly dependent. If the sum of the linear combination coefficients is zero, then the same nontrivial linear combination of $\left\{r^{(\ell-k)}\right\}_{k=1}^{m}$ is zero, so $\left\{r^{(\ell-k)}\right\}_{k=1}^{m}$ is linearly dependent,

thence $\{r^{(\ell-k)}\}_{k=0}^{m}$ is linearly dependent. We conclude that affine dependence of $\{r^{(\ell-k)}\}_{k=0}^{m}$ implies linear dependence of $\{r^{(\ell-k)}\}_{k=0}^{m}$. Therefore, by contraposition, linear independence of $\{r^{(\ell-k)}\}_{k=0}^{m}$ implies affine independence of $\{r^{(\ell-k)}\}_{k=0}^{m}$, establishing sufficiency. To establish lack of necessity, we need only identify at least one instance $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ in which $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ is both linearly dependent and affinely independent. Before doing so, observe that we implicitly established sufficiency earlier during the discussion of constrained minimization in connection with the Ni thesis.

Assume that $\{r^{(\ell-k)}\}_{k=0}^{m}$ is affinely independent. We have seen earlier that this is a necessary and sufficient condition for there to be a unique affine combination $(\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$ of $\{r^{(\ell-k)}\}_{k=0}^{m}$ closest to 0. Define $\tilde{r}^{(\ell-k)} = r^{(\ell-k)} - (\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$, for $0 \leq k \leq m$. Observing that $\tilde{r}^{(\ell-k)} - \tilde{r}^{(\ell)} = r^{(\ell-k)} - r^{(\ell)}$, for $1 \leq k \leq m$, we conclude that $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ is also affinely independent. Moreover, we see that any affine combination of $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ is just the corresponding affine combination of $\{r^{(\ell-k)}\}_{k=0}^{m}$ minus $(\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$. Consequently, the same affine combination coefficients will yield the affine combinations of $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ and $\{r^{(\ell-k)}\}_{k=0}^{m}$ closest to 0, that affine combination of $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ being 0, so 0 is in the affine span of $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$. We infer that $\{\tilde{r}^{(\ell-k)}\}_{k=0}^{m}$ is both linearly dependent and affinely independent. Therefore, linear independence of $\{r^{(\ell-k)}\}_{k=0}^{m}$ is a sufficient, but not a necessary, condition for $\{r^{(\ell-k)}\}_{k=0}^{m}$ to be affine independent. Observe that $\{r^{(\ell-k)}\}_{k=0}^{m}$ can be nearly linearly dependent while $\{r^{(\ell-k)} - r^{(\ell)}\}_{k=1}^{m}$ is not nearly linearly dependent so $\{r^{(\ell-k)}\}_{k=0}^{m}$ is not nearly affinely dependent. Note the implications for the constrained minimization approach.

Before returning to the choice of the shift vector and associated deviation basis for the affine span of an affinely independent $\{r^{(\ell-k)}\}_{k=0}^{m}$, we shall sort out some issues regarding affine fixed point problems: $g(x) = Gx + h$. There is a unique fixed point $\hat{x}$ if $(I-G)$ is nonsingular, with $\hat{x} = (I-G)^{-1}h$. Sufficient conditions for the Picard iteration to converge to a unique fixed point for any h and any initial iterant $x^{(0)}$ are $\| G \| < 1$ or $\rho(G) < 1$; thence, these are also sufficient conditions for nonsingularity of $(I - G)$. Recall that g is invertible if G is nonsingular.

For $1 \leq k \leq m$, we have

$$g(x^{(\ell-k)}) - g(x^{(\ell)}) = G(x^{(\ell-k)} - x^{(\ell)}) = (y^{(\ell-k)} - y^{(\ell)}).$$

If $\{x^{(\ell-k)} - x^{(\ell)}\}_{k=1}^{m}$ is linearly dependent, there are nontrivial η_k, $1 \leq k \leq m$, such that $\sum_{k=1}^{m} \eta_k (x^{(\ell-k)} - x^{(\ell)}) = 0$. We see that

$$G\left[\sum_{k=1}^{m} \eta_k (x^{(\ell-k)} - x^{(\ell)})\right] = \sum_{k=1}^{m} \eta_k (y^{(\ell-k)} - y^{(\ell)}) = 0,$$

so linear dependence of $\{x^{(\ell-k)} - x^{(\ell)}\}_{k=1}^{m}$ implies linear dependence of $\{y^{(\ell-k)} - y^{(\ell)}\}_{k=1}^{m}$. Since the same η_k, $1 \leq k \leq m$, are involved for both sets, we also obtain $\sum_{k=1}^{m} \eta_k (r^{(\ell-k)} - r^{(\ell)}) = 0$, so linear dependence of $\{x^{(\ell-k)} - x^{(\ell)}\}_{k=1}^{m}$

implies linear dependence of $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$. We can rephrase these two inferences as affine dependence of $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$ implies affine dependence of $\left\{y^{(\ell-k)}\right\}_{k=0}^{m}$, and affine dependence of $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$ implies affine dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$. By contraposition, these two inferences become affine independence of $\left\{y^{(\ell-k)}\right\}_{k=0}^{m}$ implies affine independence of $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$, and affine independence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ implies affine independence of $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$.

The foregoing inferences depend only on the assumption that g is affine. We now assume that g is affine and invertible, so we also have

$$G^{-1}(y^{(\ell-k)} - y^{(\ell)}) = (x^{(\ell-k)} - x^{(\ell)}),\ 1 \leq k \leq m.$$

We see that linear dependence of $\left\{y^{(\ell-k)} - y^{(\ell)}\right\}_{k=1}^{m}$ implies linear dependence of $\left\{x^{(\ell-k)} - x^{(\ell)}\right\}_{k=1}^{m}$. Combined with the foregoing, we obtain linear dependence of $\left\{y^{(\ell-k)} - y^{(\ell)}\right\}_{k=1}^{m}$ if we have linear dependence of $\left\{x^{(\ell-k)} - x^{(\ell)}\right\}_{k=1}^{m}$; thence by contraposition, we obtain linear independence of $\left\{y^{(\ell-k)} - y^{(\ell)}\right\}_{k=1}^{m}$ if we have linear independence of $\left\{x^{(\ell-k)} - x^{(\ell)}\right\}_{k=1}^{m}$. This may be rephrased as the statements that we obtain affine dependence or independence of $\left\{y^{(\ell-k)}\right\}_{k=0}^{m}$ if we have affine dependence or independence of $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$, respectively. In addition, we infer that linear dependence of $\left\{y^{(\ell-k)} - y^{(\ell)}\right\}_{k=1}^{m}$, or equivalently, affine dependence of $\left\{y^{(\ell-k)}\right\}_{k=0}^{m}$, implies linear dependence of $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$, or equivalently, affine dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$. By contraposition, we see that linear independence of $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$, or equivalently, affine independence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ implies linear independence of $\left\{y^{(\ell-k)} - y^{(\ell)}\right\}_{k=1}^{m}$, or equivalently, affine independence of $\left\{y^{(\ell-k)}\right\}_{k=0}^{m}$.

Note that in the foregoing results affine dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ appears only as a conclusion, and affine independence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ appears only as a hypothesis. Thus, we identify circumstances in which $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely dependent, and consequences of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ being affinely independent. However, for $0 \leq k \leq m$, we have $r^{(\ell-k)} = (G-I)x^{(\ell-k)} + h$, and, for $1 \leq k \leq m$, we have $(r^{(\ell-k)} - r^{(\ell)}) = (G-I)(x^{(\ell-k)} - x^{(\ell)})$. If $(I-G)$, thence $(G-I)$, is nonsingular, it follows as above that $\left\{r^{(\ell-k)} - r^{(\ell)}\right\}_{k=1}^{m}$ is linearly dependent (independent) if $\left\{x^{(\ell-k)} - x^{(\ell)}\right\}_{k=1}^{m}$ is linearly dependent (independent), or equivalently, that $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely dependent (independent) if $\left\{x^{(\ell-k)}\right\}_{k=0}^{m}$ is affinely dependent (independent). Recall that, as a practical matter, near affine (linear) dependence is usually the more salient issue, so the condition number is relevant.

We now return to the choice of the shift vector and associated deviation basis for the affine span of an affinely independent $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$. As with our previous choices, we have no principled basis for choosing without relevant information from the problem context and about the anticipated consequences thereof.

In the nonstationary Extrapolation Algorithm, the one set of iterant data that is immune to being disregarded is that corresponding to the residual chosen as the shift vector. That set ought to be $x^{(\ell)}$ and $y^{(\ell)}$, so $r^{(\ell)}$ should be chosen as the shift vector;

thence also, imposition of the constraint $\hat{\theta}_0^{(\ell)} > 0$. The motivating assumption behind the Extrapolation Algorithm is that the underlying Picard iteration is converging and we seek to increase the rate of convergence. We anticipate that the younger residuals will eventually be significantly smaller than the older residuals so $(\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$ will be close to the younger residuals, whose $\hat{\theta}_k^{(\ell)}$ will dominate.

Having chosen $r^{(\ell)}$ as the shift vector, the associated deviation basis can be rescaled and reordered for numerical purposes, and the issue of actual or near affine dependence can be addressed. The associated deviation basis could be replaced by the corresponding difference basis, to exploit the natural ordering of the iterants. These matters have been discussed in detail in the main text.

Consider the affine subspace of dimension m defined as the affine span of the affinely independent set $\{r^{(\ell-k)}\}_{k=0}^m$. Why might one wish to consider a shift vector other than one of the $r^{(\ell-k)}$, namely an affine combination of $\{r^{(\ell-k)}\}_{k=0}^m$? How could we define, determine, and manipulate an associated deviation basis? How could we use this shift vector and associated deviation basis to find the unique point $(\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$ in the affine subspace closest to 0? How could we determine the corresponding unique affine combination coefficients such that

$$(\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = \sum_{k=0}^{m} \hat{\theta}_k^{(\ell)} r^{(\ell-k)},$$

and thence determine $\hat{u}^{(\ell)} = \sum_{k=0}^m \hat{\theta}_k^{(\ell)} x^{(\ell-k)}$ and $\hat{v}^{(\ell)} = \sum_{k=0}^m \hat{\theta}_k^{(\ell)} y^{(\ell-k)}$? We shall answer these questions hereafter.

We consider as shift vector the affine combination $s^{(\ell)} = \sum_{k=0}^m \sigma_k^{(\ell)} r^{(\ell-k)}$, with $\sum_{k=0}^m \sigma_k^{(\ell)} = 1$. We shall be primarily concerned with convex combinations: $\sigma_k^{(\ell)} \geq 0, 0 \leq k \leq m$. Of particular interest will be the centroid $\bar{s}^{(\ell)}$, with $\bar{\sigma}_k^{(\ell)} = (m+1)^{-1}$, $0 \leq k \leq m$. Since the affine span of $\{s^{(\ell)}\} \cup \{r^{(\ell-k)}\}_{k=0}^m$ coincides with that of $\{r^{(\ell-k)}\}_{k=0}^m$, $\{s^{(\ell)}\} \cup \{r^{(\ell-k)}\}_{k=0}^m$ is affinely dependent. Any member $(v^{(\ell)} - u^{(\ell)})$ of the affine span of $\{r^{(\ell-k)}\}_{k=0}^m$ can be written in the form

$$(v^{(\ell)} - u^{(\ell)}) = \sum_{k=0}^{m} \theta_k^{(\ell)} r^{(\ell-k)},$$

with $\sum_{k=0}^m \theta_k^{(\ell)} = 1$, and can also be written in the form

$$(v^{(\ell)} - u^{(\ell)}) = s^{(\ell)} + \sum_{k=0}^{m} \theta_k^{(\ell)} (r^{(\ell-k)} - s^{(\ell)}).$$

We identify $\{r^{(\ell-k)} - s^{(\ell)}\}_{k=0}^m$ as a deviation spanning set associated with shift vector $s^{(\ell)}$ for the affine span of $\{r^{(\ell-k)}\}_{k=0}^m$, or equivalently $\{s^{(\ell)}\} \cup \{r^{(\ell-k)}\}_{k=0}^m$. $\{r^{(\ell-k)} - s^{(\ell)}\}_{k=0}^m$ is not a deviation basis because it constitutes a set of $m+1$ vectors in an m dimensional linear subspace and must therefore be linearly dependent. Since $\{r^{(\ell-k)}\}_{k=0}^m$ is affinely independent, the members of the set are nonzero

and distinct. If all members of $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m}$ are nonzero, there is a nontrivial linear combination thereof equal to zero: $\sum_{k=0}^{m}\eta_k(r^{(\ell-k)} - s^{(\ell)}) = 0$. There are at least two (and at most $m+1$) j such that $\eta_j \neq 0$, so $(r^{(\ell-j)} - s^{(\ell)})$ can be expressed as a linear combination of the other members of $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m}$. There is at most one j such that $(r^{(\ell-j)} - s^{(\ell)}) = 0$. For each j, the linear span of $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m} - (r^{(\ell-j)} - s^{(\ell)})$ coincides with that of $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m}$. We identify $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m} - (r^{(\ell-j)} - s^{(\ell)})$ as a deviation spanning set associated with shift vector $s^{(\ell)}$. Since $\left\{r^{(\ell-k)} - s^{(\ell)}\right\}_{k=0}^{m} - (r^{(\ell-j)} - s^{(\ell)})$ constitutes a spanning set of m vectors in an m dimensional linear subspace, this spanning set must be linearly independent, thence a deviation basis.

When the shift vector is a member of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, the customary choice, there is only one associated deviation basis. When the shift vector is not a member of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, there are at least 2 and at most $m+1$ deviation bases. For shift vector $s^{(\ell)} = \sum_{k=0}^{m}\sigma_k^{(\ell)} r^{(\ell-k)}$, with $\sum_{k=0}^{m}\sigma_k^{(\ell)} = 1$, we see that there are as many deviation bases as there are nonzero $\sigma_k^{(\ell)}$, $0 \leq k \leq m$, because $\sum_{k=0}^{m}\sigma_k^{(\ell)}(r^{(\ell-k)} - s^{(\ell)}) = 0$. In particular, for $\bar{s}^{(\ell)}$, with $\bar{\sigma}_k^{(\ell)} = (m+1)^{-1}$, $0 \leq k \leq m$, there are $m+1$ deviation bases. The $N \times (m+1)$ matrix with columns $(r^{(\ell-k)} - \bar{s}^{(\ell)})$, $0 \leq k \leq m \ll N$, will have rank m, and any subset of m columns constitutes a deviation basis associated with $\bar{s}^{(\ell)}$. How do we choose among them? If we were to use the standard scaling and pivoting strategies to construct a QR decomposition/factorization of this matrix, we would select a particular deviation basis. Interest in using the centroid $\bar{s}^{(\ell)}$ as the shift vector arises from the expectation that the resulting deviation basis matrix will have a smaller condition number than that for the deviation basis associated with $r^{(\ell)}$. "Centering" of this sort is a common stratagem in many contexts. Whether the potential gain would be worth the modest incremental cost remains to be seen and would be problem dependent.

Specifically, suppose that we set out to use the Extrapolation Algorithm as laid out in the main text to seek the unique point $(\hat{v}^{(\ell)} - \hat{u}^{(\ell)})$ closest to 0 in the affine span of $\left\{\bar{s}^{(\ell)}\right\} \cup \left\{r^{(\ell-k)}\right\}_{k=0}^{m}$ using $\bar{s}^{(\ell)}$ as the shift vector. The standard scaling, pivoting, and narrow regularization strategies will cope with the actual affine dependence and choose an associated deviation basis, characterized by j. Ordering by increasing age and using the standard scaling strategy will ensure that $j > 0$, so the iterant data $x^{(\ell)}$ and $y^{(\ell)}$ will not be disregarded in choosing the deviation basis. We assume that the corresponding basic solution will be produced, so the solution can be written in the form

$$(\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = \left[1 - \sum_{k=0}^{m}\hat{\phi}_k^{(\ell)}\right]\bar{s}^{(\ell)} + \sum_{k=0}^{m}\hat{\phi}_k^{(\ell)} r^{(\ell-k)},$$

with the understanding that $\hat{\phi}_j^{(\ell)} = 0$ for the j characterizing the chosen deviation basis. We may then obtain

$$(\hat{v}^{(\ell)} - \hat{u}^{(\ell)}) = \sum_{k=0}^{m}\hat{\theta}_k^{(\ell)} r^{(\ell-k)},$$

where

$$\hat{\theta}_k^{(\ell)} = \hat{\phi}_k^{(\ell)} + \left[1 - \sum_{i=0}^{m} \hat{\phi}_i^{(\ell)}\right] \bar{\sigma}_k^{(\ell)},$$

for $0 \leq k \leq m$. The minimal solution could easily be used instead of the basic solution. Centering is particularly attractive for problems that exhibit oscillatory behavior of the residuals. For problems exhibiting monotonic behavior of the residuals, selection of $s^{(\ell)}$ closer to the younger residuals may be preferable. The algorithm could also accommodate near affine dependence of $\left\{r^{(\ell-k)}\right\}_{k=0}^{m}$, as discussed in the main text.

References

1. Anderson, D.G.: Iterative procedures for nonlinear integral equations. J. Assoc. Comput. Mach. **12**, 547–560 (1965)
2. Bierlaire, M., Crittin, F.: Solving noisy, large-scale fixed point problems and systems of nonlinear equations. Transp. Sci. **40**, 44–63 (2006)
3. Björck, Å.: Numerical Methods for Least Squares Problems. SIAM, Philadelphia (1996)
4. Broyden, C.J.: A class of methods for solving nonlinear simultaneous equations. Math. Comput. **19**, 577–593 (1965)
5. Calef, M.T., Fichtl, E.D., Warsa, B., Carlson, N.N.: Nonlinear Krylov acceleration applied to a discrete ordinates formulation of the k-eigenvalue problem. J. Comput. Phys. **238**, 188–209 (2013)
6. Carlson, N.N., Miller, K.: Design and application of a gradient-weighted moving finite element code I: one dimension. SIAM J. Sci. Comput. **19**, 728–765 (1998)
7. Eyert, V.: A comparative study on methods for convergence acceleration of iterative vector sequences. J. Comput. Phys. **124**, 271–285 (1996)
8. Fang, H.-r., Saad, Y.: Two classes of multisecant methods for nonlinear acceleration. Numer. Linear Algebra Appl. **16**, 197–221 (2009)
9. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. The Johns Hopkins University Press, Baltimore (2013)
10. Horn, R.A., Johnson, C.A.: Matrix Analysis. Cambridge University Press, Cambridge (1985)
11. Marks, L.D., Luke, D.R.: Robust mixing for ab initio quantum mechanical calculations. Phys. Rev. B **78**, 075114 (2008)
12. Ni, P.: Anderson Acceleration of Fixed-Point Iteration with Applications to Electronic Structure Computations, Ph.D. thesis, Worcester Polytechnic Institute, Worcester, MA (2009)
13. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. Academic Press, San Diego (1970)
14. Ostrowski, A.M.: Solution of Equations and Systems of Equations, 3nd edn. Academic Press, San Diego (1966)
15. Toth, A., Kelley, C.T.: Convergence analysis for Anderson Acceleration. SIAM J. Numer. Anal. **53**, 805–819 (2015)
16. Walker, H.F., Ni, P.: Anderson Acceleration for fixed-point iterations. SIAM J. Numer. Anal. **49**, 1715–1735 (2011)