`specdicho` Version 1.0
User's guide

Miloud Sadkane
University of Brest, France
and
Ahmed Touhami
Hassan 1st University, Morocco

This document is an user guide of `specdicho`, a Matlab program of spectral dichotomy of regular matrix pencils described in the main paper. We present in detail the data structures, parameters and calling sequences. Example programs using `specdicho` are also given.

Categories and Subject Descriptors: G.1.3 [**Numerical Analysis**]: Numerical Linear Algebra; G.4 [**Mathematical Software**]: Documentation

General Terms: Algorithms, Documentation

Additional Key Words and Phrases: invariant subspace, regular matrix pencil, spectral dichotomy, spectral projector

1. SPECDICHO CALLS

We first discuss the basic calling structure of the `specdicho` function and then introduce the optional arguments that user can use to optimize performance. Our MATLAB implementation `specdicho` gathers the four algorithms (i.e. DICHOC, DICHOE, DICHOI, DICHOP) seen before. Its most basic calls are

```
>> specdicho(A)
>> [P,H] = specdicho(A)
```

where **A** is a numeric square matrix. In this call the matrix **B** is assumed to be the identity matrix and the default geometry of work (i.e. positively oriented contour in the complex plane) is the unit circle.

Author's address: Miloud Sadkane, Université de Brest. Laboratoire de Mathématiques. CNRS - UMR 6205. 6, Av. Le Gorgeu. 29238 Brest Cedex 3. France. E-mail: Miloud.Sadkane@univ-brest.fr.
Ahmed Touhami, Mathematics and Computer Science Department, Hassan 1st University, Faculty of Sciences and Technologies, BP 577, route de Casablanca, Settat, Morocco. E-mail: Ahmed.Touhami@gmail.com. This work has been partially supported by A.U.F (Agence Universitaire de la Francophonie) for visiting Département de Mathématiques, Brest, UMR CNRS 6205.

When an output argument is not specified, the call to `specdicho` displays the spectral projector onto the invariant subspace of the matrix $\mathbf{A}$ associated with the eigenvalues inside the unit circle. Otherwise it returns the matrices $\mathbf{P}$ and $\mathbf{H}$, where $\mathbf{P}$ is the spectral projector just mentioned and $\mathbf{H}$ is the matrix integral given by formula (2) from the main paper with $\mathbf{B}$ the identity matrix.

To compute the spectral projector onto the right deflating subspace of a regular matrix pencil of the form $\lambda\mathbf{B} - \mathbf{A}$ associated with the eigenvalues inside or outside the unit circle, we append $\mathbf{B}$ after $\mathbf{A}$ in the above calls, namely

```
>> specdicho(A,B)
>> [P,H] = specdicho(A,B)
```

which returns the matrices $\mathbf{P}$ and $\mathbf{H}$ as above. We note that $\mathbf{B}$ must be a numeric dense square matrix, of the same size as $\mathbf{A}$.

The `specdicho` program also uses an option structure to provide extra information to improve performance. Users can specify a set of optional parameters via a MATLAB structure. This is done by first setting the value in the structure, e.g.

```
>> opts.Ho = eye(size(A))
```

then passing `opts` to `specdicho` by calling

```
>> [P,H] = specdicho(A,opts)
>> [P,H] = specdicho(A,B,opts)
```

The informational options `specdicho` are:

`opts.geom`	positively oriented contour in the complex plane.
`opts.c`	center of the circle.
`opts.r`	radius of the circle.
`opts.a`	semi-major axis of the ellipse.
`opts.b`	semi-minor axis of the ellipse.
`opts.p`	positive real parameter of the parabola.
`opts.mxiter`	maximal number of iterations to perform.
`opts.tol`	tolerance used for convergence check.
`opts.Ho`	Hermitian positive definite matrix used for scaling purposes.

The `geom` option lets the user to specify the geometry of work. It must be equal to `'C'`or `'c'`: circle or `'E' or 'e'`: ellipse or `'I' or 'i'`: imaginary axis or `'P' or 'p'`: parabola. Its default value is `'C'`.

The `c` option lets the user to specify the center of the circle when `geom` is equal to `'C'`. It must be a real or complex number. Its default value is `0`.

The `r` option lets the user to specify the radius of the circle when `geom` is equal to `'C'`. It must be a positive real number. Its default value is `1`.

The `a` option allows the user to specify the semi-major axis of the ellipse (formula (18) from the main paper) when `geom` is equal to `'E' or 'e'`. It must be a positive real number. Its default value is `5`.

The `b` option allows the user to specify the semi-minor axis of the ellipse (fromula (18) from the main paper) when `geom` is equal to `'E' or 'e'`. It must be a positive real number and $\mathtt{a} \geq \mathtt{b} > 0$. Its default value is `1`.

The `p` option allows the user to specify the parameter of the parabola given by (formula (29) from the main paper) when `geom` is equal to `'P' or 'p'`. It must be a positive real number. Its default value is `1`.

The `mxiter` option lets the user to specify the maximal number of iterations `specdicho` will perform. The default value is `10`. The user can set a larger value for particularly difficult problems.

The `tol` option allows the user to specify the value of *tol* in (formula (17) from the main paper). Its default value is $\texttt{10}^{-10}$. It can be much smaller (e.g., `tol` = eps) for difficult problems.

The `Ho` option must be of the same size as **A** when `geom = ['C'|'c'|'I'|'i']` and of twice the order of **A** when `geom = ['E'|'e'|'P'|'p']`. Its default value is `eye(size(A))` when `geom = ['C'|'c'|'I'|'i']` and set to `eye(2*size(A))` when `geom = ['E'|'e'|'P'|'p']`.
If ever the user specifies `Ho = []`, this will not be taken into account in `specdicho` and the `Ho` option will keep its default value.

Note that the matrix **P** corresponds to the spectral projector onto the invariant subspace of the matrix **A** associated with the eigenvalues inside `geom` when `geom = ['C'|'c'|'E'|'e'|'P'|'p']` and with negative real parts when `geom = 'I' or 'i'`. For the pencil problem, when `geom = ['I'|'i'|'P'|'p']`, the non-singularity of **B** is required.

2. SAMPLE EXPERIMENTS

We now present some examples of calling `specdicho` and the corresponding outputs. All experiments were carried out using MATLAB version 6.1(R12.1). In the first set of experiments, **A** is the 6×6 Frank matrix. It can be generated in MATLAB with the command

```
>> A = gallery('frank',6);
```

The eigenvalues of the matrix **A** are `12.973   5.383   1.835   0.544   0.077   0.185`.

We first consider the problem of spectral dichotomy with respect to the unit circle. The command

```
>> specdicho(A)
```

makes `specdicho` compute the spectral projector onto the invariant subspace of the matrix **A** associated with the eigenvalues inside the unit circle using the default values for all parameters (i.e. `B = eye(6), geom = 'C', c = 0, r = 1, a = 5, b = 1, p = 1, mxiter = 10, tol = 1e-10, Ho = eye(6)`). The output is

```
 Circle of center c = (0,0) and radius r = 1
 At iteration 7
 convergence to the desired tolerance tol = 1e-10

ans =
    0.1936   -0.2172    0.0087    0.0198   -0.0020   -0.0065
   -0.6127    0.7117   -0.0902   -0.0147    0.0043    0.0045
    0.4735   -0.6628    0.3638   -0.2200    0.0217    0.0547
    0.6470   -0.4756   -0.6331    0.6272   -0.1555   -0.0518
```

```
   -0.5396     0.5348     0.2031    -0.4057     0.4202    -0.3670
   -0.7783     0.6704     0.4362    -0.2246    -0.4707     0.6835
```

Note that MATLAB creates the `ans` variable automatically when no output argument is specified.
To compute only the spectral projector $\mathbf{P}$ onto the invariant subspace of the matrix $\mathbf{A}$ associated with the eigenvalues inside the unit circle, we use

```
>> [P] = specdicho(A);
 Circle of center c = (0,0) and radius r = 1
 At iteration 7
 convergence to the desired tolerance tol = 1e-10
```

If both the spectral projector $\mathbf{P}$ and the matrix $\mathbf{H}$ given by formulas (1)-(2) form the main paper, are desired, then `specdicho` should be called as follows

```
>> [P,H] = specdicho(A);
 Circle of center c = (0,0) and radius r = 1
 At iteration 7
 convergence to the desired tolerance tol = 1e-10
```

We now present some examples using the options. The following specifies a radius of the circle (`r`), a maximal number of iterations (`mxiter`) to perform and a tolerance (`tol`).

```
>>  opts.r = 5.38;
>>  opts.mxiter = 20;
>>  opts.tol = 1e-12;
```

To compute the spectral projector $\mathbf{P}$ onto the invariant subspace of $\mathbf{A}$ associated with the eigenvalues inside the circle of center 0 and radius 5.38 and the matrix $\mathbf{H}$ and, we use

```
>> [P,H] = specdicho(A,opts);
 Circle of center c = (0,0) and radius r = 5.38
 At iteration 17
 convergence to the desired tolerance tol = 1e-12
```

To display the dichotomy condition number, the accuracy of the spectral projector measured by $\|\mathbf{P}^2 - \mathbf{P}\|$ and the trace of $\mathbf{P}$ (which corresponds to the sum of algebraic multiplicities of eigenvalues enclosed by the used circle) we use

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H)) ' NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P)) ' TRACE(P) = ', num2str(trace(P)) ]));

NORM(H) = 1533.4825     NORM(P^2 - P) = 8.8108e-14     TRACE(P) = 4
```

For the pencil problem, a user must provide the matrices $\mathbf{A}$ and $\mathbf{B}$. The matrix $\mathbf{B}$ must be numeric square with the same size as $\mathbf{A}$. For instance, we use a diagonal matrix for $\mathbf{B}$

```
>> B = diag(0:1:5);
```

The eigenvalues of the pencil $\lambda\mathbf{B}-\mathbf{A}$ are ∞ `3.427 0.788 0.211 0.072 0.033`.

We now consider the problem of spectral dichotomy of this matrix pencil with respect to an ellipse. The command

```
>> clear opts
>> opts.geom = 'E'
>> specdicho(A,B,opts);
```

yields the output

```
 Ellipse (X/a)^2 + (Y/b)^2 = 1
 with a = 5 and b = 1
 At iteration 7
 convergence to the desired tolerance tol = 1e-10
```

As mentioned in Section 1, we can specify a set of optional parameters via a MATLAB structure. The following specifies a maximal number of iterations (`mxiter`) to perform and a tolerance (`tol`).

```
>>  clear opts
>>  opts.mxiter = 6;
>>  opts.tol = 1e-2;
```

The command

```
>> [P,H] = specdicho(A,B,opts);
```

yields

```
Circle of center c = (0,0) and radius r = 1
At iteration 6
convergence to the desired tolerance tol = 0.01
```

To display the dichotomy condition number, the accuracy of the spectral projector and its trace we use

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H)) ' NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P)) ' TRACE(P) = ', num2str(trace(P)) ]));

NORM(H) = 5.2966     NORM(P^2 - P) = 3.8141e-07     TRACE(P) = 4
```

```
>> opts = struct('mxiter',20,'tol',eps)
opts =
    mxiter: 20
       tol: 2.2204e-16
```

The command

```
>> [P,H] = specdicho(A,B,opts);
```

yields

```
 Circle of center c = (0,0) and radius r = 1
 At iteration 9
 convergence to the desired tolerance tol = 2.2204e-16
```

To display the dichotomy condition number, the accuracy of the spectral projector and its trace we use

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H)) ' NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P)) ' TRACE(P) = ', num2str(trace(P)) ]));

NORM(H) = 5.2966     NORM(P^2 - P) = 1.2943e-15     TRACE(P) = 4
```

In the following experiments, we consider two examples. The first one shows that the choice of $\mathbf{H}^{(0)}$ may have an influence on the dichotomy condition number $\|\mathbf{H}\|$ especially when there is a problem of scaling. The second one indicates a negative impact of a large value of $\|\mathbf{H}\|$ on the numerical quality of $\mathbf{P}$.

The first example is the matrix pencil $\lambda\mathbf{B} - \mathbf{A}$ where

$$\mathbf{A} = \begin{pmatrix} 10^{-3} & 10^{3} \\ 0 & 10^{-3} \end{pmatrix} \quad \text{and} \quad \mathbf{B} = \begin{pmatrix} 10^{-5} & 0 \\ 0 & 1 \end{pmatrix}. \tag{1}$$

The eigenvalues of this matrix pencil are 10^2 and 10^{-3}. We first compute the spectral projector $\mathbf{P}$ onto the right deflating subspace associated with the eigenvalues inside the unit circle and the matrix $\mathbf{H}$ using default values for all parameters.

```
>> [P,H] = specdicho(A,B);
 Circle of center c = (0,0) and radius r = 1
 At iteration 4
 convergence to the desired tolerance tol = 1e-10
```

To display the dichotomy condition number, the accuracy of the spectral projector and its trace we use

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H)) ' NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P)) ' TRACE(P) = ', num2str(trace(P)) ]));

NORM(H) = 1.0001e+12     NORM(P^2 - P) = 2.2204e-32     TRACE(P) = 1
```

Note the large value of $\|\mathbf{H}\|$ despite the good numerical quality of $\mathbf{P}$.
We now specify $\mathbf{H}^{(0)} = \mathbf{A}^\star\mathbf{A} + \mathbf{B}^\star\mathbf{B}$. This is done by setting the value in the MATLAB structure:

```
>> opts.Ho = A'*A + B'*B;

>> [P,H] = specdicho(A,B,opts);
 Circle of center c = (0,0) and radius r = 1
 At iteration 4
 convergence to the desired tolerance tol = 1e-10
```

The dichotomy condition number, the accuracy of the spectral projector and its trace are given by

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H))'NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P))' TRACE(P) = ', num2str(trace(P)) ]));

 NORM(H) = 201.5231     NORM(P^2 - P) = 2.2204e-32     TRACE(P) = 1
```

In the second example **A** is a Jordan block of size 10 with eigenvalue 0.88 generated as

```
>> A = gallery('jordbloc',10,0.88);
```

Using the default values for the parameters we obtain

```
>> [P,H] = specdicho(A);
 Circle of center c = (0,0) and radius r = 1
 At iteration 10
 convergence to the desired tolerance tol = 1e-10
```

The dichotomy condition number, the accuracy of the spectral projector and its trace are given by

```
>> disp(sprintf(['\n NORM(H) = ',num2str(norm(H)) 'NORM(P^2 - P) = ' ...
      num2str(norm(P^2-P)) ' TRACE(P) = ', num2str(trace(P)) ]));

 NORM(H) = 3.1881e+16      NORM(P^2 - P) = 4.1787e-8     TRACE(P) = 10
```

These values remain essentially the same whatever the parameters `mxiter` and `tol` are.