Generalized Polynomial Power Method

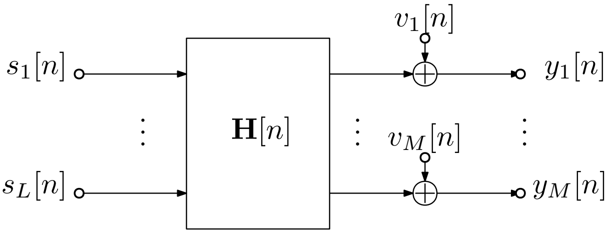
F. A. Khattak, I. K.Proudler and Stephan Weiss

Department of Electronic and Electrical Engineering
University of Strathclyde, Glasgow

Sensor Signal Processing for Defence (SSPD), 2023

Background: Polynomial or Convolutive Mixing

- Lets assume a MIMO system of $L-$sources and $M-$sensors



where $\boldsymbol{v}_i$ refers to additive noise.
Elements of $\mathbf{H}[n]$ are time-sequences i.e. FIR filters.

Background: Polynomial Singular Value Decomposition

- Z-transform of $\mathbf{H}[n]$ i.e. $\boldsymbol{H}(z)$ will be a polynomial matrix. An example polynomial matrix can be

$$\boldsymbol{H}(z) = \begin{bmatrix} 1 + z^{-1} & -1 + z^{-1} \\ -1 + 2z^{-1} & 2 - z^{-1} + z^{-2} \end{bmatrix}$$

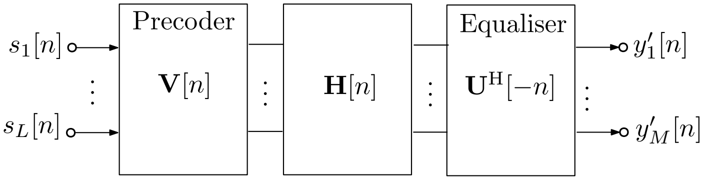
- To decouple or perform pre-coding or equalisation of such MIMO systems, SVD of $\boldsymbol{H}(z)$, defined as

$$\boldsymbol{H}(z) = \boldsymbol{U}(z)\boldsymbol{\Sigma}(z)\boldsymbol{V}^{\mathrm{P}}(z)$$

where $\boldsymbol{\Sigma}(z)$ is a diagonal matrix containing singular values, and $\boldsymbol{U}(z)$, $\boldsymbol{V}(z)$ are paraunitary matrices, can be useful for precoding and equalisation.

Background: Polynomial Singular Value Decomposition

Precoding and equalisation can be carried out as



because

$$\boldsymbol{y}'(z) = \boldsymbol{U}^{\mathrm{P}}(z)\boldsymbol{U}(z)\boldsymbol{\Sigma}(z)\boldsymbol{V}^{\mathrm{P}}(z)\boldsymbol{V}(z)\boldsymbol{s}(z) = \boldsymbol{\Sigma}(z)\boldsymbol{s}(z)$$

Paraunitarity of $\boldsymbol{U}(z)$ and $\boldsymbol{V}(z)$ means two things:

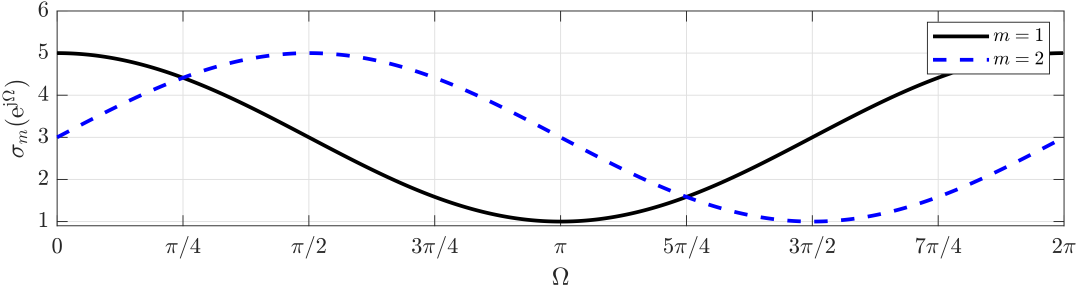
- transmit power remains same.
- noise power is not amplified.

Background: Polynomial Matrix PSVD Example

Singular values of

$$\boldsymbol{H}(z) = \begin{bmatrix} \frac{1-\mathrm{j}}{2}z + 3 + \frac{1+\mathrm{j}}{2}z^{-1} & \frac{1+\mathrm{j}}{2}z^{2} + \frac{1-\mathrm{j}}{2} \\ \frac{1-\mathrm{j}}{2}z^{-2} + \frac{1+\mathrm{j}}{2} & \frac{1-\mathrm{j}}{2}z + 3 + \frac{1+\mathrm{j}}{2}z^{-1} \\ 0 & 0 \end{bmatrix} \in \mathbb{C}^{3\times 2}$$

evaluated on unit circle



PSVD Algorithms

- 2-Polynomial eigenvalue decomposition (PEVDs): PSVD can be computed with help of two PEVDs, but it is computationally expensive and produces complex singular values. [1]
- Multiple iterations of polynomial QR decomposition (PQRD) has also been used but it is also expensive. [2]
- Generalized sequential second order best rotation (GSBR2) [3], a dedicated PSVD algorithm, utilizing Kogbetliantz transformation combined with SBR approach. Multiple shift or similar strategies cannot be applied and therefore, is slow in convergence. It produces complex singular values as opposed to convention.

Research Intent

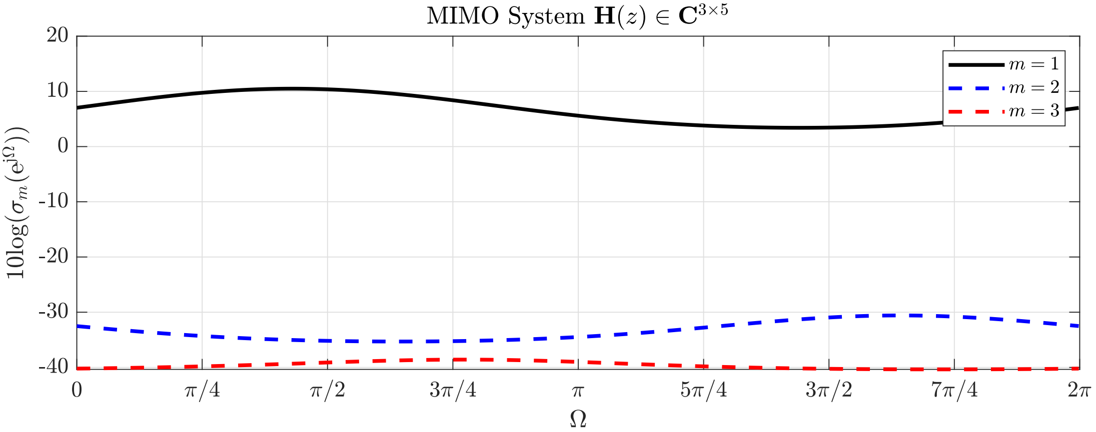
Polynomial power method, an extension of the power method, from Hermitian to para-Hermitian matrices can be extended to general polynomial matrices for the dominant singular value and its vectors extraction. [4]

Similar to deflation approach combined with the polynomial power method for low-rank PEVD, deflation can be combined with generalized polynomial power method for computing low-rank PSVD.

Motivation

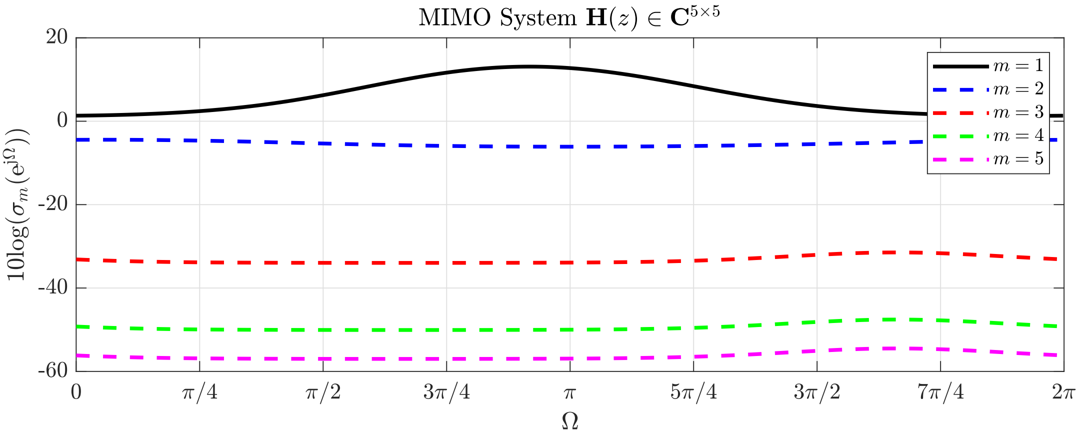
Consider two case

1. Single dominant singular value



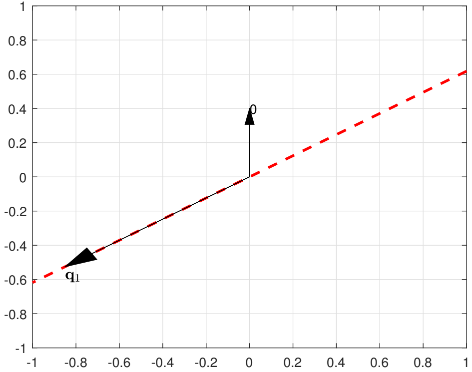
Motivation

- Multiple, but comparatively fewer dominant singular values compared to the dimension of $\boldsymbol{H}(z)$



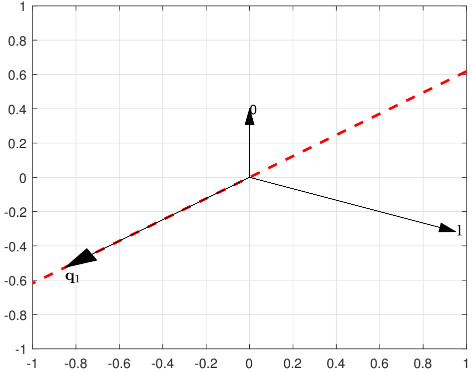
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



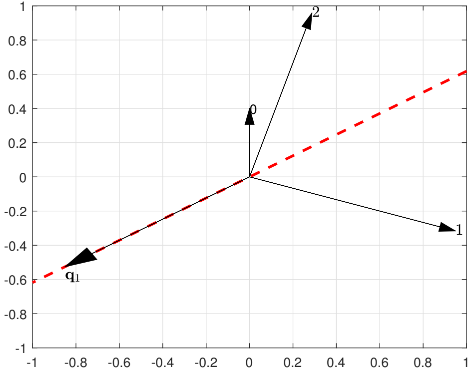
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \dots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



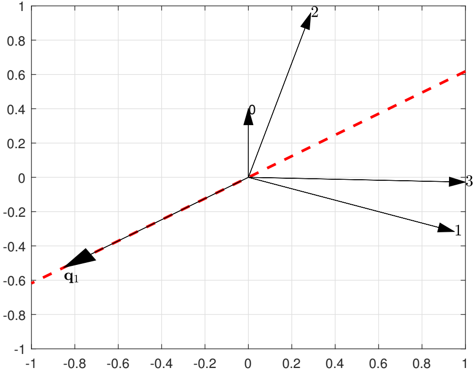
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



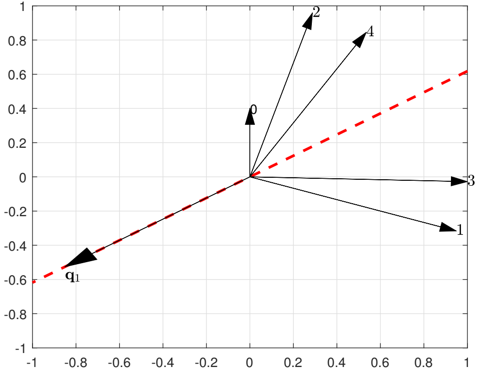
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



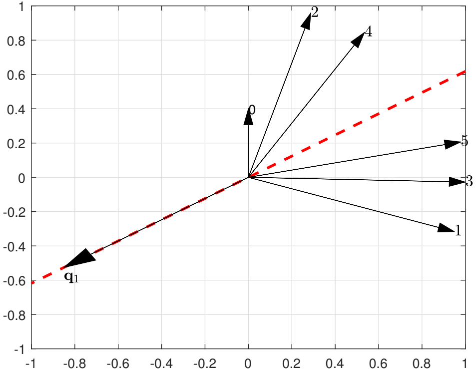
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



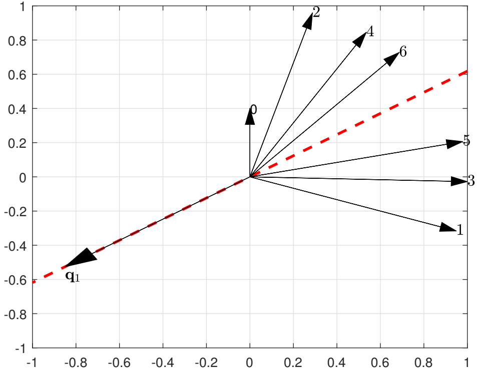
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



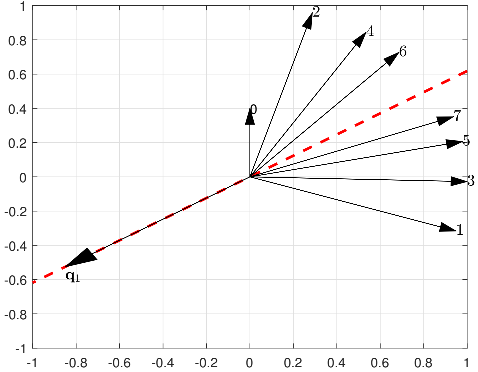
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



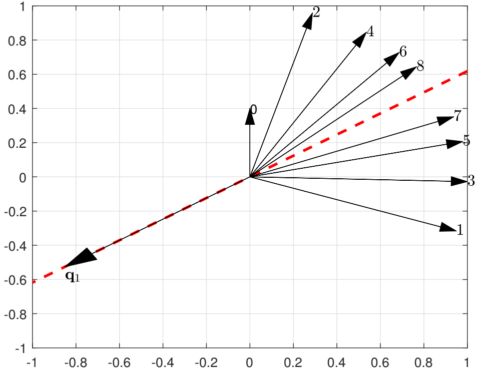
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



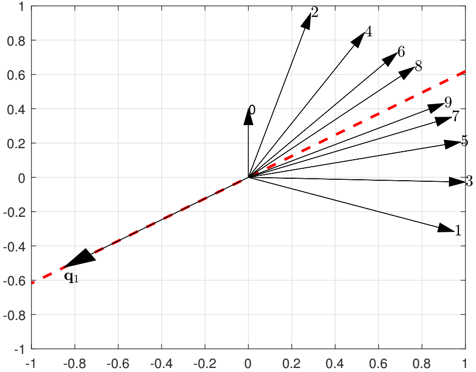
Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



Power Method: Dominant Eigenpair Extraction

A sequence of vector generated as $\mathbf{x}^{(k)} = \mathbf{A}\mathbf{x}^{(k-1)}, k = 1, 2, \ldots$ eventually converges to the dominant eigenvector. For example for $\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ which has $\mathbf{q}_1 = [-0.8507, -0.5257]^{\mathrm{T}}, \lambda_1 = 3.8541$, with $\mathbf{x}^{(0)} = [0, 0.4]^{\mathrm{T}}$, we have



Polynomial Power Method

- Similar sequence of polynomial vectors is generated

$$\boldsymbol{x}^{(k)}(z) = \boldsymbol{R}(z)\boldsymbol{x}^{(k-1)}(z),\ k = 1, 2, ...$$

where $\boldsymbol{R}(z)$ is a para-Hermitian matrix.

- Post-iteration, $\boldsymbol{x}^{(k)}(z)$ is normalized to unit-norm on each frequency on unit-circle, i.e.

$$||\boldsymbol{x}^{(k-1)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})||_2 = 1\ \forall\ \Omega$$

- Algorithm is stopped once the Hermitian angle $\alpha(\Omega)$, defined as

$$\alpha(\Omega) = \angle\{\boldsymbol{x}^{(k)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega}), \boldsymbol{x}^{(k-1)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})\} = \mathrm{acos}\left(\frac{|\boldsymbol{x}^{(k),\mathrm{H}}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})\boldsymbol{x}^{(k-1)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})|}{\|\boldsymbol{x}^{(k)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})\|_2 \cdot \|\boldsymbol{x}^{(k-1)}_{\mathrm{norm}}(\mathrm{e}^{\mathrm{j}\Omega})\|_2}\right)$$

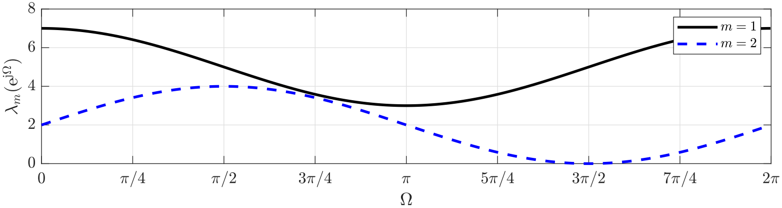
is negligibly small at each Ω.

Polynomial Power Method: Dominant Eigenpair Extraction

Consider a para-Hermitian polynomial matrix $\boldsymbol{R} \in \mathbb{C}^{2\times 2}$

$$\boldsymbol{R}(z) = \frac{1}{2}\begin{bmatrix} (1-\mathrm{j})z + 7 + (1+\mathrm{j})z^{-1} & (1+\mathrm{j})z^{2} + 3z + 1 - \mathrm{j} \\ (1+\mathrm{j}) + 3z^{-1} + (1-\mathrm{j})z^{-2} & (1-\mathrm{j})z + 7 + (1-\mathrm{j})z \end{bmatrix}$$

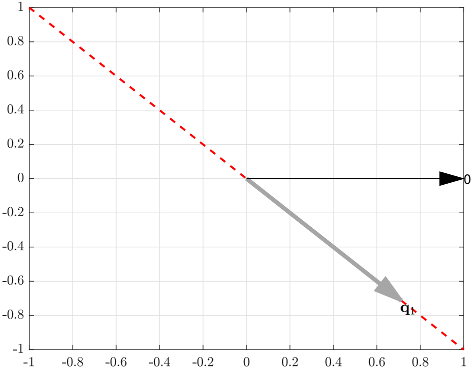
with spectrally majorised eigenvalues as shown



with $\boldsymbol{q}_{1,2}(z) = 1/\sqrt{2}[1, \pm z^{-1}]^{\mathrm{T}}$.

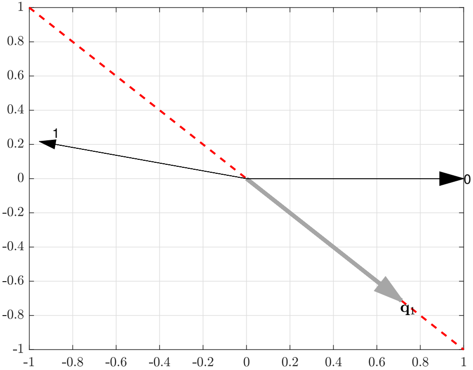
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt(2)[1, -1]^{\mathrm{T}}$



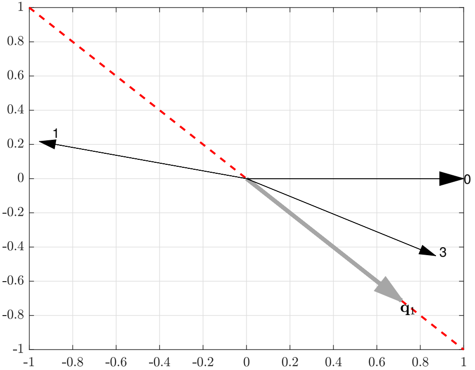
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt(2)[1, -1]^{\mathrm{T}}$



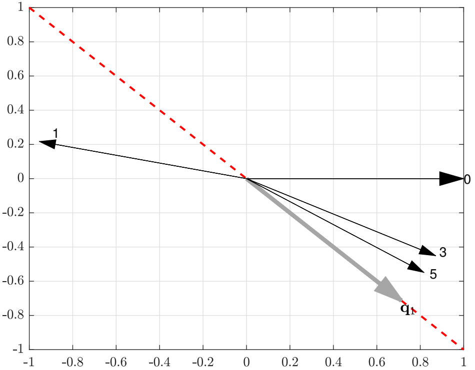
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt(2)[1, -1]^{\mathrm{T}}$



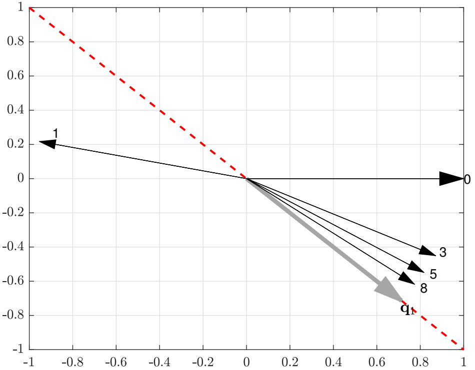
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt{(2)}[1, -1]^{\mathrm{T}}$



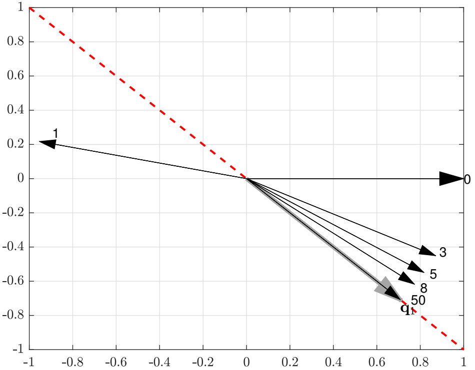
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt(2)[1, -1]^{\mathrm{T}}$



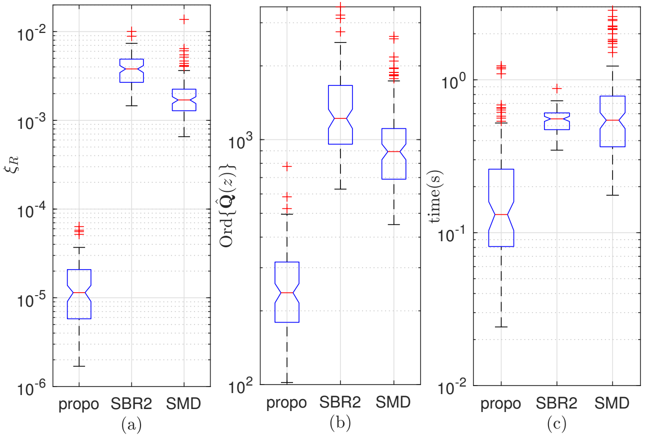
Polynomial Power Method: Dominant Eigenpair Extraction

Similar to ordinary power method, with $\boldsymbol{x}^{(0)}(z) = [1, 0]^{\mathrm{T}}$, the normalized product vector $\boldsymbol{x}^{(k)}_{\mathrm{norm}}(z)$ at $z = \mathrm{e}^{\mathrm{j}\pi}$ is shown to converge to $\boldsymbol{q}_1(\mathrm{e}^{\mathrm{j}\pi}) = 1/\sqrt{(2)}[1, -1]^{\mathrm{T}}$



Polynomial Power Method: Combined with Deflation

PPM method combined with deflation performance on $\boldsymbol{R}(z) = \boldsymbol{H}(z)\boldsymbol{H}^{\mathrm{P}}(z)$ where $\boldsymbol{H}(z) \in 6 \times 2$ versus SBR2[5] and SMD [6]



Box plot illustration ensemble results: (a) reconstruction error, (b) order of decomposition (c) execution time.

Power Method For Ordinary Non-Hermitian Matrices

- Lets assume a non-Hermitian matrix $\mathbf{A} \in \mathbf{C}^{M \times N}, M \geq N$ with singular values $|\sigma_i| > |\sigma_{i+1}|,\ i = 1, \ldots, (N-1)$.
- Power method can be applied to $\mathbf{A}^{\mathrm{H}}\mathbf{A}$, which is Hermitian matrices, to estimate its dominant eigenvector which will be the dominant right-singular vector.
- There are two ways to determine the singular value and the left-singular vector as:
  1. singular value and the left-singular vector is obtained as $\hat{\sigma}_1 = ||\mathbf{A}\hat{\mathbf{v}}_1||_2$ and $\hat{\mathbf{u}}_1 = \mathbf{A}\hat{\mathbf{v}}_1/\hat{\sigma}_1$.
  2. apply power iteration to $\mathbf{A}\mathbf{A}^{\mathrm{H}}$ to obtain $\hat{\mathbf{u}}_1$ and obtain $\hat{\sigma}_1 = \hat{\mathbf{u}}_1^{\mathrm{H}}\mathbf{A}\hat{\mathbf{v}}_1$.
- Second method produces singular vectors which are not phase coupled with makes the singular value complex.

Generalized Polynomial Power Method

- PPM can be applied to $\boldsymbol{A}^{\mathrm{P}}(z)\boldsymbol{A}(z)$ to estimate $\hat{\boldsymbol{v}}_1(z)$
- Singular value can be estimated through a DFT domain method $\hat{\sigma}_1(\mathrm{e}^{\mathrm{j}\Omega}) = ||\boldsymbol{A}(\mathrm{e}^{\mathrm{j}\Omega})\hat{\boldsymbol{v}}_1(\mathrm{e}^{\mathrm{j}\Omega})||_2$
- Singular value is estimated at sufficiently large DFT size at which the time-domain aliasing is minimum

$$\zeta_{\hat{\sigma}} = \sum_{\tau} \frac{|\hat{\sigma}_1^{(K)}[\tau] - \hat{\sigma}_1^{(K/2)}[\tau]|^2}{|\hat{\sigma}_1^{(K)}[\tau]|^2} \text{ where } \hat{\sigma}_1^{(K)}[\tau] \text{ is obtained via } K-\text{IDFT} \quad (1)$$

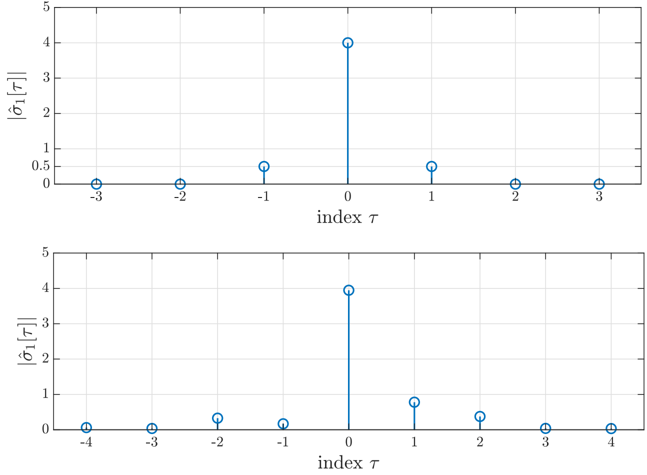
- Left-singular vector is obtained via DFT domain method $\hat{\boldsymbol{u}}_1(\mathrm{e}^{\mathrm{j}\Omega}) = \hat{\boldsymbol{A}}(\mathrm{e}^{\mathrm{j}\Omega})\hat{\boldsymbol{v}}_1(\mathrm{e}^{\mathrm{j}\Omega})/\hat{\sigma}_1(\mathrm{e}^{\mathrm{j}\Omega})$ at DFT size at which

$$\xi_u = \sum_{\tau} |\hat{\mathbf{u}}_1^{\mathrm{H}}[-\tau] * \hat{\mathbf{u}}_1[\tau] - \delta[\tau]|_2^2, \quad \tau \in \mathbb{Z} \quad (2)$$

is minimum.

Simulation: Numerical Example

Polynomial matrix with ground-truth $\sigma_1(z) = z/2 + 4 + z^{-1}/2$, $\sigma_2(z) = z/4 + 1 + z^{-1}/4$



Generalized polynomial power method versus generalised SBR2 performance on an example polynomial matrix: illustrating the time-domain coefficients for the estimated dominant singular value.

Simulation: Ensemble Setting

Proposed method is compared with GSBR2 over an ensemble of 500 randomized polynomial matrices $\boldsymbol{A}(z) \in \mathbb{C}^{3\times 2}$ where the dominant singular value spectrally majorises the second. Following performance metrics are selected:

- order of resulting decomposition elements $\hat{\boldsymbol{U}}(z), \hat{\boldsymbol{V}}(z), \hat{\boldsymbol{\Sigma}}(z)$.
- average execution time.
- accuracy of the estimated singular value $\xi_\sigma = \frac{\sum_\tau |\sigma_1[\tau] - \hat{\sigma}_1[\tau]|^2}{\sum_\tau |\sigma_1[\tau]|^2}$

Simulation: Ensemble Results

TABLE I
PERFORMANCE COMPARISON OF GSBR2 AND GPPM

Metrics	GSBR2	GPPM
$\mathcal{O}\{\hat{\boldsymbol{u}}_1(z)\}$	966 ± 185	10
$\mathcal{O}\{\hat{\boldsymbol{v}}_1(z)\}$	422 ± 126	10
$\mathcal{O}\{\hat{\sigma}_1(z)\}$	96 ± 38	57 ± 4
ξ_v	$(1.2 \pm 0.8) \times 10^{-3}$	$(5.5 \pm 4.5) \times 10^{-5}$
ξ_u	$(1.6 \pm 0.85) \times 10^{-3}$	$(5.5 \pm 4.5) \times 10^{-5}$
ξ_σ	0.09 ± 0.07	$(1.5 \pm 1.3) \times 10^{-5}$
time(s)	0.67 ± 0.15	0.44 ± 0.19

Conclusion

- Polynomial power method is extended to general polynomial matrices for dominant singular value and singular vectors extraction.
- The proposed extension assumes the polynomial matrix to be positive semi-definite and spectrally majorised.
- Unlike already developed PSVD algorithms, GPPM provides real singular values on the unit circle.
- Deflation approach can similarly be applied for computing reduced-rank PSVD.

References I

[1] Z. Wang, A. Sandmann, J. G. McWhirter, and A. Ahrens, "Multiple shift SBR2 algorithm for calculating the SVD of broadband optical MIMO systems," in *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*, 2016, pp. 433–436.

[2] J. A. Foster, J. G. McWhirter, M. R. Davies, and J. A. Chambers, "An algorithm for calculating the QR and singular value decompositions of polynomial matrices," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1263–1274, 2010.

[3] J. G. McWhirter, "An algorithm for polynomial matrix SVD based on generalised kogbetliantz transformations," in *2010 18th European Signal Processing Conference*, 2010, pp. 457–461.

[4] F. A. Khattak, I. K. Proudler, and S. Weiss, "Extension of power method to para-hermitian matrices: Polynomial power method," in *2023 European Signal Processing Conference (EUSIPCO)*, accepted.

[5] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An evd algorithm for para-hermitian polynomial matrices," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2158–2169, 2007.

[6] S. Redif, S. Weiss, and J. G. McWhirter, "Sequential matrix diagonalization algorithms for polynomial evd of parahermitian matrices," *IEEE Transactions on Signal Processing*, vol. 63, no. 1, pp. 81–89, 2015.