PAT - a Reliable Path Following Algorithm [*]

Dany Mezher [a] Bernard Philippe [b]

[a] *ESIB-USJ, Campus des Sciences et Technologies, Beirut, Lebanon.*
E-mail: dany.mezher@fi.usj.ed.lb
[b] *IRISA-INRIA, Campus de Beaulieu, 35042 Rennes Cedex, France.*
E-mail: bernard.philippe@irisa.fr.

This paper presents a new technique for the reliable computation of the $\sigma-$pseudospectrum defined by $\Lambda_\sigma(A) = \{z \in \mathbb{C} : \sigma_{\min}(A - zI) \leq \sigma\}$ where $\sigma_{\min}$ is the smallest singular value. The proposed algorithm builds an orbit of adjacent equilateral triangles to capture the level curve $\Upsilon_\sigma(A) = \{z \in \mathbb{C} : \sigma_{\min}(A - zI) = \sigma\}$ and uses a bisection procedure on specific triangle vertices to compute a numerical approximation to $\Upsilon_\sigma(A)$. The method is guaranteed to terminate, even in the presence of round-off errors.

Keywords: Path following, pseudospectrum, simplicial, orbit, bisection, smallest singular value.

AMS Subject classification: 65H20, 58C07, 65G50, 65H17

1. Introduction

The $\sigma-$pseudospectrum [16] of the matrix $A \in \mathbb{C}^{n\times n}$ is the region of the complex plane defined by

$$\Lambda_\sigma(A) = \{z \in \mathbb{C} : z \text{ is an eigenvalue of } A + E \ \text{ where } ||E||_2 \leq \sigma\}.$$

The pseudospectrum reveals the sensitivity of the eigenvalues of the matrix with respect to perturbations of norm smaller than or equal to σ. The following equivalent definition provides an effective criterion to determine if a given complex z belongs to $\Lambda_\sigma(A)$:

$$\Lambda_\sigma(A) = \{z \in \mathbb{C} : \sigma_{\min}(A - zI) \leq \sigma\},$$

[*] This work was done during a visit of the first author to IRISA with the financial support of EGIDE.

where $\sigma_{\min}$ denotes the smallest singular value. The numerical evaluation of the smallest singular value is a computation of high cost; it can be performed with different algorithms depending on the matrix structure [5,10,12,14].

To compute efficiently $\Lambda_\sigma(A)$, it is now commonly accepted [15] that the path following techniques that follow the curve

$$\Upsilon_\sigma(A) = \{z \in \mathbb{C} : \sigma_{\min}(A - zI) = \sigma\}$$

are of much smaller complexity than methods based on a grid discretization of the complex region. The first attempt in this direction was done by Brühl [4]. Since it involves a continuation with a predictor corrector scheme [1,9,7,13], the process may fail in the case of singular points. In [2], Bekas and Gallopoulos develop a hybrid algorithm that uses a continuation scheme coupled with a fine local grid to compute the pseudospectrum of a matrix. In [6], Huitfieldt and Ruhe developed a continuation method for following the solution of a non linear eigenvalue problem. With a Euler-Newton procedure, they predict singular points along the level curve using an augmented system. In the survey [1], Allgower and Georg present the piecewise-linear methods for the curve tracing of non smooth functions $H : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$; the *PAT* algorithm which is described in this paper can be considered as a specialization of the piecewise-linear methods for the pseudospectrum problem. This particular case where $n = 1$ enables us to prove additional properties regarding the reliability, stability and termination of the process.

The algorithm presented in this paper offers enhanced reliability for the computation of $\Upsilon_\sigma(A)$ at a low cost. Furthermore, it guarantees termination even in the presence of round-off errors. The main idea of the proposed algorithm is to line up a set of equilateral triangles along the level curve and to use a bisection algorithm [17] over the triangle vertices to compute $\Upsilon_\sigma(A)$.

The paper is organized as follows. The mathematical foundations are given in section 2; the algorithm to compute a single component $\Upsilon_\sigma^{(i)}(A)$ is presented in section 3, while its complexity is estimated in section 3.1. Section 3.2 provides a backward error analysis and a guarantee of termination. Section 4 discusses a technique used to compute all connected components of the level curve. In section 5, numerical tests are described demonstrating the reliability and efficiency of the proposed algorithm.

2. Mathematical foundations

We present in this section the mathematical background of the algorithm used to capture the pattern of a single connected component of the level curve $\Upsilon_\sigma^{(i)}(A)$. We start by defining a lattice of uniformly distributed nodes and the corresponding triangular mesh, and then we consider a subset $\mathcal{T}_L$ of the mesh triangles which contains connected components of $\Upsilon_\sigma(A)$; we prove that $\mathcal{T}_L$ is a finite set and we construct a transformation to generate subsets of $\mathcal{T}_L$ from an initial triangle $T \in \mathcal{T}_L$. The set $\mathcal{T}_L$ will provide the edges from which some points of $\Upsilon_\sigma(A)$ (one by edge) are extracted using a bisection.

Definition 2.1. For any $\sigma > 0$, the *interior* of $\Upsilon_\sigma(A)$ is the set

$$\Delta_\sigma(A) = \{z : \phi(z) < \sigma\}.$$

where $\phi(z) = \sigma_{\min}(A - zI)$. The closure $\Lambda_\sigma(A)$ of $\Delta_\sigma(A)$ is bounded and the *exterior* of $\Upsilon_\sigma(A)$ defined by

$$\Gamma_\sigma(A) = \{z : \phi(z) > \sigma\}$$

is an unbounded open set.

The fact that $\lim_{|z|\to\infty} \phi(z) = +\infty$ implies that for every $\alpha > 0$, there exists $R_\alpha \in \mathbb{R}^+$ such that $\phi(z) > \alpha$ whenever $|z| > R_\alpha$. The special case of $\alpha = \sigma$ leads to

$$\Lambda_\sigma(A) \subset \overline{B}(0, R_\sigma)$$

where $\overline{B}(0, R_\sigma)$ is the closed disk with center at 0 and radius R_σ.

Definition 2.2. Two distinct points a and b are said to be σ-separated when, by definition :

$$\{a, b\} \cap \Lambda_\sigma(A) \neq \emptyset \quad \text{and} \quad \{a, b\} \cap \Gamma_\sigma(A) \neq \emptyset.$$

In this case, the segment $[a, b]$ is said to be σ-separated.
Given $(z_i, z_e) \in \mathbb{C}^2$ such that $z_i \neq z_e$,

$$S(z_i, z_e) = \{S_{kl} = z_i + k(z_e - z_i) + l(z_e - z_i)e^{i\frac{\pi}{3}}, (k, l) \in \mathbb{Z}^2\}$$

defines a uniform lattice (see fig 1) of nodes satisfying

$$|S_{k,l+1} - S_{k,l}| = |S_{k+1,l} - S_{k,l}| = |z_i - z_e|.$$

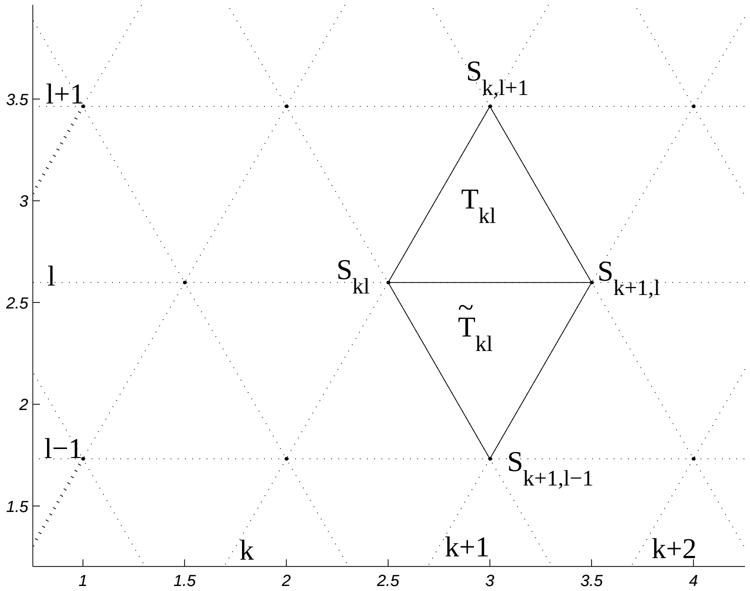


Figure 1. The lattice for $z_i = 0$ and $z_e = 1$

The obtained triangular mesh is the union of two classes of equilateral triangles :

$$\Omega(z_i, z_e) = \Psi(z_i, z_e) \cup \tilde{\Psi}(z_i, z_e)$$

where

$$\Psi = \{T_{kl} = \{S_{k,l}, S_{k+1,l}, S_{k,l+1}\} : (k, l) \in \mathbb{Z}^2\}$$

and

$$\tilde{\Psi} = \{\tilde{T}_{kl} = \{S_{k,l}, S_{k+1,l}, S_{k+1,l-1}\} : (k, l) \in \mathbb{Z}^2\}.$$

In the following, we denote by $\mathcal{T}_L$ the subset of $\Omega(z_i, z_e)$ where $T \in \mathcal{T}_L$ if, and only if, T has at least two σ-separated vertices.

Proposition 2.1. For all $z_i \neq z_e$, $\mathcal{T}_L$ is a finite set.

Proof. For any triangle $T \in \mathcal{T}_L$, there exists a vertex $S_{ij} \in \Lambda_\sigma(A)$. Since $\Lambda_\sigma(A)$ is bounded then $S(z_i, z_e) \cap \Lambda_\sigma(A)$ is a finite set and $card(\mathcal{T}_L) \leq 6 \times card(S(z_i, z_e) \cap \Lambda_\sigma(A))$. □

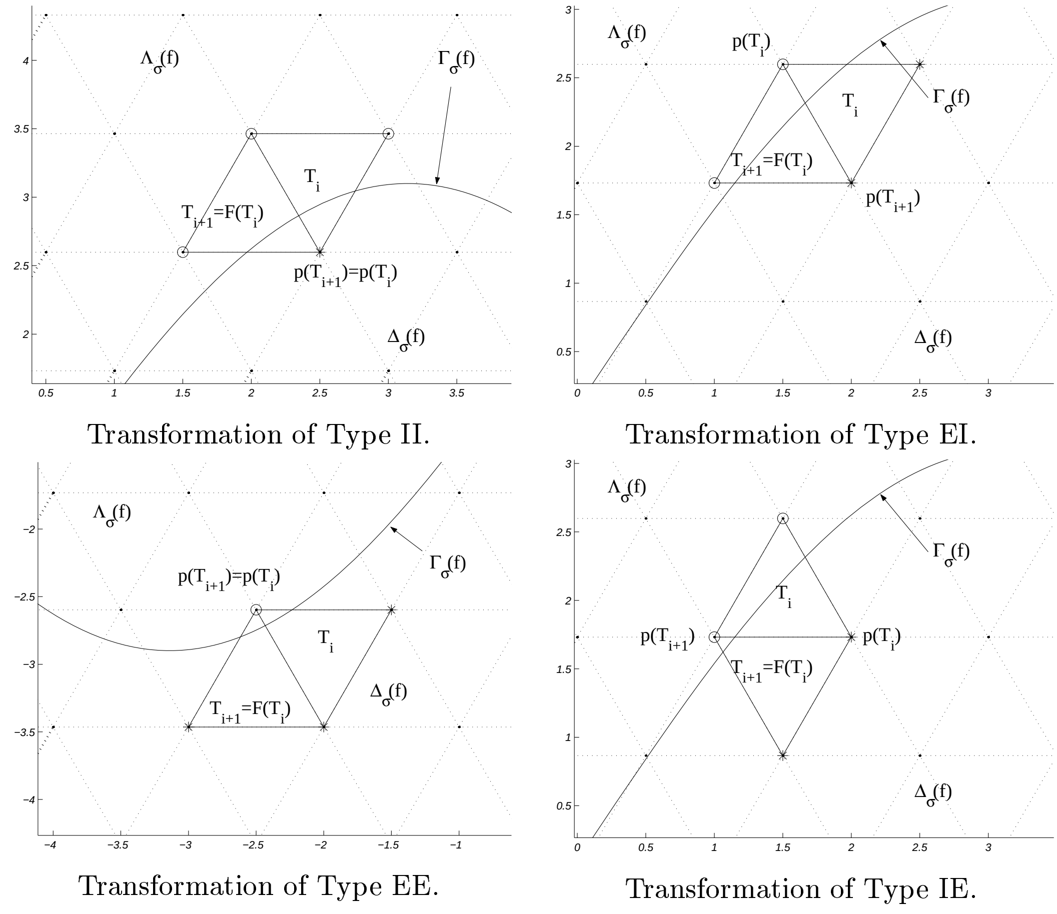
We can easily show that if T is an element of $\mathcal{T}_L$ then T has two, and only two, σ-separated edges. Therefore, we may define the pivot of each element of $\mathcal{T}_L$:

Definition 2.3. The common vertex to the σ-separated sides is called the pivot of T and denoted $p(T)$.

Definition 2.4. The transformation F (see figure 2) is defined by

$$F(T) = R(p(T), sgn(\sigma - \phi(p(T))) \times \frac{\pi}{3})(T)$$

where $T \in \mathcal{T}_L$, $sgn(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$ and $R(z, \theta)$ is the rotation centered at $z \in \mathbb{C}$ with angle θ. F maps every triangle T of $\mathcal{T}_L$ into a triangle $F(T)$.



Transformation of Type II.

Transformation of Type EI.

Transformation of Type EE.

Transformation of Type IE.

Figure 2. The transformation F

Figure 2 exhibits the four cases of the F transformation, while table (1) presents the corresponding rotation angles. The four situations are identified by two letters which specify the location of $p(T)$ and $p(F(T))$ with respect to the curve (I : interior , E : exterior).

Transformation Type	T Type	$F(T)$ Type	$p(T)$	$p(F(T))$	θ
II	I	I	$\in \Lambda_\sigma(A)$	$\in \Lambda_\sigma(A)$	$\frac{\pi}{3}$
EI	E	I	$\in \Gamma_\sigma(A)$	$\in \Lambda_\sigma(A)$	$-\frac{\pi}{3}$
EE	E	E	$\in \Gamma_\sigma(A)$	$\in \Gamma_\sigma(A)$	$-\frac{\pi}{3}$
IE	I	E	$\in \Lambda_\sigma(A)$	$\in \Gamma_\sigma(A)$	$\frac{\pi}{3}$

Table 1

The four possible situations for T and $F(T)$

Proposition 2.2. For any element $T \in \mathcal{T}_L$, we claim that

1. $F(T) \neq T$,
2. $p(T)$ is a vertex of $F(T)$,
3. T and $F(T)$ are adjacent,
4. The common edge to T and $F(T)$ is σ-separated, therefore $F(T) \in \mathcal{T}_L$,
5. $p(F(T))$ is a vertex of T,
6. $F^2(T) \neq T$,
7. if $T \in \Psi(z_i, z_e)$ then $F(T) \in \tilde{\Psi}(z_i, z_e)$ and if $T \in \tilde{\Psi}(z_i, z_e)$ then $F(T) \in \Psi(z_i, z_e)$.

Proof.

1. Follows from the definition of F.
2. $p(T)$ is the center of the rotation, hence invariant by F.
3. Since $F(T)$ is a rotation centered at a vertex of T with angle $\pm\frac{\pi}{3}$, and since T is equilateral T and $F(T)$ are adjacent.
4. $p(T)$ is the common vertex to the $\sigma-$separated sides of T and $p(T) \in F(T)$; the fact that T and $F(T)$ are adjacent leads to the result.

5. Follows from the definition of $p(F(T))$ and from 4.

6. We know that T and $F(T)$ are adjacent; furthermore $p(T) \in T \cap F(T)$ and $p(F(T)) \in T \cap F(T)$. There are two distinct cases :

   either $p(T) = p(F(T)) \rightsquigarrow F^2(T) = R(p(T), \pm\frac{2\pi}{3})(T) \neq T$

   or $p(T) \neq p(F(T))$; in this case

   $$\begin{aligned} F^2(T) &= R(p(F(T)), -\theta) \circ R(p(T), \theta)(T) \\ &= \Xi[(p(F(T)) - p(T))(1 - e^{-i\theta})](T) \\ &\neq T \end{aligned}$$

   where $\Xi[u](x) = x + u$ is the translation of vector u.

7. Obvious.

□

Proposition 2.3. F is a bijection from $\mathcal{T}_L$ onto $\mathcal{T}_L$.

Proof. We prove that F is a one-to-one mapping. Let us assume that there exist two triangles T and T' such that

$$F(T) = F(T') \text{ and } T \neq T'.$$

We know from proposition 2.2 that $p(F(T)) \in T$, $p(F(T)) \in F(T)$, $p(F(T)) \in F^2(T)$ and $p(F(T)) \in T'$ therefore

$$p(F(T)) \in T \cap F(T) \cap F^2(T) \cap T'$$

Furthermore, we know that T, $F^2(T)$ and T' are adjacent to $F(T)$. This can only be true if $T' = F^2(T)$; therefore,

$$\begin{aligned} F(T') &= F(F^2(T)) \\ &= F^2(F(T)) \end{aligned}$$

and since $F^2(T) \neq T, \forall T \in \mathcal{T}_L$, then $F(T') \neq F(T)$. This proves that $F : \mathcal{T}_L \to \mathcal{T}_L$ is a one-to-one mapping and therefore a bijection onto the finite set $\mathcal{T}_L$. □

Definition 2.5. For any given $T \in \mathcal{T}_L$ we define the $F-$orbit of T to be the set $O(T) = \{T_n \equiv F^n(T), n \in \mathbb{Z}\}$.

Proposition 2.4. Let $O(T)$ be the $F-$orbit for a given triangle T;

1. $O(T)$ is a finite set.
2. If $n = card(O(T))$ then n is even and is the smallest positive integer such that $T_n = T$.
3. $\sum_{i=0}^{n-1} \theta_i = 0 \quad (2\pi)$ where θ_i is the rotation angle of F for triangle T_i.
4. If $O(T')$ is the $F-$orbit for a triangle T' then $O(T) = O(T')$ or $O(T) \cap O(T') = \emptyset$.

Proof.

1. $O(T) \subset \mathcal{T}_L$ and $\mathcal{T}_L$ is a finite set.
2. Since $T_n \in O(T)$, there exists an integer j such that, $0 \leq j < n$ and $T_n = T_j$. Therefore,
$$T_{n-1} = T_{j-1};$$
this is only true if $j = 0$, otherwise $card(O(T)) < n$.
Since T and $F(T)$ belong to the two disjoint classes $\Psi(z_i, z_e)$ and $\tilde{\Psi}(z_i, z_e)$, the integer n is even.
3. Let $O(T_0) = \{T_0, T_1, ..., T_{n-1}\}$ be the $F-$orbit of T_0; in the following, $\{P_i, S_i, U_i\}$ denotes the triangle T_i such that P_i is the pivot of T_i and $(\overrightarrow{P_iS_i}, \overrightarrow{P_iU_i}) = \theta_i \quad (2\pi)$ such that $F(T_i) = R(P_i, \theta_i)(T_i)$. Therefore, we can state
$$T_i \cap T_{i+1} = [P_i, U_i] = [P_{i+1}, S_{i+1}].$$
We have now two cases. The first case is when T_i and T_{i+1} are of the same type (I or E, see Table 1) and therefore $P_{i+1} = P_i$ and $S_{i+1} = U_i$ leading to
$$\overrightarrow{P_{i+1}S_{i+1}}=\overrightarrow{P_iU_i} \ .$$
The second case is when T_i and T_{i+1} are of different type, and therefore $P_{i+1} = U_i$ and $S_{i+1} = P_i$ and
$$\overrightarrow{P_{i+1}S_{i+1}}= - \overrightarrow{P_iU_i} \ . \tag{1}$$
Let us consider the sum
$$\begin{aligned} \sum_{i=0}^{n-1} \theta_i &= \sum_{i=0}^{n-1} (P_iS_i, \overrightarrow{P_iU_i}) \quad (2\pi) \\ &= (\overrightarrow{P_0S_0}, \overrightarrow{P_0S_0}) + k\pi \quad (2\pi) \end{aligned}$$

where k is the number of triangles satisfying (1). Since $F_n(T_0) = T_0$, we can claim that k is even; finally

$$\sum_{i=0}^{n-1} \theta_i = 0 \ \ (2\pi).$$

4. Let T" be an element of $O(T) \cap O(T')$; there exist two integers i, j such that

$$T\text{"} = F_i(T) = F_j(T') = F_{n+j}(T') \Rightarrow T = F_{j+n-i}(T')$$

where $n = card(O(T'))$, leading to $O(T) \subset O(T')$. In the same way we can prove that $O(T') \subset O(T)$.

□

```
**(a)** Start :

Given $\sigma$,$\tau$ and $\tilde{z}_0$ such that $\sigma_{\min}(A - \tilde{z}_0 I) \leq \sigma$
**for** k=1,2,...
    $\tilde{z}_k = \tilde{z}_0 + 2^{k-1}\tau e^{i\theta}$
    **if** $\sigma_{\min}(A - \tilde{z}_k I) > \sigma$ **break**
**end for**
Use a bisection algorithm over $\tilde{z}_0$ and $\tilde{z}_k$,
    to compute $z_i$ and $z_e$ where
    $\sigma_{\min}(A - z_i I) \leq \sigma < \sigma_{\min}(A - z_e I)$ and $|z_i - z_e| = \tau$
Let $T_0 = \{z_i, z_e, z_i + (z_e - z_i)e^{\frac{i\pi}{3}}\} \in \mathcal{T}_L$ be the initial triangle

**(b)** Build the $F-$orbit of $T_0$ :

**for** $i = 0, 1, 2, ...$
    $T_{i+1} = F(T_i)$
    **if** $T_{i+1} = T_0$ **break**
**end for**
```

Figure 3. Path Following algorithm

3. The path following algorithm

In this section, we analyze the implementation of the path following algorithm presented in figure 3. The goal of this algorithm is twofold:

1. To find a suitable lattice and the corresponding $F-$orbit O to capture the pattern of a connected component of the level curve. We start by determining two points $z_i \in \Lambda_\sigma(A)$ and $z_e \in \Gamma_\sigma(A)$ such that $|z_i - z_e| = \tau$ where τ defines the resolution of the lattice. The two points z_i and z_e are used to generate the lattice $S(z_i, z_e)$ where $T_0 = \{S_{00}, S_{10}, S_{01}\}$ is an element of $\mathcal{T}_L$ which generates the $F-$orbit.
   In our implementation, the technique used to compute z_i and z_e is as follows :

   - Given $\tilde{z}_0$ such that $\sigma_{\min}(A - \tilde{z}_0 I) \leq \sigma$, consider the complex sequence defined by
     $$\tilde{z}_{k+1} = \tilde{z}_0 + 2^k \tau e^{i\theta}$$
     where θ is a random angle; in this case,
     $$\lim_{k \to +\infty} \sigma_{\min}(A - \tilde{z}_k I) = +\infty.$$
     Let $k_0 \in \mathbb{N}^*$ be a value of k such that $\sigma_{\min}(A - \tilde{z}_{k_0} I) > \sigma$
   - use a bisection algorithm, starting from $\tilde{z}_0$ and $\tilde{z}_{k_0}$, to compute the two points z_i and z_e.

   The $F-$orbit is built by successively applying F from the initial triangle T_0.

2. To find an approximation to $\Upsilon_\sigma^{(i)}(A)$; this is achieved by performing, over each element of O, a bisection algorithm starting from the σ-separated vertices. Bisection algorithm, presented in figure 4, was chosen for its reliability.

3.1. Complexity

Let $O(T)$ be the $F-$orbit of T built to compute a numerical approximation of a connected component of a level curve $\Upsilon_\sigma(A)$; we limit our study to the case[1] where the cardinal of the orbit satisfies $|O(T)| > 6$.

[1] It is obvious that the smallest orbits have 6 elements.

```
Let $x$, $y$ be two complex values such that $\sigma_{\min}(A - xI) \leq \sigma$ and $\sigma_{\min}(A - yI) > \sigma$
while $|x - y| > \rho|x|$
    $m = \frac{x+y}{2}$
    if $\sigma_{\min}(A - mI) > \sigma$ then
      $y = m$
    else
      $x = m$
    end if
end while
```

Figure 4. The bisection algorithm

Proposition 3.1. The number n of triangles needed to capture the pattern of a level curve of length l satisfies

$$\frac{l}{\tau} \leq n \leq \frac{10}{\sqrt{3}} \left(\frac{l}{\tau}\right).$$

Proof. For any given element $U \in O(T)$, there exists an integer $0 \leq k < 5$ such that $p(F^k(U)) \neq p(F^{k+1}(U))$, otherwise $F^6(U) = U$ and $card(O(T)) = 6$.

In the following, we denote by $V = F^k(U)$, $W = F(V)$ and $[x, y] = V \cap W$. We may assume that $x = p(V)$ and $y = p(W)$. Let v be the third vertex of V and w the third vertex of W; the two segments $[x, v]$ and $[y, w]$ are σ-separated and parallel. If $z^{(v)} \in [x, v] \cap \tilde{\Upsilon}_\sigma(A)$ and $z^{(w)} \in [y, w] \cap \tilde{\Upsilon}_\sigma(A)$ then

$$|z^{(v)} - z^{(w)}| \geq d([x, u], [y, v]) = \frac{\tau\sqrt{3}}{2}.$$

Therefore, we are guaranteed to progress by a distance not smaller than $\frac{\tau\sqrt{3}}{2}$ for five consecutive triangles. □

3.2. Round-off errors and termination

As shown in proposition 2.4, the process terminates when $F(T_k) = T_0$. To guarantee the process termination even in the presence of round-off errors, we use integer coordinates to identify the lattice nodes. Therefore, the node S_{kl} is identified by the two integers k and l and the evaluation of $\sigma_{\min}(A-S_{kl}I)$ requires the evaluation of

$$\sigma_{\min}(A-(z_i+(z_e-z_i)(k+le^{\frac{i\pi}{3}}))I).$$

3.2.1. Backward error analysis

Let us now consider the effect of roundoff errors on the evaluation of $\sigma_{\min}(A-S_{kl}I)$ where S_{kl} is a mesh node. It might be possible that in some cases the computed value $fl(\sigma_{\min}(A-S_{kl}I))$ and the exact value $\sigma_{\min}(A-S_{kl}I)$ are $\sigma-$seperated and therefore, the constructed orbit differs from the exact one. We shall prove that the computed orbit is the exact orbit of a function ψ approximating $\phi(z)=\sigma_{\min}(A-zI)$ within a precision defined by the computational precision of the algorithm used to compute $\sigma_{\min}$.

In the following, we define the *forward error* e_{kl} of the evaluation of $\phi(S_{kl})$ by

$$fl(\phi(S_{kl}))=\phi(S_{kl})+e_{kl}.$$

Furthermore, we assume that the evaluation of ϕ is performed by a reliable and stable computation which means that:

$$|e_{kl}|=O(\epsilon\eta_{kl}),$$

where ϵ is the precision parameter used in the singular value computation, and $\eta_{S_{kl}}=\max(\epsilon,\phi(S_{kl}))$. This implies that

$$\max_{z\in S(z_i,z_e)}\frac{|e_z|}{\eta_z}=O(\epsilon)$$

Theorem 3.1. For any given $\sigma-$level curve and any given acceptable[2] triangle T, we consider the numerically computed $F-$orbit $\tilde{O}(T)$ computed using a floating point arithmetic with a given singular value solver working at precision ϵ. The $F-$orbit $\tilde{O}(T)$ may be considered as the exact $F-$orbit for a function ψ satisfying

$$\left|\left|\frac{\phi-\psi}{\eta}\right|\right|_\infty=O(\epsilon)$$

[2] An acceptable triangle is any triangle such that $Sgn(fl(\sigma_{\min}(A-zI)-\epsilon))=Sgn(\sigma_{\min}(A-zI)-\epsilon)$ when z is any of the three vertices.

where

$$\eta(z) = \begin{cases} \eta_z \text{ if } z \text{ is a mesh node,} \\ \max(\eta_u, \eta_v) \text{ if } z \in (u,v) \text{ edge of the lattice,} \\ \max(\eta_u, \eta_v, \eta_w) \text{ if } z \text{ belongs to a triangle } \{u,v,w\} \text{ of the mesh.} \end{cases}$$

Proof. Since the forward error is defined at each vertex of the mesh, we can linearly interpolate it on each mesh triangle based on the values at the vertices: for any z belonging to the triangle $\{u,v,w\}$, we consider (α,β,γ) the barycentric coordinates of z where $\alpha \geq 0$, $\beta \geq 0$, $\gamma \geq 0$ and $\alpha+\beta+\gamma = 1$; in this case, $z = \alpha u + \beta v + \gamma w$. Let us define $e(z) = \alpha e_u + \beta e_v + \gamma e_w$. The function ψ defined by $\psi(z) = \sigma_{\min}(A - zI) + e(z)$ interpolates the computed evaluations of ϕ over the mesh and

$$\begin{aligned} \frac{|e(z)|}{\eta(z)} &\leq \alpha \frac{|e_u|}{\eta(z)} + \beta \frac{|e_v|}{\eta(z)} + \gamma \frac{|e_w|}{\eta(z)} \\ &\leq \alpha \frac{|e_u|}{\eta_u} + \beta \frac{|e_v|}{\eta_v} + \gamma \frac{|e_w|}{\eta_w} \\ &\leq O(\epsilon) \end{aligned}$$

□

When the orbit $O(T)$ is built, the bisection process is run to obtain a polygon with vertices $\{z^{(i)}\}_{i=0,n-1}$ such that $|z^{(i+1)} - z^{(i)}| \leq \tau$. In the stopping criterion $|x-y| \leq \rho|x|$, ρ is chosen an order of magnitude larger than ϵ and smaller than τ

$$\epsilon \ll \rho \ll \tau.$$

4. Computing disjoint connected components of $\Upsilon_\sigma(A)$

Given two sets of points $\mathcal{I} \subset \Lambda_\sigma(A)$ and $\mathcal{E} \subset \Gamma_\sigma(A)$, we develop in this section a reliable technique used to compute multiple connected components surrounding all the points in $\mathcal{I}$. We limit our study to the case where $\mathcal{E} \neq \emptyset$ since one can easily compute external points to append to $\mathcal{E}$. The points in $\mathcal{I}$ can be user defined or numerical approximations of the eigenvalues of A.

Definition 4.1. For any orbit $O(T)$, the internal vertices of the triangles in $O(T)$ define the *interior polygon* denoted $l_i(O(T))$, whereas the exterior vertices define the *exterior polygon* denoted $l_e(O(T))$.

Proposition 4.1. $l_i(O(T))$ and $l_e(O(T))$ are disjoint.

Proof. Since $l_i(O(T))$ and $l_e(O(T))$ have adjacent mesh nodes as adjacent vertices, therefore the intersection of $l_i(O(T))$ and $l_e(O(T))$ will include a mesh node x satisfying

$$\sigma_{\min}(A - xI) \leq \sigma \quad \text{and} \quad \sigma_{\min}(A - xI) > \sigma.$$

□

Definition 4.2. An orbit $O(T)$ is said to be direct if $l_e(O(T))$ encloses $l_i(O(T))$; otherwise, the orbit is said to be reversed.

To decide if a given $z \in l_i(O(T))$ is surrounded by $l_e(O(T))$, we consider the criterion

$$(z \in l_i(O(T)) \text{ is surrounded by } l_e(O(T))) \Leftrightarrow Ind_{l_e(O(T))}(z) \neq 0$$

where

$$Ind_{l_e(O(T))}(z) = \frac{1}{2i\pi} \int_{l_e(O(T))} \frac{du}{u - z}$$

We know that $Ind_{l_e(O(T))}(z)$ is an integer-valued function [11]. Numerically, $Ind_{l_e(O(T))}(z)$ can be evaluated by the sum (for small enough τ)

$$Ind_{l_e(O(T))}(z) = \frac{1}{2\pi} \sum_{u^{(i)} \in l_e(O(T))} \arg \frac{u^{(i+1)} - z}{u^{(i)} - z} \tag{2}$$

where $u^{(i)}$ and $u^{(i+1)}$ are two consecutive points of $l_e(O(T))$.

Theorem 4.1. A reversed orbit $O(T)$ is enclosed within a direct orbit $O(T')$.

Proof. Follows from the fact that $\Lambda_\sigma(A)$ is bounded. □

Let z_i and z_e be the two points used to generate a triangulation where $\sigma_{\min}(A - z_iI) \leq \sigma < \sigma_{\min}(A - z_eI)$. We limit our study to the case where each point in $\mathcal{I}$ is included in a mesh triangle having a vertex in $\Lambda_\sigma(A)$ and each point in $\mathcal{E}$ is included in a mesh triangle having a vertex in $\Gamma_\sigma(A)$ (if this is not the case, a finer mesh should be considered). In the following, every point in $\mathcal{I}$ and $\mathcal{E}$ is replaced by the vertex of the corresponding triangle having the same location with respect to the curve (interior or exterior).

Theorem 4.2. If $S_{k_il_i} \in \mathcal{I}$ and $S_{k_el_e} \in \mathcal{E}$ are two mesh nodes, then we can find a triangle T_α such that

1. $T_\alpha \in \mathcal{T}_L$,
2. T_α has a vertex x satisfying:
   (a) $s(x) \leq \sigma$,
   (b) $||x - S_{k_el_e}|| \leq ||S_{k_el_e} - S_{k_il_i}||$.
3. T_α has a vertex y satisfying:
   (a) $s(y) > \sigma$,
   (b) $||y - S_{k_il_i}|| \leq ||S_{k_el_e} - S_{k_il_i}||$.

Furthermore, if $S_{k_il_i}$ and $S_{k_el_e}$ are not adjacent mesh nodes, we can state that, at least one of the inequalities 2b and 3b is strict.

Proof. Let $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ be three mesh nodes (see Figure 5) such that:

1. $\mathcal{A} = S_{k_il_i}$,
2. the triangle $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ is equilateral,
3. $(\overrightarrow{\mathcal{A}\mathcal{B}}, \overrightarrow{z_iz_e}) = k\frac{\pi}{3}$ where $k \in \mathbb{Z}$,
4. $S_{k_el_e}$ is along the edge $\mathcal{BC}$.

Let J be the midpoint of the edge $\mathcal{BC}$, we have now two cases. The first case is when $S_{k_el_e} \in [\mathcal{B}J]$, and therefore we define $S_{\alpha\beta}$ to be the mesh node along the edge $\mathcal{AB}$ such that the triangle $(S_{\alpha\beta}, \mathcal{B}, S_{k_el_e})$ is equilateral. The second case is when $S_{k_el_e} \in]J\mathcal{C}]$, and therefore $S_{\alpha\beta}$ will be the mesh node along $\mathcal{AC}$ such that $(S_{\alpha\beta}, \mathcal{C}, S_{k_el_e})$ is equilateral. We can easily show that $[S_{k_il_i}, S_{k_el_e}]$ is the tallest edge of the triangle $(S_{k_il_i}, S_{\alpha\beta}, S_{k_el_e})$ and that we can find two adjacent mesh nodes x and y, along one of the following edges $[S_{k_il_i}, S_{\alpha\beta}]$ or $[S_{\alpha\beta}, S_{k_el_e}]$, such that

$$\sigma_{\min}(A - xI) \leq \sigma < \sigma_{\min}(A - yI),$$

$$||x - S_{k_el_e}|| \leq ||S_{k_il_i} - S_{k_el_e}||$$

and

$$||y - S_{k_il_i}|| \leq ||S_{k_il_i} - S_{k_el_e}||.$$

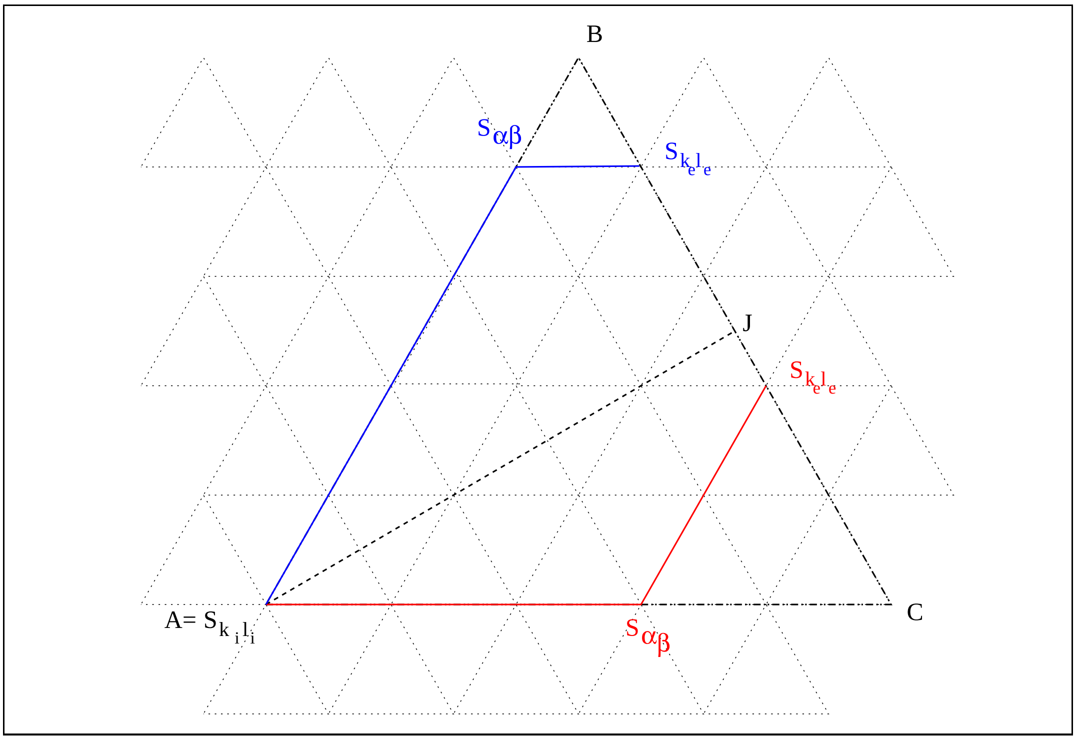


Figure 5. Building multiple connected components.

Finally the triangle T_α is any of the two equilateral triangles having the vertices x and y. □

The technique used to enclose all the points in $\mathcal{I}$ is a four steps procedure:

1. Determine $z_{\mathcal{I}} \in \mathcal{I}$ and $z_{\mathcal{E}} \in \mathcal{E}$ that minimize the distance $d(x,y) = ||x - y||$.
2. Find triangle T_α and compute the orbit $O(T_\alpha)$.
3. $\mathcal{I} = \mathcal{I} \cup l_i(O(T_\alpha))$ and $\mathcal{E} = \mathcal{E} \cup l_e(O(T_\alpha))$
4. $$\mathcal{I} = \mathcal{I} - \{z \in \mathcal{I} : Ind_{l_e(O(T_\alpha))}(z) \neq 0\}$$

   and

   $$\mathcal{E} = \mathcal{E} - \{z \in \mathcal{E} : Ind_{l_i(O(T_\alpha))}(z) \neq 0\}.$$

Notice that after steps 3 and 4, the vertices of $l_i(O(T))$ are appended to $\mathcal{I}$ if $O(T_\alpha)$ is reversed whereas the vertices of $l_e(O(T))$ are appended to $\mathcal{E}$ when $O(T_\alpha)$ is direct.

The procedure described earlier is repeated until $\mathcal{I} = \emptyset$ or $\mathcal{E} = \emptyset$. In the later case, we compute new external points that are not enclosed by any of the computed orbits, append them to $\mathcal{E}$ and restart the procedure.

```
While $\mathcal{I} \neq \emptyset$
  If $\mathcal{E} = \emptyset$
    Find $z \in \mathcal{I}$ with the largest magnitude.
    Build the sequence $z_k = z + 2^k \tau z/|z|$ until $s(z_k) \leq \sigma$.
    $\mathcal{E} = \{z_k\}$.
  End If
  Find $z_i \in \mathcal{I}$ and $z_e \in \mathcal{E}$ that minimize $|z_e - z_i|$.
  Compute $T_\alpha$.
  Compute $O(T_\alpha)$.
  $\mathcal{I} = \mathcal{I} \cup l_i(O(T_\alpha))$ and $\mathcal{E} = \mathcal{E} \cup l_e(O(T_\alpha))$.
  Delete all enclosed points of $\mathcal{I}$ and $\mathcal{E}$.
End While
```

Figure 6. Compute multiple connected components.

The procedure, described in Algorithm 6, is guaranteed to terminate; in fact, let us consider that in a given iteration of the process we compute a triangle T_α of a computed orbit $O(T)$. Since T_α is a triangle of a closed orbit not enclosing $z_\mathcal{I}$ nor $z_\mathcal{E}$, we can state that the polygons $l_i(O(T))$ and $l_e(O(T))$ intersect the line $(z_\mathcal{I}, S_{\alpha\beta}, z_\mathcal{E})$ two times at least. Let x, x_1 and y, y_1 be two points of the intersections of $(z_\mathcal{I}, S_{\alpha\beta}, z_\mathcal{E})$ with $l_i(O(T))$ and $l_e(O(T))$ respectively (see Figure 7), and consider the two points

$$(\tilde{x}, \tilde{y}) = \begin{cases} (x_1, y_1) \text{ when } \{x, y\} \cap \{z_\mathcal{I}, z_\mathcal{E}\} \neq \emptyset \\ (x, y) \quad \text{otherwise.} \end{cases}$$

The vertices $\tilde{x}$ and $\tilde{y}$ satisfy

$$s(\tilde{x}) \leq \sigma < s(\tilde{y}) \tag{3}$$

$$||\tilde{y} - z_\mathcal{I}|| < ||z_\mathcal{E} - z_\mathcal{I}|| \tag{4}$$

$$||\tilde{x} - z_\mathcal{E}|| < ||z_\mathcal{E} - z_\mathcal{I}|| \tag{5}$$

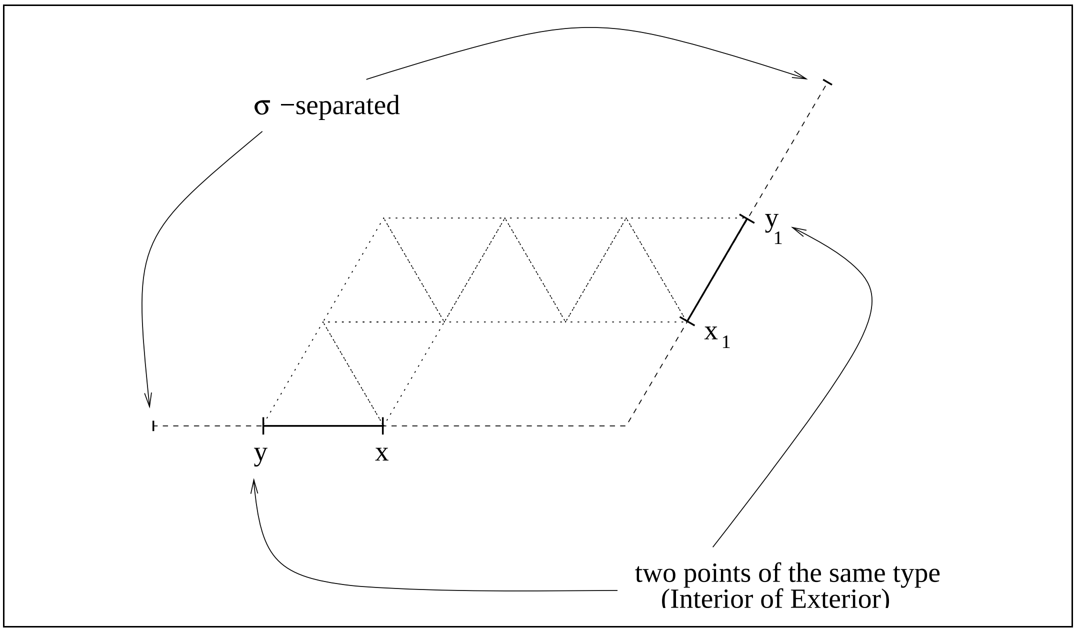


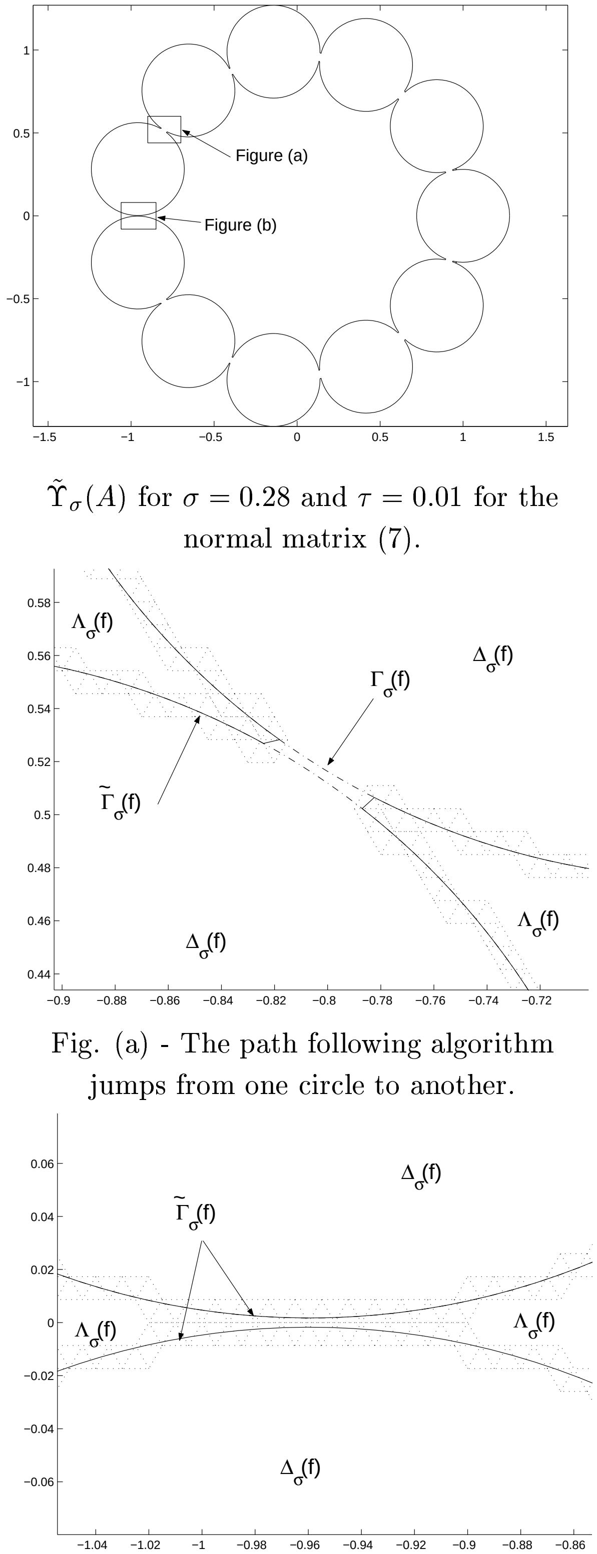
Figure 7. The intersection of $O(T)$ and $z_{\mathcal{I}}, S_{\alpha\beta}, z_{\mathcal{E}}$.

$$\{\tilde{x}, \tilde{y}\} \cap (\mathcal{I} \cup \mathcal{E}) \neq \emptyset, \tag{6}$$

which contradicts the fact that $z_{\mathcal{I}}$ and $z_{\mathcal{E}}$ minimize the distance $d(x, y) = ||y - x||$.

Since every point of $\mathcal{I}$ is enclosed within a connected component, and since the number of connected components is bound by the matrix dimension; therefore the described procedure is guaranteed to terminate.

In case of close components, the path following algorithm might *jump* from one component to another; therefore, it might capture the pattern of multiple components of the level curve as if they were one. Figures 8, shows different behaviors of the algorithm for the same level curve $\sigma = 0.28$ of the matrix defined in (7). In this case, A is normal; therefore, the level curve is composed of the set of circles centered at the eigenvalues $z_k = e^{\frac{2ik\pi}{11}}$ with radius $r = 0.28$. In figure 8-(a), the algorithm jumps from one circle to another whereas, in figure 8-b, the algorithm keeps track of the two circles; this is mainly due to the relative orientation of the triangles with respect to the level curve. This is also true for the cases where the level curve crosses itself; this is shown in Figure 9 where the path following algorithm computes one of the two connected components (top) or merges the two components (bottom).



$\tilde{\Upsilon}_\sigma(A)$ for $\sigma = 0.28$ and $\tau = 0.01$ for the normal matrix (7).

Fig. (a) - The path following algorithm jumps from one circle to another.

Fig. (b) - The algorithm keeps track of two distinct circles.

Figure 8. Merging two close components of the computed level curve $\tilde{\Upsilon}_\sigma(A)$ for $\sigma = 0.28, \tau = 0.01$ in the case of the normal matrix (7).

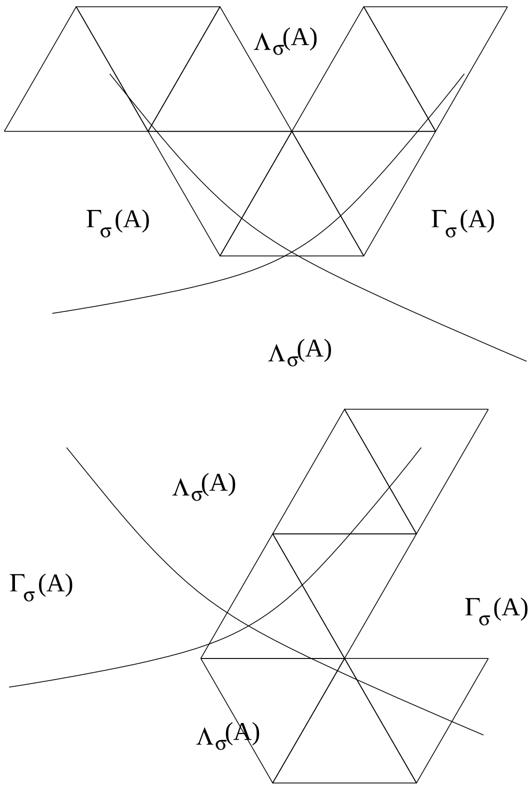


Figure 9. A crossing point along the level curve.

5. Test problems and numerical results

In this section, we report some numerical results obtained by using a Matlab prototype implementation[3] of the procedure previously described to demonstrate its performance.

Let us first consider the orthogonal matrix

$$A_1 = \begin{pmatrix} 0 & 1 \\ I_{10} & 0 \end{pmatrix} \tag{7}$$

and I_{10} is the 10×10 identity matrix. Matrix A_1 is normal, therefore, the level curve for a given σ is the union of the circles centered at the eigenvalues with radius σ. Figures 10 and 11 show the level curves for $\sigma = 0.28, 0.3, 0.5, 1$ and $\tau = 0.01, 0.1$. For the particular case where $\sigma = 0.28$, adjacent circles are only

[3] This code is available via anonymous ftp from the site `ftp.irisa.fr/local/aladin/philippe/PAT` .

separated by a distance $d \simeq 0.0034 < \tau$; therefore, the path following algorithm jumps from one component of the level curve to another.

Next, we consider the non normal `GRCAR` matrix $A \in \mathbb{R}^{100\times 100}$ given by

$$A_2 = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & \dots & \dots & 0 \\ -1 & 1 & 1 & 1 & 1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & 1 \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & 1 \\ \vdots & & & & \ddots & \ddots & \ddots & 1 \\ 0 & \dots & \dots & \dots & \dots & 0 & -1 & 1 \end{pmatrix} \tag{8}$$

This matrix is ill-conditioned with respect to its eigensystem. Figure 12 shows the level curves for the input presented in the following table :

σ	τ	$\mathcal{I}$	$\mathcal{E}$
8.730×10^{-12}	0.01	$\{2i, -2i\}$	$\{\}$
4.585×10^{-5}	0.01	$\{1.25\}$	$\{\}$
4.712×10^{-3}	0.01	$\{-0.5 - 2i\}$	$\{\}$
1.494×10^{-1}	0.01	$\{-0.75 - 2i\}$	$\{\}$

Since the evaluation of ϕ is more expensive than in the previous tests, we used a parallel implementation of the code [8]. This example shows that the proposed algorithm copes easily with directional discontinuities along the level curve.

Table 2 gathers the statistics concerning the different numerical examples where `Length` is the length of the computed level curve, `Triangles` is the triangle count and $\sigma_{\min}$ is the number of evaluations of $\sigma_{\min}$. Notice that we used the bisection in the MatLab prototype to compute the level curve points whereas `ZEROIN` is used in the parallel implementation of PAT presented in [8]. The procedure `ZEROIN` which is available on `Netlib` combines efficiently the bisection, the secant method and even a quadratic interpolation [3]. However, the tests performed for an acceptable precision equal to $10^{-2}\tau$ using `ZEROIN` and the bisection process proved that both algorithms are equivalent (requiring 7 evaluations of $s(z)$ per segment). Finally, Figure 13 presents the ratio (Effective Number of triangles) / (Maximum number of triangles).

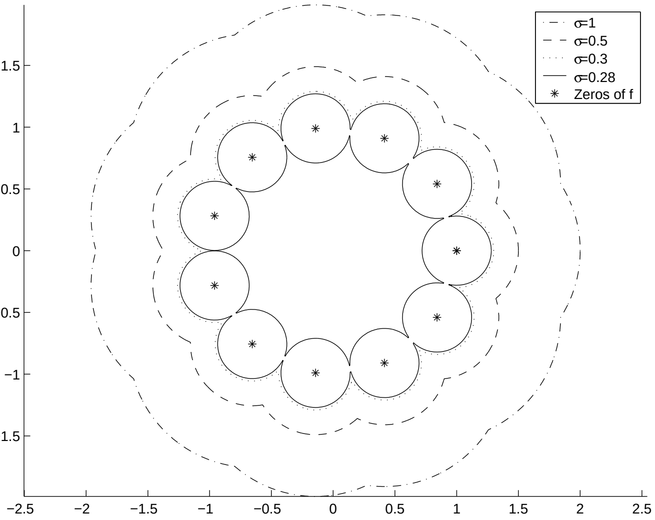


Figure 10. Level curves for $\phi(z) = \sigma_{\min}(A_1 - zI)$ with $\tau = 0.01$

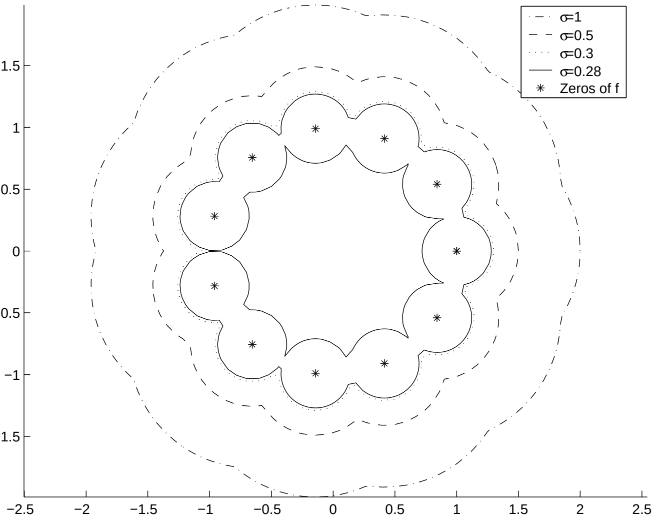


Figure 11. Level curves for $\phi(z) = \sigma_{\min}(A_1 - zI)$ with $\tau = 0.1$

6. Conclusion

We have shown that the combination of the $F-$orbit with a bisection algorithm results in a numerically stable path following strategy for determining pseudospectra. The construction of the $F-$orbit is guaranteed to terminate even in the presence of round-off errors. The proposed technique is able to handle

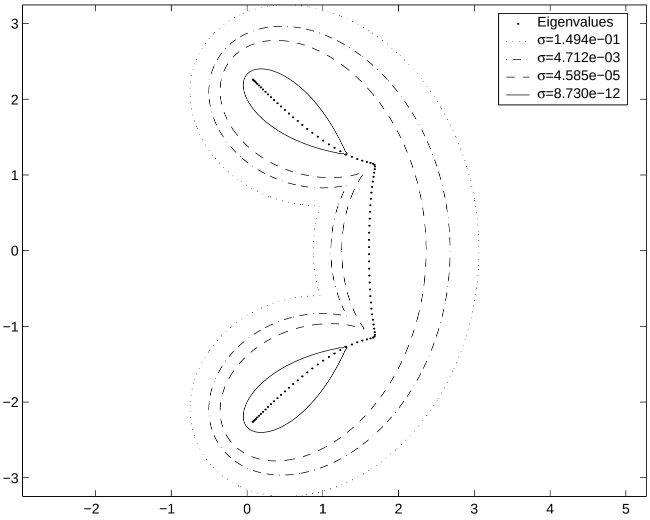


Figure 12. Level curves for $\phi(z) = \sigma_{\min}(A_2 - zI)$ with $\tau = 0.01$

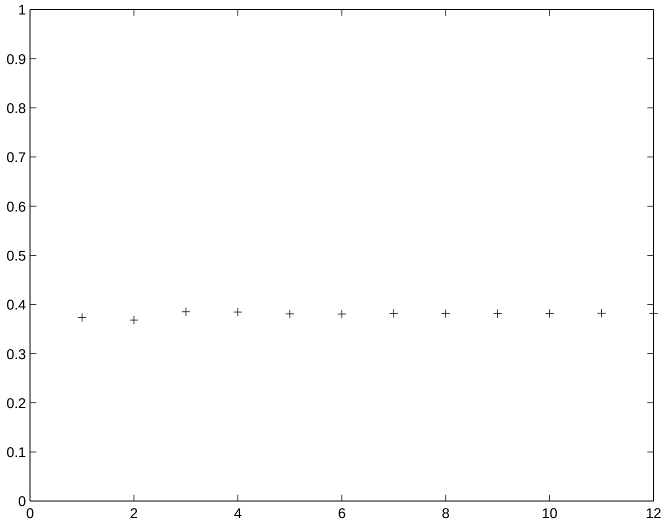


Figure 13. Matrix A_2 : Ratio (Triangle Count)/$\frac{10l}{\tau\sqrt{3}}$.

singular points along the level curve without difficulty.

The fact that the computation of the $F-$orbit is separated from the bisection process means that one can parallelize the bisection on a cluster of machines [8]. This is important, given the high cost of computing $\sigma_{\min}$, since the bisection process on different triangles are completely independent and can be run in

σ	Length	Triangles	$\frac{l}{\tau}$	$\frac{10l}{\tau\sqrt{3}}$	$\sigma_{\min}$
$\phi(z) = \sigma_{\min}(A - zI)$, A is normal					
0.280	16.434	356	165	953	2850
0.300	9.308	200	94	543	1602
0.500	9.669	216	97	561	1730
1.000	12.552	280	126	728	2242
0.280	18.492	4068	1850	10681	32546
0.300	9.845	2164	985	5687	17314
0.500	9.719	2144	972	5612	17154
1.000	12.565	2768	1257	7258	22146
$\phi(z) = \sigma_{\min}(A - zI)$, GRCAR problem					
8.730×10^{-12}	7.602	1675	761	4394	13402
4.585×10^{-5}	16.858	3716	1686	9735	29730
4.712×10^{-3}	17.813	3932	1782	10288	31458
1.494×10^{-1}	19.168	4220	1917	11068	33762

Table 2
Level curve length and the corresponding triangle count

parallel. This may yield a substantial speed up of the path following. In [8], successful tests with a matrix of order 8192 are reported.

7. Acknowledgment

The authors are very grateful to J. Belward and J. Erhel for their helpful suggestions during the preparation of this paper.

References

[1] Allgower, E., and Georg, K. Continuation and path following. *Acta Numerica* (1993), 1–64.

[2] Bekas, C., and Gallopoulos, E. Cobra: Parallel path following for computing the matrix pseudospectrum. To appear in Parallel Computing.

[3] Brent, R. *Algorithms for minimization without derivatives.* Prentice-Hall, 1973.

[4] Brühl, M. A curve tracing algorithm for computing the pseudospectrum. *BIT 36*, 3 (1996), 441–454.

[5] Golub, G., and Loan, C. V. *Matrix Computations*, 2nd ed. The John Hopkins University Press, 1989.

[6] Huilfieldt, J., and Ruhe, A. A new algorithm for numerical path following applied to an example from hydrodynamical flow. *SIAM J. Sci. Statist. Comput.*, 11 (1990), 1181–1192.

[7] R. Mejia. Mejia, R. Conkub: A conversational path-follower for systems of nonlinear equations. *J. Comput. Phys.*, 63 (1986), 67–84.

[8] Mezher, D., and Philippe, B. Parallel computation of the pseudospectrum of large sparse matrices. To appear in Parallel Computing.

[9] Lui, S. Computation of pseudospectra by continuation. *SIAM Journal on Scientific Computing 18*, 2 (1997), 565–573.

[10] Philippe, B., and Sadkane, M. Computation of the fundamental singular subspace of a large matrix. *Linear algebra and its application*, 257 (1997), 77–104.

[11] Rudin, W. *Real and Complex analysis*, 3 ed. McGraw-Hill International Editions, 1986.

[12] Saad, Y. *Numerical methods for large eigenvalue problems.* Series in Algorithms and Architectures for advanced scientific computing. Manchester University Press, 1992.

[13] Schwetlick, H., Timmermann, G., and Losche, R. Path following for large nonlinear equations by implicit block elimination based on recursive projections. *Lectures in Applied Mathematics 32* (1996), 715–732.

[14] Schwetlick, H., and Schnabel, U. Iterative computation of the smallest singular value and the corresponding singular vectors of a matrix. Preprint IOKOMO-06-97, Techn. Univ. Dresden (1997).

[15] Trefethen, L. Computation of pseudospectra. *Acta Numerica* (1999), 247–295. Available at web.comlab.ox.ac.uk/oucl/work/nick.trefethen.

[16] Trefethen, L. Pseudospectra of linear operators. *SIAM Revue 39*, 3 (1997), 383–406.

[17] Ueberhuber, C. *Numerical Computation 2, Methods, Software and Analysis.* Springer, 1997.