EXCLUSION REGIONS AND FAST ESTIMATION OF PSEUDOSPECTRA

I. KOUTIS* AND E. GALLOPOULOS†

Abstract. The construction of an accurate approximation of the ϵ-pseudospectrum $\Lambda_\epsilon(A)$ of a matrix A by means of the standard grid method (GRID) is a highly demanding computational task, even for matrices of medium size. At each point of the domain of interest, (GRID) computes $\sigma_{\min}(zI - A)$ and uses that information in order to to classify the point as belonging to $\Lambda_\epsilon(A)$ or not. In that sense, GRID makes only "pointwise" use of the information it computes at each z. In this paper we prove that knowledge of the minimum singular triplet $[\sigma_{\min}(zI - A), u_{\min}, v_{\min}]$ at $z \in \mathbb{C}$ provides much more information that can be used to locate the pseudospectrum. In particular, from every z where we compute the triplet, we show that it is possible to construct "exclusion disks" that do not intersect the pseudospectrum. These results are used in the context of an "inclusion-exclusion" methodology to implement the rapid and judicious pruning of the initial domain enclosing the pseudospectrum. We propose two versions of the method and show that these lead to a substantial reduction of the cost of the computation while retaining the robustness and embarassingly parallel processing advantage of the standard grid approach.

AMS subject classifications. 65F20.

Key words. Matrix pseudospectrum, spectral portrait, modified grid method, exclusions, eigenvalues

1. Introduction and motivation.

The pseudospectrum, $\Lambda_\epsilon(A)$, of a matrix $A \in \mathbb{C}^{n\times n}$ has become a tool for the investigation of the behavior of several matrix-dependent algorithms, ranging from the behavior of iterative methods for large linear systems to the behavior of time-stepping algorithms. It is widely acknowledged, however, that computing pseudospectra with current technology is expensive, even when the matrix in question is of moderate size and this has led to several research activities attempting to address this problem. We refer to the recent paper by L.N. Trefethen ([18]) for a survey of the state-of-the-art.

If we denote by $\Lambda(A)$ the eigenvalues of A, and by $\sigma_{\min}(B)$ the minimum singular value of a matrix B, two equivalent definitions of $\Lambda_\epsilon(A)$ are given in Table 1.1. We call the standard reference method for computing pseudospectra GRID and note that it estimates Λ_ϵ at a region of the complex plane Ω by first discretizing the region with a grid Ω_h, then computing $\sigma_{\min}(z_k I - A)$ for all $z_k \in \Omega_h$, and finally plotting the ϵ-contours. This function is implemented as function pscont in the popular Test Matrix Toolbox for MATLAB [12]. Therefore, the cost of GRID is approximately equal to $T_{\texttt{GRID}} = |\Omega_h| C_{\sigma_{\min}}$, where $|\Omega_h|$ denotes the number of points of the grid and $C_{\sigma_{\min}}$ denotes the average cost for extracting the $\sigma_{\min}$.

It is clear that the cost of GRID becomes prohibitive as the grid or matrix sizes increase. Methods that attempt to reduce the cost of computing pseudospectra via a reduction of the factor $C_{\sigma_{\min}}$ we call *matrix-based*, while those that are based on a reduction of $|\Omega_h|$ we call *domain-based*.

[*] Department of Computer Science, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh , PA 15213. E-mail: jkoutis@cs.cmu.edu. Supported in part by the European INCO–COPERNICUS Scientific Program, Project STABLE: CP 960237 while the author was a student in Patras.

[†] Department of Computer Engineering and Informatics, University of Patras, 26500 Patras, Greece. E-mail: stratis@hpclab.ceid.upatras.gr. Supported in part by the European INCO–COPERNICUS Scientific Program, Project STABLE: CP 960237 and by the Greek General Secretariat for Research and Development, Project ΠΕΝΕΔ 99-07.

Definition 1:

$$\Lambda_\epsilon(A) = \{z \in C : \quad z \in \Lambda(A+E) \quad with \quad \|E\| \leq \epsilon, \epsilon \geq 0\}.$$

Definition 2:

$$\Lambda_\epsilon(A) = \{z \in C : \quad \sigma_{\min}(zI - A) \leq \epsilon, \epsilon \geq 0\},$$

TABLE 1.1
Equivalent definitions of the matrix pseudospectrum Λ_ϵ.

We focus here on domain-based methods, i.e. methods that attempt to use available information in order to reduce the number of points where it is necessary to compute the minimum triplets.

When the matrix is real, the simplest known method for reducing the work is domain-based and is used routinely, e.g. in the Test Matrix toolbox; this is none other than applying the fact that complex eigenvalues of real matrices come in conjugate pairs, therefore we only need to compute the pseudospectrum at either the positive or negative imaginary halfplanes.

Other than this obvious but effective approach, the two major domain based methods that exist in the literature are: *i*) the method of Gallestey [9], that starts from an initial region and then builds an approximation to the pseudospectrum by means of systematic pruning and/or expansion of that region. These steps are based on the subharmonicity of the function $\|(zI-A)^{-1}\|$. *ii*) The approach of Bruehl [5] to compute $\Lambda_\epsilon(A)$ by tracing its boundary $\partial\Lambda_\epsilon(A)$ using predictor-corrector path following; this method was further developed into a parallel and more robust algorithm by Bekas and Gallopoulos [2]. Path following has been shown to be very effective, but in some cases it might fail or miss certain parts of the pseudospectrum, e.g. when there are multiple components. Furthermore, even though it has been extended in [2] to offer large grain parallelism, the parallelism is moderate and will not readily scale to run on a large number of processors for a fixed matrix size and resolution of the pseudospectrum boundary.

In these respects, the standard workhorse `GRID` method, still appears to offer greater robustness and more parallelism. Unfortunately, much of the work in `GRID` is also redundant, since a portion of the gridpoints lie outside the region of interest. Recalling that the pseudospectral regions form nested sets, this means that many gridpoints lie outside $\Lambda_\epsilon(A)$ for the largest ϵ of interest.

In view of this, Braconnier et al. have studied the problem of approximating a region enclosing the eigenvalues which could also serve as an approximation to the pseudospectrum [4]. Amongst the four methods they examined - tabulated in Table 1.2 - they found that the one based on the field of values provides the most useful approximation to $\Lambda_\epsilon(A)$ - in terms of pertinent information and computational costs.

Regarding the methods of Table 1.2, we note that *i*) the first three produce regions Ω enclosing $\Lambda(A)$; on the other hand, there is no guarantee that $\Lambda_\epsilon(A) \subset \Omega$ as well - even though in many cases Ω is so large that it actually encloses the specific region; *ii*) all except the Gershgorin disks return convex shaped regions Ω.

It is worth noting also that a slight modification of the first - "eigenvalue" method - is the default in function `pscont` mentioned earlier. The vertices of the rectangle are computed by first obtaining the eigenvalues, getting their minimum and maximum real and imaginary parts, say $[\lambda^R_{\min}, \lambda^R_{\max}, \lambda^I_{\min}, \lambda^I_{\max}]$, and then defining the vertices

From eigenvalues: Use $\Omega = [\min_{i=1:n} Re(\lambda_i), \max_{i=1:n} Re(\lambda_i)] \times [-d, d] \supset \Lambda(A)$, where $d = max_{i=1:n}|\lambda_{i+1} - \lambda_i|$ and λ_i are the eigenvalues of A computed using a backward stable algorithm.

From matrix norms: Use $\Omega = D(0, \|A\|_2) \supset \Lambda(A)$.

From Gershgorin disks: Use

$$\Omega = \bigcup_{i=1:n} D(\alpha_{ii}, \sum_{i \neq j} |\alpha_{ij}|) \supset \Lambda(A)$$

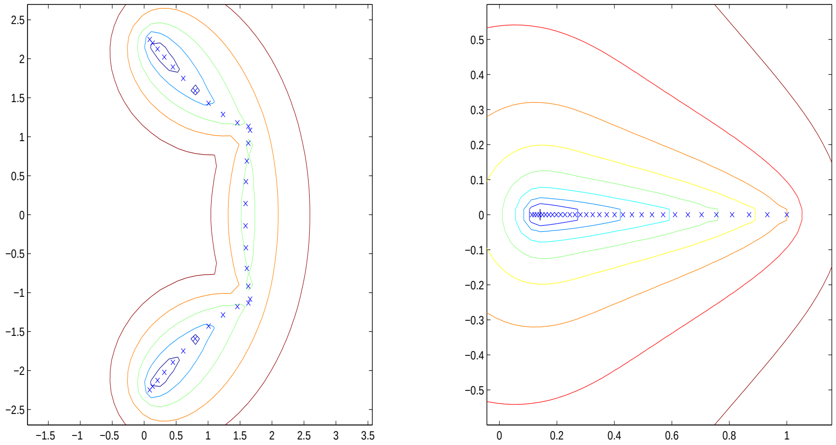
Field of values (FOV): Use

$$\Omega = \mathrm{FOV}(A) + D(0, \epsilon\|A\|_2) \supset \Lambda_\epsilon(A),$$

where $\mathrm{FOV}(A)$ denotes the field of values of A.

TABLE 1.2
Some methods for approximating a region Ω enclosing $\Lambda(A)$.

FIG. 1.1. *Pseudospectrum contours, computed using* `pscont`*, for matrices* `grcar` *(left) and* `kahan` *(right) of size $n = 32$ and several values of ϵ.*

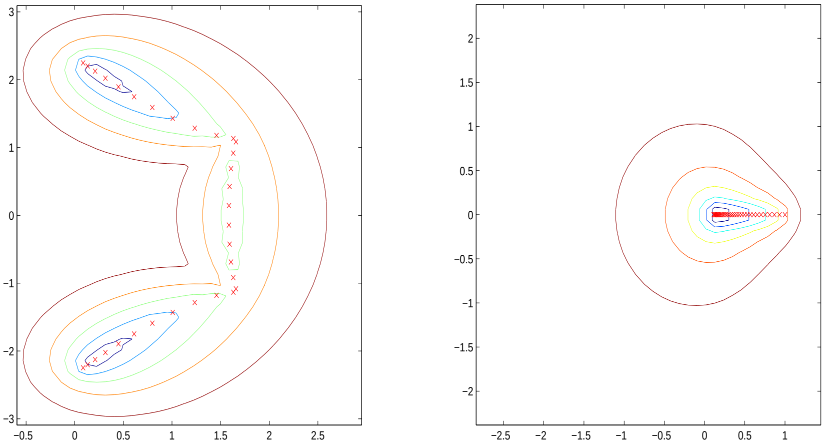


as $\lambda^R_{\min} - \alpha(\lambda^R_{\max} - \lambda^I_{\min})$, etc. where α is some expansion factor (the default in `pscont` is 0.1). It is easy, however, to construct examples where, for typical values of ϵ, the resulting Ω fails to include all of $\Lambda_\epsilon(A)$. For instance, Figure 1.1, shows the pseudospectrum contours generated using `pscont` for matrices (also from [12]) `grcar` and `kahan`, of size $n = 32$, corresponding to values of ϵ from 10^{-10} to 10^{-1}; unfortunately, some contours lie outside the region used by `pscont`.

If, on the other hand, one follows the field of values criterion described in Table 1.2 to estimate Ω, then it is certain that $\Lambda_\epsilon(A) \subset \Omega$. For that reason, in the remainder of this paper, we will use this criterion in its simplest form. In particular, we let the Hermitian, $A_H = \frac{A+A^*}{2}$, and skew Hermitian, $A_{SH} = \frac{A-A^*}{2}$, parts of A we use the known relations $FOV(A_H) = ReFOV(A)$ and $FOV(A_{SH}) = iImFOV(A)$ and let

$$\begin{aligned} \Omega = {} & [\lambda_{\min}(A_H) - \epsilon\|A\|, \lambda_{\max}(A_H) + \epsilon\|A\|] \\ & \times[\lambda_{\min}(A_{SH}) - \epsilon\|A\|, \lambda_{\max}(A_{SH}) + \epsilon\|A\|], \end{aligned} \tag{1.1}$$

FIG. 1.2. *Pseudospectrum contours, computed using* `pscont` *and the field of values criterion implemented by relation (1.1), for matrices* `grcar` *(left) and* `kahan` *(right) of size $n = 32$ and several values of ϵ.*



1. Obtain an inclusion region $\hat{\Omega} \supset \Lambda_\epsilon(A)$.
2. Compute set of exclusion regions Δ_j intersecting $\hat{\Omega}$, i.e. $\hat{\Omega} \cap \Delta_j \neq \emptyset$, so that $\Lambda_\epsilon(A) \cap \Delta_j = \emptyset, j = 1 : n$ and set $\Omega = \hat{\Omega} \backslash \cup_{j=1:n} \Delta_j$.
3. Discretize Ω and call the resulting grid $\Omega_h \subset \Omega$.
4. Compute the $\sigma_{\min}$'s on Ω_h.

TABLE 1.3
General Inclusion-Exclusion *methodology*

we are assured that Ω will fully enclose the pseudospectrum of A. Furthermore, the computation of the bounding rectangle is relatively easy since it only involves the computation of the norm of A and of the extreme eigenvalues of Hermitian matrices. On the other hand, this is a convex region that might cause many redundant computations in `GRID`. This is illustrated in Figure 1.2, where we used the above region together with function `pscont` for matrices `grcar` and `kahan`. Observe that in both cases `GRID` would need to perform computations on too many extraneous points: In the first case because of the mismatch of Ω (convex) with Λ_ϵ (non-convex), while in the latter because of a significant overestimate of the bounding region.

The above discussion motivates the design of methods that *i*) guarantee that the enclosing region contains the pseudospectrum and *ii*) the computational complexity is not severely penalized by any mismatch of the shapes of Ω and $\Lambda_\epsilon(A)$. The general approach that we propose in this paper we call Inclusion-Exclusion, abbreviated $\mathcal{IE}$, and has the general structure shown in Table 1.3. The idea is to start from an initial region that contains the pseudospectrum (we call it "inclusion region") and then proceed to prune it from extraneous points, that do not belong to the pseudospectrum. The question then becomes one of rapid computation of effective exclusion regions.

The original `GRID` method is a trivial case of $\mathcal{IE}$: It computes the singular value decomposition at every point of Ω and depending on the result it classifies a point as being inside or outside the pseudospectrum. In that sense, `GRID` makes only "pointwise" use of the information it computes at each z. In this paper we prove that knowledge of the minimum singular value provides much more information that can be applied to effectively remove larger regions. The process can be further enhanced if we know the corresponding minimum singular vectors. In particular, from every z where we compute the triplet, we show that it is possible to construct "exclusion disks" that do not intersect the pseudospectrum. These results are used in the context of the "inclusion-exclusion" methodology to implement an algorithm, we call *Modified Grid Algorithm* (`MoG`), for the rapid computation of the pseudospectrum. We propose two versions of the method, one based on removing regions that provably exclude the pseudospectrum and one based on removing regions that exclude larger regions that most often exclude the pseudospectrum.

We show that both variants of `MoG` substantially reduce the number of singular value evaluations while retaining the advantages of `GRID`, such as simplicity, robustness and independent computations lending themselves to parallelism of high granularity. Our numerical examples include standard test matrices of small size as well as larger matrices whose singular values are computed by means of ARPACK [15]. We also show that the algorithm is not very sensitive to the choice of the initial enclosing region, in the sense that even if the region is much larger than the largest sought pseudospectrum, it leads to very fast pruning of most of the extraneous regions.

The paper is organized as follows. In Section 2 we present the main theoretic results that enable the construction of `MoG`. In Section 3 we describe `MoG` and analyze its main characteristics. In Section 4 we present numerical experiments with `MoG`. Finally, in Section 5 we set forth directions for future research.

2. Theoretical background.

We first introduce some notation. Denote by $D(z,\varrho)$ and $D^\circ(z,\varrho)$ the closed and open disk respectively, with center z and radius ϱ. For any given (closed) disk D, we denote by D^c its complement. For nonsingular $A \in \mathbb{C}^{n\times n}$, $\mathrm{cond}(A) = \|A\|\|A^{-1}\|$ denotes the spectral condition number. For a given eigenvalue z_e of a matrix, the eigenvalue condition number or eigenvalue sensitivity is the reciprocal of the cosine of the angle between the right and left eigenvectors of A corresponding to z_e, i.e. $\kappa(z_e) = \frac{\|y^*\|\cdot\|x\|}{|y^*x|}$.

For any matrix B with singular value decomposition $B = U^*\Sigma V$, we will denote by $(\sigma_{\min}, u_{\min}, v_{\min})$ the triplet consisting of the minimum singular value and the corresponding left and right singular vectors of B and call it *minimum triplet*. We also use the letter N to denote a generic normal matrix.

We first collect known results that are for our presentation. One of the theorems in the original paper of Bauer and Fike has an immediate interpretation in terms of pseudospectra:

THEOREM 2.1. *[1, Thm.IIIa] . Let $A \in \mathbb{C}^{n\times n}$ be diagonalizable and $X^{-1}AX = \Lambda$. Let also $\|E\| = \epsilon$. Then the union of the disks $D(z_k, \mathrm{cond}(X)\cdot\epsilon)$ contains all the eigenvalues of $A+E$, that is $\Lambda_\epsilon(A) \subseteq \bigcup_i D(z_i, \mathrm{cond}(X)\cdot\epsilon)$.*

When the matrix is normal, Theorem 2.1 can be refined further:

COROLLARY 2.2. *Let $N \in \mathbb{C}^{n\times n}$ be normal and $z_i, i = 1,\dots,n$ its eigenvalues, then*

$$\Lambda_\epsilon(N) = \bigcup_i D(z_i, \epsilon).$$

Proof. Since N normal then $\mathrm{cond}(X) = 1$ and the rightward inclusion follows from Theorem 2.1. Let $Q^*NQ = \Lambda$ where Λ is diagonal and Q unitary. The equality follows since for any $z \in \bigcup_i D(z_i,\epsilon)$, the diagonal matrix $E = (z - z_i)I$ is such that

$\|E\| \leq \epsilon$ and one eigenvalue of $N+E$ is z since $Q^*(N+E)Q = \Lambda + (z-z_i)I$, where Λ is the diagonal matrix of eigenvalues of N, therefore, the eigenvalues of N that are equal to z_i are driven to z by the perturbation E. Furthermore $\|E\| = \|\tilde{\Lambda}\| = |z - z_i| \leq \epsilon$. □

We next list some properties of the pseudospectrum that will be useful in the sequel.

Properties 2.1.

1. *$\Lambda(A) \subset \Lambda_\epsilon(A)$ for $\epsilon > 0$.*
2. *$\epsilon < \epsilon_1 \Leftrightarrow \Lambda_\epsilon(A) \subset \Lambda_{\epsilon_1}(A)$.*
3. *If N is a normal matrix whose eigenvalues coincide with those of A, i.e. $\Lambda(A) = \Lambda(N)$, then $\Lambda_\epsilon(N) \subseteq \Lambda_\epsilon(A)$.*

The first two properties follow directly from the definition of the pseudospectrum. The last one is a consequence of Theorem 2.1, Corollary 2.2 and the facts that the condition number of any matrix is at least 1 and that minimum is attained by all normal matrices.

2.1. Localization theorems based on exclusions. In this section we develop the theoretical tools that will form the bases of our algorithms.

Definition 2.3. *Given a matrix A and $\epsilon \geq 0$, we define a region Δ of the complex plane to be an* exclusion region with respect to the ϵ-pseudospectrum of A *("exlusion region" for short when the context is clear) if Δ does not contain any point of $\Lambda_\epsilon(A)$.*

In many cases, the exclusion region will be a disk, in which case we will be referring to is as "exclusion disk".

Our first result provides a simple means for computing an exclusion region on the basis of $\sigma_{\min}(zI - A)$.

Theorem 2.4. *If $\sigma_{\min}(zI - A) = r > \epsilon$ then $D^\circ(z, r - \epsilon) \cap \Lambda_\epsilon(A) = \emptyset$.*

To prove the above theorem, we need the following lemma, whose proof can be found in [13].

Lemma 2.5. *Let $B, E \in \mathbb{C}^{n \times n}$. Then the following inequality holds:*

$$|\sigma_{\min}(B + E) - \sigma_{\min}(B)| \leq \sigma_{\max}(E)$$

We can now prove the previous theorem.

Proof. Let $\sigma_{\min}(zI - A) = r > \epsilon$ and $\tilde{z} \in D^\circ(z, r - \epsilon)$. We need to show that $\sigma_{\min}(\tilde{z}I - A) > \epsilon$. Then we can write $\tilde{z}I - A = zI - A - (zI - \tilde{z}I)$. Let $B = zI - A$ and $E = zI - \tilde{z}I$. Using Lemma 2.5 it follows that

$$|\sigma_{\min}(\tilde{z}I - A) - \sigma_{\min}(zI - A)| \leq \sigma_{\max}(E) = |z - \tilde{z}|.$$

Therefore

$$-|\tilde{z} - z| \leq \sigma_{\min}(\tilde{z}I - A) - \sigma_{\min}(zI - A)$$

and since $\sigma_{\min}(zI - A) = r$ and $|z - \tilde{z}| < r - \epsilon$ the result follows. □

The above result tells us that if at any $z \in \mathbb{C}$ we have computed $\sigma_{\min}(zI - A) = r$, then $D(z, r - \epsilon)$ is an exclusion disk of A. Clearly, the knowledge of exclusion regions can reduce the number of points where $\sigma_{\min}$ is sought.

We next show that when a matrix is normal, the exclusion region obtained from Theorem 2.4 is the disk with maximum radius that we can create that will not contain any points of the pseudospectrum. In that sense, assuming that we follow the theorem

	$A(0.2)$	$A(0.4)$	$A(0.6)$	$A(0.8)$
$z_1 = 0.5 + 0.5i$	1.49e-03	1.02e-02	5.92e-02	1.72e-01
$z_2 = 0.2 + 0.2i$	6.03e-07	1.78e-04	6.06e-03	5.32e-02
$z_3 = -0.2 + 0.5i$	4.01e-09	2.36e-06	2.19e-04	3.91e-03

TABLE 2.1

Values of $\sigma_{\min}(z_kI - A(\alpha))$ *where* $A(\alpha)$=`pentoep(32,0,`α`,0,0,1)`.

and obtain exclusion regions that are disks, the exclusions will be maximal, in the sense that they "touch" $\Lambda_\epsilon(A)$ and therefore cannot be expanded any further.

COROLLARY 2.6. *If* $N \in \mathbb{C}^{n\times n}$ *is normal with eigenvalues* $z_i, i = 1, \ldots, n,$ *and* $\sigma_{\min}(zI - N) = r > \epsilon$ *then the boundary* $\partial D(z, r - \epsilon)$ *contains at least one point of* $\Lambda_\epsilon(N)$.

Proof. From the definitions, it follows that $z \in \Lambda_r(N)$. Theorem 2.2 implies that $\Lambda_r(N) = \bigcup_i D(z_i, r)$, therefore there is some eigenvalue z_k for which $z \in \partial D(z_k, r)$. Therefore the disks $D(z_k, \epsilon)$ and $D(z, r - \epsilon)$ have exactly one point in common. The proof follows from the observation that $D(z_k, \epsilon) \subset \Lambda_\epsilon(N)$. □

The corollary also extends a result of Hald ([10]) and hence the list of necessary conditions for normality presented in [6].

The following lemma is also useful:

LEMMA 2.7. *If* $N \in \mathbb{C}^{n\times n}$ *is normal then for any* $A \in \mathbb{C}^{n\times n}$ *that has exactly the same eigenvalues as* N *and any given* z*, it holds that* $D(z, r_A - \epsilon) \subset D(z, r_N - \epsilon)$*, assuming that* $r_A = \sigma_{\min}(zI - A)$ *and* $r_N = \sigma_{\min}(zI - N) \geq 0$*. Furthermore, the exclusion disk* $D(z, r - \epsilon)$ *will be the same for any other normal matrix that has exactly the same eigenvalues.*

Proof. From standard results $|\lambda_{\min}(A)|^{-1} \leq \|A^{-1}\| = 1/\sigma_{\min}(A)$. Therefore $\sigma_{\min}(A) \leq |\lambda_{\min}(A)| = \sigma_{\min}(N)$. Furthermore, if $\hat{N}$ is also normal and has the same eigenvalues as N then $\sigma_{\min}(N - zI) = |\lambda_{\min}(N - zI)| = |\lambda_{\min}(\hat{N} - zI)| = |\sigma_{\min}(\hat{N} - zI)|$. □

The above lemma tells us that for a given spectral set, normal matrices achieve maximal exclusion region. When the matrix is highly non-normal, however, the exclusion region suggested by Theorem 2.4 becomes smaller and hence less effective. This is illustrated in Table 2.1, which shows the values of $\sigma_{\min}(zI - A)$ for several points z and matrices A that are increasingly non-normal, the measure used being the Frobenius norm $\nu(A) = \|A^*A - A^*A\|_{\mathrm{F}}$ [11]. Specifically, we use the matrices $A(\alpha)$=`pentoep(32,0,`α`,0,0,1)` [12]. We note that the value of α has little effect on the distribution of the eigenvalues of $A(\alpha)$. On the other hand, for this matrix we observed that as α decreases, the condition of the eigenvalues (computed by the function `eigsens` of Test Matrix Toolbox) as well as the spectral condition number of $A(\alpha)$ increase monotonically.

An interesting interpretation of the numbers in Table 2.1 is that increasing non-normality leads to faster expansion of the pseudospectrum. This motivates us to define a measure of local sensitivity for the pseudospectrum.

DEFINITION 2.8. *Let* $g_A(z) = \sigma_{\min}((x + iy)I - A)$ *where* $z = x + iy \in C \setminus \Lambda(A)$*. We call* sensitivity of the pseudospectrum of a matrix A at a point z_0 *the inverse of the magnitude of the gradient of* g_A *at* z_0*, that is*

$$pss_A(z_0) = |\nabla g_A(z)|^{-1}_{z=z_0}. \tag{2.1}$$

Subsequently, we will use the notation $pss(z)$ whenever is clear from the context to which matrix we refer. We next investigate the properties of $pss(z)$. When $z = x + iy \in C \setminus \Lambda(A)$, what we call pseudospectrum sensitivity is immediately related to the sensitivity of the minimum (simple) singular value, according to the definition of Sun in [8]. Therein, the sensitivity of a simple, non-zero, $\sigma_{\min}(zI - A)$ with respect to x and y was defined as the pair of absolute values $|\frac{\partial}{\partial x}\sigma_{\min}(xI + iyI - A)|$ and $|\frac{\partial}{\partial y}\sigma_{\min}(xI + iyI - A)|$.

How difficult is to compute the above pseudospectrum sensitivity? Not much, as the following following theorem, stated in [5], indicates.

THEOREM 2.9. *Let $x+iy \in C \setminus \Lambda(A)$. Then $g(x, y)$ is real analytic with respect to each coordinate x and y in a neighborhood of (x, y) if $\sigma_{\min}((x + iy)I - A)$ is a simple singular value. Furthermore, the gradient of g at z is*

$$\nabla g(x, y) = (Re(v^*_{\min}u_{\min}), Im(v_{\min}u_{\min})) = v^*_{\min}u_{\min}.$$

It follows that $pss(z)$ is computable from the left and right minimum singular vectors, which in turn can be computed along with the minimum singular value. In particular:

COROLLARY 2.10. *Using the above notation, at any point z where $\sigma_{\min}(zI - A)$ is simple, the pseudospectrum sensitivity is*

$$pss(z) = \frac{1}{|v^*_{\min}u_{\min}|} \geq 1.$$

The inequality follows directly from the definition of the inner product and the fact that the singular vectors are of unit magnitude.

Since any normal matrix is diagonalizable by means of orthogonal similarity transformations, Corollary 2.10 implies the following:

COROLLARY 2.11. *If N is normal then $pss_N(z) = 1$ at every point $z \in C \setminus \Lambda(A)$.* Therefore, the minimum value of pss is 1 and is attained at all points where g is analytic by the class of normal matrices. It is worth noting that Theorem 2.9 was also fundamental in establishing the original path following method of Bruehl for computing pseudospectra [5].

The fact that $g(x, y)$ is real analytic gives rise to an interesting association between its values and paths in the complex plane. In particular, we prove the following:

THEOREM 2.12. *Let $z_0 = x_0 + iy_0$ and $z_1 = x_1 + iy_1$ be two points and let pss be defined on those as well as on all points of the line segment L connecting them. If $pss(z_0) < pss(z)$ for every point z in L then*

$$|g(x_1, y_1) - g(x_0, y_0)| \cdot pss(z_0) \leq |z_1 - z_0|$$

Proof. By definition $g(x, y) - g(x_0, y_0) = \oint_{(x_0,y_0)}^{(x,y)} \nabla g(x, y)\vec{n}_L dL$, where $\vec{n}_L$ is the unit vector parallel to L oriented from z_0 to z_1. Then using Theorem 2.9

$$|g(x_1, y_1) - g(x_0, y_0)| = |\oint_{(x_0,y_0)}^{(x_1,y_1)} \nabla g(x, y) \cdot \vec{n}_L \; dL| \leq \oint_{(x_0,y_0)}^{(x_1,y_1)} |\nabla g(x, y) \cdot \vec{n}_L| \; dL$$
$$= \oint_{(x_0,y_0)}^{(x_1,y_1)} pss(z)^{-1} \; dL \leq pss(z_0)^{-1} \oint_{(x_0,y_0)}^{(x_1,y_1)} dL = pss(z_0)^{-1} \cdot |z_1 - z_0|.$$

□

Following the notation used in Theorem 2.12 we give a corollary that will be useful in the localization of $\Lambda_\epsilon(A)$. Let $R = (g(z_0) - \epsilon) \cdot pss(z_0)$.

COROLLARY 2.13. *Let $z_1 \in \partial\Lambda_\epsilon(A)$. If the assumptions of Theorem 2.12 hold, then $D^c(z_0, R) \cap \Lambda_\epsilon(A) \neq \emptyset$. If furthermore z_1 is a point which minimizes the distance $d(z_0, \Lambda_\epsilon(A))$, then $D^\circ(z_0, R)$ is an exclusion disk.*

If A is normal, then $R = g(z_0) - \epsilon$ because of Corollary 2.11. In that case, we know from Theorem 2.4 that the disk $D^\circ(z_0, R)$ is an exclusion disk, hence Corollary 2.13 reduces to Theorem 2.4, which is known to hold without the need to impose the special assumptions of Theorem 2.12. In order to use Corollary 2.13, however, it becomes necessary to discuss the implications of the aforementioned assumptions.

We next extend the definition of pseudospectrum sensitivity at points, z_e, that are eigenvalues of A. Then $z_eI - A$ would be singular, therefore $\sigma_{\min}(z_eI - A) = 0$. Theorem 2.9 specifically excluded points of the spectrum because it was based on a result of Sun ([7]) that did not cover the case of zero singular values, which, in general, are not differentiable. Different analyses (see [7] and [17]) have shown that in the neighborhood of a simple zero singular value, the following expansion holds:

$$\begin{aligned}\sigma^2_{\min}(zI - A) &= |v^*_{\min} E u_{\min}|^2 + O(\|E\|^3), \quad \text{where } E = (z - z_e)I,\\ &= |v^*_{\min} u_{\min}|^2 |z - z_e|^2 + O(\|E\|^3),\end{aligned}$$

from which it follows that

$$|\frac{|\sigma_{\min}(zI - A) - \sigma_{\min}(z_eI - A)|}{|z - z_e|} - |v^*_{\min} u_{\min}|| = O(\|E\|^{1/2}), \tag{2.2}$$

which tends to 0 as $z \to z_e$. Therefore, $|\nabla\phi(z)|_{z=z_e} = |v^*_{\min} u_{\min}|$, consequently, $pss(z_e) = \frac{1}{|v^*_{\min} u_{\min}|}$. We are now in the position to prove the following:

THEOREM 2.14. *Let z_e be a simple eigenvalue of $A \in \mathbb{C}^{n\times n}$. Then*

$$\lim_{z \to z_e} pss(z) = \kappa(z_e)$$

Proof. We have already shown that $\lim_{z\to z_e} pss(z) = pss(z_e) = \frac{1}{|v^* u|}$. Therefore, it is enough to show that $\frac{1}{|v^* u|} = \kappa(z_e)$. As before, matrix $A - z_eI$ is singular, therefore it has a zero eigenvalue and its minimum singular value is 0. We also assumed that z_e is a simple eigenvalue. Then $(A - z_eI)v_{\min} = \sigma_{\min}(A - z_eI)u_{\min} = 0$, therefore $Av_{\min} = z_e v_{\min}$. This implies that $v_{\min}$ is parallel to x, the right eigenvector of A corresponding to z_e. In the same manner, $u^*_{\min}(A - z_eI) = 0$, therefore $u^*_{\min}A = z_e u^*_{\min}$. Hence $u^*_{\min}$ is parallel to the left eigenvector y^*. Since $u_{\min}, v_{\min}$ are unit vectors, it follows that

$$|v^*_{\min} u_{\min}| = |\frac{x^*}{\|x\|}\frac{y}{\|y\|}| = \frac{|x^* y|}{\|x\|\|y\|},$$

which proves the assertion. □

We now give some general characteristics of the behavior of $pss_A(z)$ which are independent from A, and which can justify the use of Corollary 2.13 as a localization tool, even when it is not known whether the prerequisite assumptions hold.

1. The function pss is continuous, therefore, if z is a point at which $pss(z) < pss(z_e) = \kappa(z_e)$, the line segment from z to z_e is expected to contain a neighborhood in which the hypothesis of Theorem 2.14 holds.
2. Let $Q^*AQ = \Lambda + R$, where R strictly upper triangular be a Schur form of A. Then $\|R\|$ can be used as a measure of A's the departure from normality.

Since $Q^*(zI-A)Q = zI-\Lambda-R = z(I-\frac{\Lambda+R}{z})$, it follows that for points z that are far from the spectrum, matrix $I-\frac{A}{z}$ becomes close to being diagonal and thus normal. Since $pss(z)$ is continuous and $pss(z)=1$ for normal matrices (see Corollary 2.11), $pss(z)$ will approach 1 from above as z becomes very distant from the spectrum. For such large z, the exlusion disk predicted by Corollary 2.13 will be of size that is similar to the exclusion disk predicted by the original Theorem 2.4. It is thus possible, at those points, to apply only the latter theorem, thus avoiding the computation of singular vectors that appear to be needed to apply the former Corollary.

3. Since $g(z)$ is differentiable and real-valued, the mean value theorem states that there is $\xi \in L$, the line segment between z_0 and z_1, where $|g(z_0) - g(z_1)| = \frac{|z_0-z_1|}{pss(\xi)}$. The basic assumption of Theorem 2.12 can be relaxed to $pss(z) \geq pss(\xi)$ corresponding to an exclusion disk centered at ξ with radius $R = |g(z_1) - \epsilon| pss(\xi)$.

The first observation above also implies that if the conditions of the Theorem cannot be guaranteed, it is also possible that there will be *erroneous exclusions* , i.e. the exclusion disk will contain parts of Λ_ϵ(A). We next conduct experiments to show that in practice, this situation rarely occurs. In particular, we used MATLAB and a 100 × 100 grid and checked the gridpoints excluded by means of Corollary 2.13 for four matrices, from [12], that are commonly used to test pseudospectrum codes (`propeller, triangle, smoke, grcar`). Figure 2.1 illustrates, for given ϵ, the eigenvalues of each matrix (denoted by "`x`"), as well as the grid points (denoted by "·") that are erroneously considered not to belong to the pseudospectrum because they fall inside the exclusion disk corresponding to Corollary 2.13. It is clear from the illustrations that relatively few points are excluded erroneously, and that most are clustered in the neighborhood of eigenvalues. Since the $\mathcal{IE}$ methodology is principally designed to provide a fast pruning of the region, such points are of little interest anyway. We conclude that even when the assumptions leading to Corollary 2.13 cannot be checked, the computed disks are very likely to be exclusion regions and can be used as long as we have a mechanism in place to correct any wrongly excluded points.

3. The Modified Grid Algorithms.

In the remainder of this paper we demonstrate the effectiveness of these ideas. We describe two simple implementations of `MoG` that automate the process of pruning. We emphasize that our objective was to see the concepts established in the discussion so far in practice rather than providing the most efficient implementation.

For a given grid, the objective of the proposed algorithms is to classify gridpoints as "`in`", if they belong $\Lambda_\epsilon(A)$ or as "`out`" otherwise. The algorithm sweeps the gridpoints in some predefined order, e.g. from left to right and from top to bottom, the gridpoints, and marks the points appropriately. Initially, all points are marked as belonging to the pseeudospectrum. If for the point z that is currently under consideration it is found that $\sigma_{\min}(zI-A) \leq \epsilon$, then the point lies in the pseudospectrum. Otherwise, $\sigma_{\min}(zI-A) > \epsilon$, in which case we have two alternatives:

Standard exclusion: We compute the disk centered at z with radius $r = \sigma_{\min}(zI-A) - \epsilon$ and mark all gridpoints within the disk as not belonging to the pseudospectrum. According to Theorem 2.4, these points can safely be excluded from future consideration.

Large exclusion: We compute the disk centered at z with radius $R = (\sigma_{\min}(zI-A) - \epsilon)pss(z)$. According to Corollary 2.13, z is not in the pseudospectum

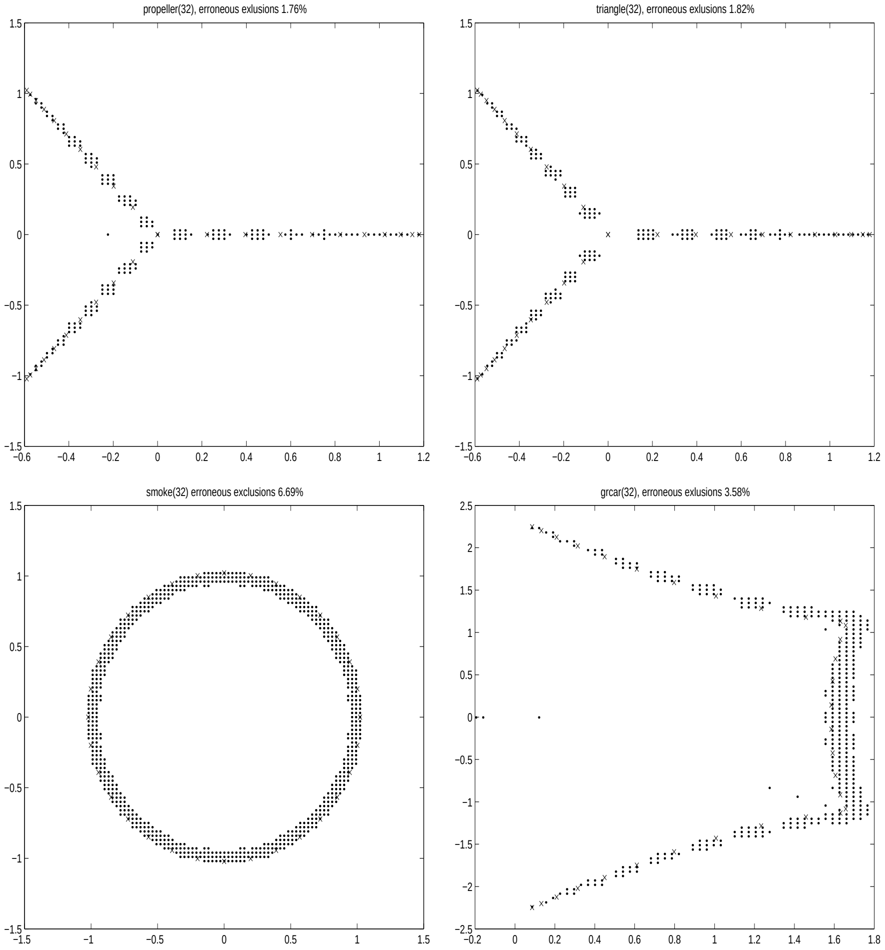


FIG. 2.1. *Erroneous exclusions and their frequency.*

while all other points inside the disk are marked as candidates for exclusion; in fact, as the numerical evidence presented above suggests, in all likelihood, these points will be outside the pseudospectrum.

In the `MoG` algorithm, presented in Table 3.1, variable `mmx` is set by the user to select one of the above versions. When `mmx=1`, we must apply a correction procedure, to recover any points that were mistakenly excluded. There are several alternative ways for setting up a systematic correction procedure, applied whenever `mmx = 1`, to recover any points that were erroneously excluded by Corollary 2.13; see [9] for one such example. Our correction consists of two parts.

`proc_corr_A` Before applying an exclusion, check whether $D^\circ(z,R)$ would contain points $\hat{z}$ for which it is already known that they satisfy $\sigma_{\min}(\hat{z}I-A)<\epsilon$. If yes, exclude only the gridpoints in the disk $D^\circ(z,\sigma_{\min}(zI-A))$.

`proc_corr_B` Inspect all excluded gridpoints that are adjacent to points that are marked as "`in`" and check if they satisfy $\sigma_{\min}(zI-A)<\epsilon$. For any point that does, find the center, $\hat{z}$, of the disk that led to its exclusion, mark again all points of the excluded disk as "`in`" and repeat the exclusion using the (smaller) disk

```
1. choose enclosing region $\Omega \supseteq \Lambda_\epsilon(A)$,
   discretize it with $\Omega_h$ and mark all points of $\Omega_h$ as in.
2. for every grid point $z \in \Omega_h$
      compute $\sigma = \sigma_{\min}(zI - A)$.
      if $\sigma > \epsilon$ then
          case mmx = 0: mark all points in $D^\circ(z, \sigma - \epsilon)$ as out
                        and set $\Omega_h = \Omega_h \setminus D^\circ(z, \sigma - \epsilon)$
          case mmx = 1: mark all points in $D^\circ(z, pss(z)(\sigma - \epsilon))$ as out
                        and set $\Omega_h = \Omega_h \setminus D^\circ(z, pss(z)(\sigma - \epsilon))$.
                        Apply proc_corr_A
   end for
3. if mmx = 1 apply proc_corr_B.
4. Draw contours for the selected values of $\epsilon$ at all points marked as "in".
```

TABLE 3.1
The MoG *algorithm.*

```
1. Choose $\Omega$ and discretize $\Omega_h \subset \Omega$.
2a. On each grid column move downwards using exclusions.
    Stop when the first point inside $\Lambda_\epsilon(A)$ is encountered.
2b. On each grid column move upwards using exclusions.
    Stop when the first point inside $\Lambda_\epsilon(A)$ is encountered.
3a. On each grid line move rightwards using exclusions.
    Stop when the first point inside $\Lambda_\epsilon(A)$ is encountered.
3b. On each grid line move leftwards using exclusions.
    Stop when the first point inside $\Lambda_\epsilon(A)$ is encountered.
```

TABLE 3.2
The MoGB *algorithm for approximating the pseudospectrum boundary.*

$D^\circ(\hat{z}, \sigma_{\min}(zI - A))$. Repeat until no further erroneous exclusions appear.

We finally note that if we are only interested to estimate the boundary $\partial\Lambda_\epsilon(A)$ for some given value of ϵ, MoG can be easily modified to approximate such a boundary at much less cost than its full version. The idea is to apply exclusions from the outer boundary of the enclosing region Ω moving inwards, in some systematic manner, until $\Lambda_\epsilon(A)$ is reached. We call this, the *Modified Grid Boundary Algorithm* (MoGB) and present it in Table 3.2.

3.1. MoG basic characteristics and analysis.. The cost of MoG is approximately equal to

$$T_{\texttt{MoG}} = (|\Omega_h| - |X_h|)C_{\sigma_{\min}}, \tag{3.1}$$

where $|\Omega_h|$ denotes the number of points of the grid, X_h denotes the number of points of the grid that were excluded, and $C_{\sigma_{\min}}$ denotes the average cost for extracting the $\sigma_{\min}$. The term $|X_h|$ depends clearly on ϵ, and on the specific topology of Λ_ϵ. Thus, the task of determining approximately the size of $|X_h|$ is difficult and we do not pursue such an analysis here. On the other hand, MoG retains the advantages of GRID. It is robust, since erroneous exclusions can be corrected with not much difficulty, and accurate, in the sense that for a given grid detail, it will produce the best possible approximation to $\Lambda_\epsilon(A)$. Like GRID, MoG contains a great amount of large-grain parallelism. As we will see, the use of exclusion theorems, dramatically reduces the cost, which becomes practically independent from the choice of the initial region Ω.

Regarding MoGB, we need to compare it with path following methods, which are considered the most economical means for computing the pseudospectrum boundary. The two advantages of MoGB are *i*) that it is able to compute all connected components of $\partial\Lambda_\epsilon$, as long as the initial enclosing region already contains them; *ii*) MoGB has much more potential for large-grain parallelism, compared to path following, where this potential is only moderate [2]. Thus, MoGB could be an attractive alternative to path following. A disadvantage is that without further enhancements, MoGB will only access boundary points $z^* = (x^*, y^*)$ that have the following property: At least one of the lines $x = x^*, y = y^*$ reaches from a boundary gridpoint of Ω_h to z^* without encountering on the way any other point of the pseudospectrum.

Finally, we make some remarks concerning the domain-based algorithm of Gallestey (called "SH" in [9]) which shares some features with MoG. SH also attempts to compute exclusion regions. This is achieved using the fact that the function $\|(zI-A)^{-1}\|$ is subharmonic in regions that do not include any eigenvalues and therefore obeys a maximum principle. For a given value of ϵ, SH, like MoG, labels every gridpoint z contained in $\Lambda_\epsilon(A)$ with the corresponding $\sigma_{\min}(zI-A)$. On the other hand, SH starts from a region that does not necessarily enclose all of Λ_ϵ on which it applies a combination of subdivisions, exclusions and expansions to approximate the pseudospectrum. As a result, it differs from the $\mathcal{IE}$ methodology. The exclusion regions are rectangles instead of disks. The exclusions are performed using a discrete form of the maximum principle; because of discretization errors, the process always requires the application of correction steps. This is in contrast to our method, where corrections are either unnecessary version (MoGO) or rarely needed.

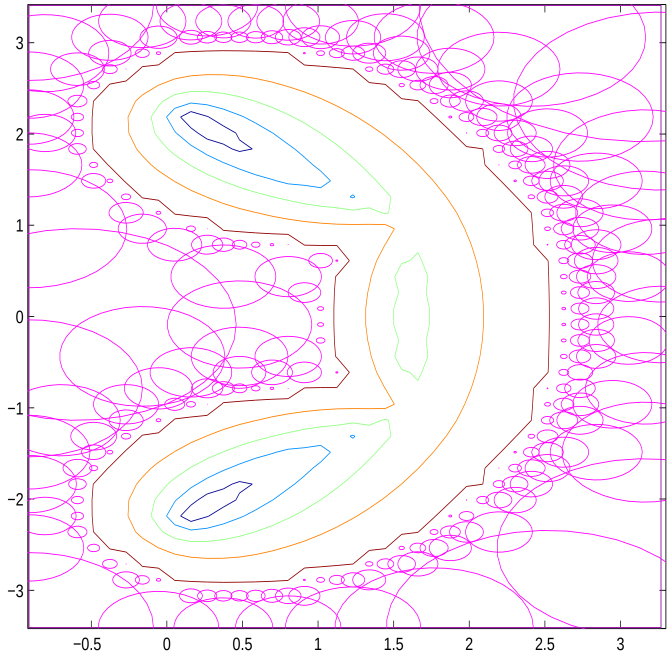
4. Numerical Results. We have experimented with MoG using codes developed in MATLAB 5.3 and running on a 700 MHz Pentium based system. All test matrices are from the Test Matrix Toolbox [12].

We used as Ω and grid Ω_h the rectangular regions computed by pscont using a user-set gridsize. The minimum triplets were computed using MATLAB's intrinsic svd function. In the sequel, in order to distinguish the performance of the two versions of MoG, we call MoGO the first version, based on the guaranteed exclusions of Theorem 2.4 to distinguish from the second version, based on the aggressive strategy of Corollary 2.13. In all examples the grid was swept in a natural order, from left to right and from top to bottom.

The first three experiments are with matrices kahan and grcar of size $n = 32$, $\epsilon = 1e-1$ and a 40×40 grid. Figure 4.1 shows the disks that are generated from the application of Theorem 2.4 and the contours that result from those points that do not belong to any exclusion disk. In this particular experiment, we did not exploit the fact that the image is symmetric with respect to the real axis. The total number of $\sigma_{\min}$ computations was 894 as opposed to 1600 for GRID, highlighting the effectiveness of the method.

The next experiment shows the impact of Theorem 2.4 (Figure 4.2 and of Corollary 2.13 (Figure 4.3) to matrix kahan. The center of each disk is marked with '+'. The total number of $\sigma_{\min}$ computations using the former was 372, whereas for the latter it was 273. The initial region is marked in the rectangle with thick sides, while the rest we provide for convenience. As you can see, in addition to the reduction to the

FIG. 4.1. *Application of Theorem 2.4 using grid of size* 40×40 *for matrices* `grcar` *of size* $n = 32$ *and* $\epsilon = 1e-1$.



matrix	region Ω	`MoG0`	`MoG`
`grcar(n)`	$[-0.91, 3.27] \times [-3.41, 3.41]$	659	617
`kahan(n)`	$[-3.29, 1.89] \times [-2.83, 2.83]$	248	183
`chebspec(n,0)`	$[-428.49, 428.49] \times [-317.65, 317.65]$	220	86
`chebspec(n,1)`	$[-456.45, 164.68] \times [-337.55, 337.55]$	154	60
`chow(n)`	$[-2.63, 19.60] \times [-12.22, 12.22]$	228	135
`triangle(n)`	$[-0.87, 1.37] \times [-1.22, 1.22]$	776	762
`propeller(n)`	$[-1.16, 1.63] \times [-1.50, 1.50]$	748	723
`fish(n)`	$[-0.62, 3.82] \times [-1.70, 1.70]$	697	675

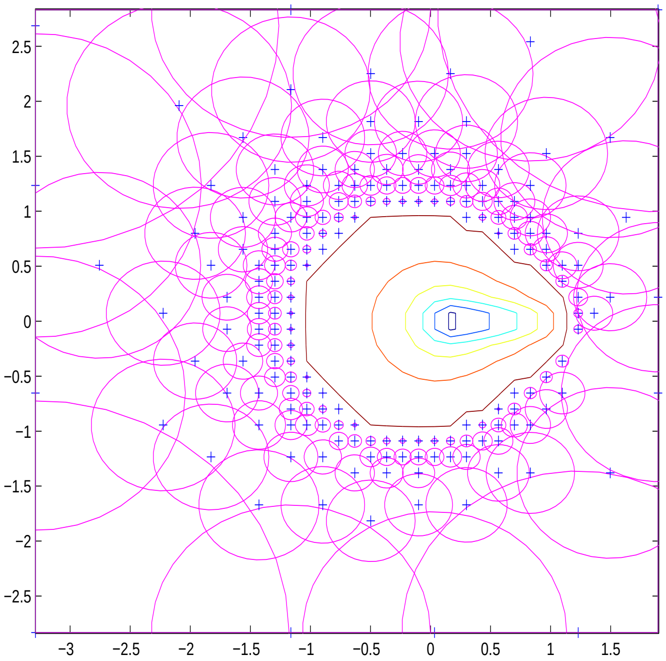
TABLE 4.1

Number of $\sigma_{\min}$ *evaluations in a* 50×50 *grid using the two versions of algorithm* `MoG` *for* $\epsilon = 1e-1$ *and* $n = 32$.

number of triplet computations, these diagrams suggest that there is little damage in overestimating the size of the domain.

The next experiments are with several matrices with interesting pseudospectra, all of size $n = 32$, using for initial domain Ω the rectangles defined by the field of values criterion for two different $\epsilon = 1e-1$ and $1e-3$. Once again, the improvements in performance are significant. Tables 4.1 and 4.2 show the number of $\sigma_{\min}$ evaluations using both versions of the algorithm: The conservative, based on Theorem 2.4, as well as the more aggressive one based on Corollary 2.13. In the latter case, the algorithm was instrumented to correct any erroneous exclusions. We observe that both versions of the algorithm offer significant reductions, while the aggressive version at times achieving threefold reductions in the number of $\sigma_{\min}$ calculations.

Fig. 4.2. *Application of Theorem 2.4 using grid of size* 40×40 *for matrices* `kahan` *of size* $n = 32$ *and* $\epsilon = 1e - 1$. *The center of each disk is marked with '+'.*



matrix	region Ω	`MoG0`	`MoG`
`grcar(n)`	$[-0.59, 2.95] \times [-3.09, 3.09]$	678	397
`kahan(n)`	$[-2.84, 1.45] \times [-2.38, 2.38]$	231	92
`chebspec(n,0)`	$[-376.03, 376.03] \times [-265.18, 265.18]$	271	95
`chebspec(n,1)`	$[-400.66, 108.89] \times [-281.76, 281.76]$	200	64
`chow(n)`	$[-0.52, 17.49] \times [-10.11, 10.11]$	285	104
`triangle(n)`	$[-0.75, 1.24] \times [-1.09, 1.09]$	726	595
`propeller(n)`	$[-1.02, 1.48] \times [-1.35, 1.35]$	709	524
`fish(n)`	$[-0.27, 3.48] \times [-1.35, 1.35]$	832	624

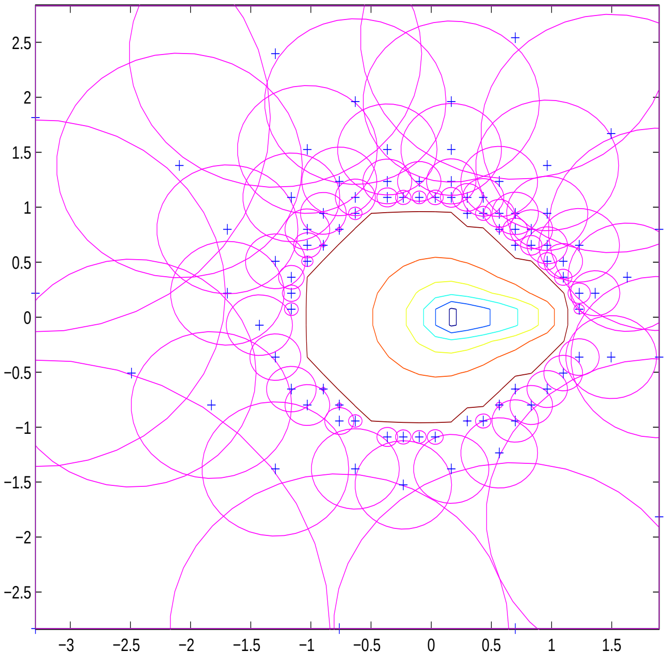
Table 4.2

Number of $\sigma_{\min}$ *evaluations in a* 50×50 *grid using the two versions of algorithm* `MoG` *for* $\epsilon = 1e - 3$ *and* $n = 32$.

Table 4.3 shows the effect of increasing gridsize (70×70 and 90×90) on some of the matrices used above.

We already saw earlier (Figure 4.2) that when the field of values criterion is applied to obtain a starting region for `kahan` of size $n = 32$ and $\epsilon = 1e - 1$ on a 40×40 grid, the initial domain was $[-3.3, 1.9] \times [-2.9, 2.9]$ and the total number of singular value evaluations was 372. How would the number of evaluations be if the initial region were selected to be much larger? We know that in the case of `GRID`, such a choice would increase significantly the cost. To see the effect on our algorithm, we applied it on the same matrix, using this time an initial region that was expanded by 10 units in each direction, i.e. $[-13.3, 11.9] \times [-12.9, 12.9]$. In order to obtain the same kind of resolution for the pseudospectrum, for the latter case we use the same grid with

FIG. 4.3. *Application of Corollary 2.13 using grid of size* 40×40 *for matrices* `kahan` *of size* $n = 32$ *and* $\epsilon = 1e - 1$. *The center of each disk is marked with '+'.*



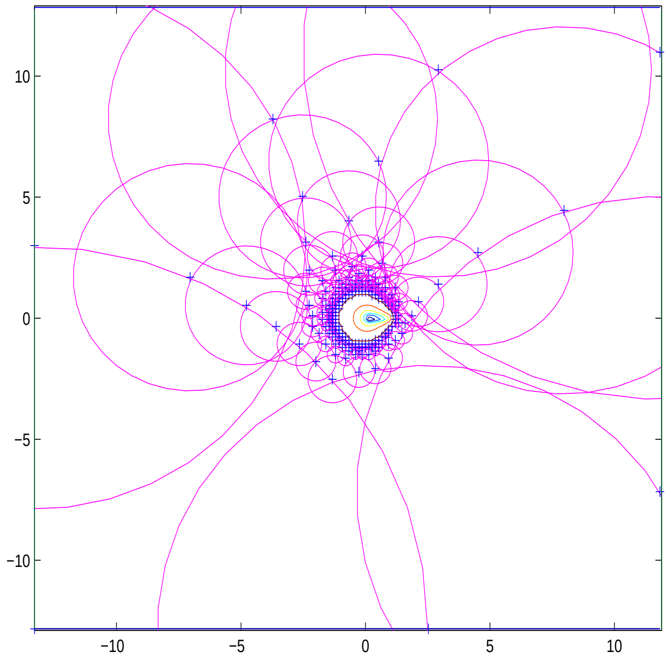
matrix	size n	region Ω	`MoGO`	`MoG`
`grcar`	70	$[-0.59, 2.95] \times [-3.09, 3.09]$	1216	625
	90		1881	881
`kahan`	70	$[-2.84, 1.45] \times [-2.38, 2.38]$	379	131
	90		558	189
`triangle`	70	$[-0.75, 1.24] \times [-1.09, 1.09]$	1377	1045
	90		2210	1615
`propeller`	70	$[-1.02, 1.48] \times [-1.35, 1.35]$	1327	906
	90		2095	1367

TABLE 4.3

Number of $\sigma_{\min}$ *evaluations for varying grid sizes using two versions of algorithm* `MoG` *and* $\epsilon = 1e - 3$.

the smaller initial region. This led to a grid of 190×177. The application of `MoGO` required 383 singular value evaluations, that is only 11 more than the use of the much tighter domain. On the other hand, `GRID` would have required $33,630 = 190 \times 177$ evaluations compared to $1,600$. The disks are shown in Figure 4.4. The large size of the disks that are far away from the pseudospectrum enable the fast pruning of the initial region. The above shows that the algorithm is not sensitive to the choice of the initial enclosing region, in the sense that even if the region is much larger than the largest sought pseudospectrum, the exclusion disks rapidly reduce the size of the domain. As remark 2 after Theorem 2.14 indicates, when $|z|$ is large relative to the elements of A, $zI - A$ is near diagonal with diagonal terms almost equal to z, therefore $\sigma_{\min}(zI - A) \approx |z|$, therefore the exclusion disk $D(z, \sigma_{\min}(zI - A) - \epsilon)$ would have

FIG. 4.4. *Application of Corollary 2.13 using grid of size* 190 × 177 *for matrix* `kahan` *of size* $n = 32$ *and* $\epsilon = 1e-1$ *in a large initial region. The center of each disk is marked with '+'.*



large radius and hence would exclude many of the redundant points of Ω.

5. Further Research. We have presented new algorithms for the computation of matrix pseudospectrum and the localization of matrix spectrum. Our results were encouraging and suggest that pseudospectrum localization is a powerful tool for the fast computation of pseudospectra. There are several issues that are currently under investigation.

The first question is how matrix-based methods can be integrated efficiently in our algorithms. It could be interesting to find if exclusion theorems can be transferred in the context of these methods. In general, we believe that our results could be proven very useful in the task of integrating different approaches in state-of-the-art software, as proposed in [18]. The second issue is the implementation of these algorithms, especially in a parallel environment. Our initial experiments, using ARPACK on an SGI Origin 2000 system with 8 processors, and running `MoG` in MIPS Fortran led to encouraging speedups; a state-of-the-art implementation is under development. Another line of research follows from the observation that our exclusion results can also help for spectrum localization. Furthermore, the fact that $\sigma_{\min}(zI - A)$ is 0 when z is an eigenvalue, can be used as a starting point for algorithms for computing pseudospectra [3] and eigenvalues [14].

Notes and acknowledgments. Part of this work was included in the Diploma Thesis of the first author while he was a student at the Department of Computer Engineering and Informatics of the University of Patras. We thank Mr. A. Sidiropoulos, who implemented the algorithm in `Java` and Mr. C. Bekas for helpful discussions. The authors would also like to thank the organizers of the celebration of Professor Richard Varga's 70th birthday at Kent State University as well as of the Foundations of Computational Mathematics Conference (FOCM) at Oxford University in 1999 for inviting them to present this work.

REFERENCES

[1] F. Bauer and C. Fike, *Norms and exclusion theorems*, Numer. Math., 2 (1960), pp. 137–141.

[2] C. Bekas and E. Gallopoulos, *Cobra: Parallel path following for computing the matrix pseudospectrum*, Parallel Computing, (to appear). Extended version available from the authors.

[3] K. Bekas and E. Gallopoulos, *Parallel computation of pseudospectra by fast descent*, tech. rep., Computer Engineering and Informatics Dept., University of Patras, November 2000. Presented at , Neuchatel, Switzerland. In URL `www.ceid.upatras.gr/scgroup/Reports`. Submitted for publication.

[4] T. Braconnier, R. McCoy, and V. Toumazou, *Using the field of values for pseudospectra generation*, Tech. Rep. TR/PA/97/28, CERFACS, Toulouse, Sept. 1997.

[5] M. Brühl, *A curve tracing algorithm for computing the pseudospectrum*, BIT, 33 (1996), pp. 441–445.

[6] L. Elsner and K.-D. Ikramov, *Normal matrices: an update*, Lin. Alg. Appl., 285 (1998), pp. 291–303.

[7] J. g. Sun, *A note on simple non-zero singular values*, J. Comput. Math., 6 (1988), pp. 258–266.

[8] ———, *Sensitivity analysis of zero singular values and multiple singular values*, J. Comput. Math., 6 (1988), pp. 325–335.

[9] E. Gallestey, *Computing spectral value sets using the subharmonicity of the norm of rational matrices*, BIT, 38 (1998), pp. 22–33.

[10] O. Hald, *A converse to the Bauer-Fike theorem*, Lin. Alg. Appl., 9 (1974), pp. 267–274.

[11] P. Henrici, *Bounds for iterates, inverses, spectral variation and field of values of nonnormal matrices*, Numer. Math., 4 (1962), pp. 24–40.

[12] N. Higham, *The Test Matrix Toolbox for MATLAB (version 3.0)*, Tech. Rep. 276, Manchester Centre for Computational Mathematics, Sept. 1995.

[13] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.

[14] I. Koutis and E. Gallopoulos, *Hermitian methods for computing eigenvalues.* In preparation, 2000.

[15] R. Lehoucq, D. Sorensen, and C. Yang, *Arpack User's Guide: Solution of Large-Scale Eigenvalue Problems With Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, 1998.

[16] S. Lui, *Computation of pseudospectra with continuation*, SIAM J. Sci. Comput., 18 (1997), pp. 565–573.

[17] G. Stewart and J. g. Sun, *Matrix Perturbation Theory*, Academic Press, Boston, 1990.

[18] L. Trefethen, *Computation of pseudospectra*, in Acta Numerica 1999, vol. 8, Cambridge University Press, 1999, pp. 247–295.