Corr, Jamie and Thompson, Keith and Weiss, Stephan and McWhirter, John G. and Proudler, Ian K. (2015) Performance trade-offs in sequential matrix diagonalisation search strategies. In: 2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP). IEEE, Piscataway, NJ., pp. 25-28. ISBN 9781479919635 , http://dx.doi.org/10.1109/CAMSAP.2015.7383727

Performance Trade-Offs in Sequential Matrix Diagonalisation Search Strategies

Jamie Corr*, Keith Thompson*, Stephan Weiss*, John G. McWhirter†, Ian K. Proudler‡
*Department of Electronic & Electrical Engineering, University of Strathclyde, Glasgow, Scotland
†School of Engineering, Cardiff University, Cardiff, Wales, UK
‡School of Electrical, Electronics and Systems Engineering, Loughborough University, Loughborough, UK
{jamie.corr,keith.thompson,stephan.weiss}@strath.ac.uk

***Abstract*—Recently a selection of sequential matrix diagonalisation (SMD) algorithms have been introduced which approximate polynomial eigenvalue decomposition of parahermitian matrices. These variants differ only in the search methods that are used to bring energy onto the zero-lag. Here we analyse the search methods in terms of their computational complexities for different sizes of parahermitian matrices which are verified through simulated execution times. Another important factor for these search methods is their ability to transfer energy. Simulations show that the more computationally complex search methods transfer a greater proportion of the off-diagonal energy onto the zero-lag over a selected range of parahermitian matrix sizes. Finally we compare the real time convergence of the search methods as part of their respective SMD algorithms. The real time convergence experiments indicate that despite taking a longer time to compute each iteration the more complex algorithms that transfer more energy converge faster in real time.**

I. Introduction

Sequential matrix diagonalisation encompasses a family of iterative algorithms that can factorise a parahermitian matrix into an approximate polynomial matrix eigenvalue decomposition (PEVD). The PEVD extends the wide-ranging utility of the EVD from narrowband to broadband problems, and iterative PEVD algorithms have in the past found use in optimal subband [1] and multichannel coding [2]; channel coding [3], transmit and receive beamforming across broadband MIMO channel [4], [5], angle of arrival estimation [6]. It can also provide a preprocessing stage for beamforming by applying denoising [7], decorrelation [8] and optimum subband decompositions [9], or enable novel MVDR beamforming approaches [10].

Parahermitian matrices arise e.g. by including an explicit lag τ into the space-time covariance $\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n-\tau]\}$. The matrix elements are auto- and cross-correlation sequences that create a symmetry, $\mathbf{R}[\tau] = \mathbf{R}^{\mathrm{H}}[-\tau]$, i.e. a parahermitian matrix is equal to its complex conjugate, time reversed version. For its z-transform, the cross spectral density matrix $\boldsymbol{R}(z) \bullet\!\!-\!\!\circ \mathbf{R}[\tau]$, the parahermitian property is expressed as $\boldsymbol{R}(z) = \tilde{\boldsymbol{R}}(z)$, where the parahermitian operator $\widetilde{\{\cdot\}}$ implies complex conjugation and time reversal. A polynomial EVD [11]–[13] of such a parahermitian matrix,

$$\boldsymbol{\Lambda}(z) \approx \boldsymbol{Q}(z)\boldsymbol{R}(z)\tilde{\boldsymbol{Q}}(z) \quad , \tag{1}$$

is claimed to exist in close approximation for FIR paraunitary matrices $\boldsymbol{Q}(z)$ of sufficiently high order [14].

A number of iterative PEVD algorithms have been introduced, including second order sequential best rotation (SBR2) methods [1], [13], which eliminate the maximum off-diagonal element at every iteration. An approximate PEVD [15] operates on a fixed order paraunitary matrix, but unlike SBR2 has not been proven to converge. More recently, a sequential matrix diagonalisation (SMD) algorithm has been introduced [2], which is also proven to converge but seems capable of attaining better diagonalisation of the space-time covariance matrix $\boldsymbol{R}(z)$ than SBR2 algorithms [1], [13]. The SMD family has been extended by a multiple-shift version, which has been found to transfer even more energy per iteration [16], with additional searches the only cost increase over SMD. Therefore, the aim of this paper is to explore some of the performance trade-off details w.r.t. computational cost and diagonalisation, between the different SMD algorithms.

The paper is organised as follows. Iterative PEVD algorithms based on the idea of sequential matrix diagonalisation are introduced in Sec. II. The difference in the SMD search steps and their associated cost is investigated in Sec. III. Finally, performance metrics and simulation results are presented in Sec. IV followed by conclusions in Sec. V.

II. Sequential Matrix Diagonalisation

The sequential matrix diagonalisation algorithm (SMD) [2] and a number of derivative versions [16]–[18] iteratively diagonalise a parahermitian matrix to approximate its PEVD. The initialisation step of any SMD algorithm fully diagonalises the zero-lag of the parahermitian matrix, $\mathbf{R}[0]$, achieved via the modal matrix $\mathbf{Q}^{(0)}$ of the ordered EVD of $\mathbf{R}[0]$, which is applied to all lags of the parahermitian matrix,

$$\boldsymbol{S}^{(0)}(z) = \mathbf{Q}^{(0)}\boldsymbol{R}(z)\mathbf{Q}^{(0)\mathrm{H}} \quad . \tag{2}$$

Each iteration of the SMD algorithm includes a shift operation which brings off-diagonal energy onto the zero-lag,

$$\boldsymbol{S}^{(i)\prime}(z) = \boldsymbol{\Lambda}^{(i)}(z)\boldsymbol{S}^{(i-1)}(z)\tilde{\boldsymbol{\Lambda}}^{(i)}(z) \; , \quad i = 1 \ldots I \; . \tag{3}$$

The shift matrix, $\boldsymbol{\Lambda}^{(i)}(z)$, is determined by the search strategy which varies between SMD versions and will be discussed further in Sec III.

An SMD iteration is then completed by transferring energy from the zero-lag on to the diagonal, like the initialisation step, this consists of applying the EVD modal matrix to the entire parahermitian matrix,

$$\boldsymbol{S}^{(i)}(z) = \mathbf{Q}^{(i)}\boldsymbol{S}^{(i)\prime}(z)\mathbf{Q}^{(i)\mathrm{H}} \quad . \tag{4}$$

This diagonalises [2], [16]–[18] the zero-lag matrix $\mathbf{S}^{(i)}[0]$.

The SMD algorithms repeat steps (3) & (4) for either a set number of iterations or until some threshold based on the entries of the parahermitian matrix is reached (e.g. max off-diagonal element). Thus after a total of I iterations, the paraunitary matrix which approximately diagonalises $\boldsymbol{R}(z)$ is obtained by the product of the matrices produced by the initialisation and steps (3) & (4) from each iteration i.e.

$$\hat{\boldsymbol{Q}}(z) = \boldsymbol{G}^{(I)}(z)\ldots\boldsymbol{G}^{(1)}(z)\boldsymbol{G}^{(0)}(z) \quad , \tag{5}$$

where each $\boldsymbol{G}^{(i)}(z)$ is constructed from the delay and energy transfer matrices from the ith step i.e.

$$\boldsymbol{G}^{(i)}(z) = \mathbf{Q}^{(i)}\boldsymbol{\Lambda}^{(i)}(z) \quad . \tag{6}$$

This approximates a PEVD with

$$\boldsymbol{S}^{(I)}(z) = \hat{\boldsymbol{Q}}(z)\boldsymbol{R}(z)\tilde{\hat{\boldsymbol{Q}}}(z) \quad . \tag{7}$$

Convergence proofs for the various SMD algorithms show that for a sufficiently high I, the off-diagonal energy in $\boldsymbol{S}^{(I)}(z)$ can be reduced to an arbitrarily low bound. Spectral majorisation of $\boldsymbol{S}^{(I)}(z)$ cannot be guaranteed, but is encouraged through appropriate ordering in step (4).

III. Search Methods

This section discusses how SMD algorithms identify the elements to be transferred in the first SMD step outlined in Sec. II. We mainly consider the order $\mathcal{O}(\cdot)$ of the computational complexity in terms of multiply-accumulate operations, which depends on the matrix dimension M and the lag dimension L, whereby it must be noted that L grows with every iteration [2], [13] and the extent of the growth varies with both algorithm and input parahermitian matrix $\boldsymbol{R}(z)$ [19].

A. Column Norm / SMD-Algorithm

The original SMD algorithm [2] in its ith iteration inspects the vectors $\hat{\mathbf{s}}_k^{(i-1)}[\tau]$, which are the columns of $\mathbf{S}^{(i-1)}[\tau]$ but modified by removing its on-diagonal elements. The set

$$\{k^{(i)}, \tau^{(i)}\} = \arg\max_{k,\tau} \|\hat{\mathbf{s}}_k^{(i-1)}[\tau]\|_2 \ , \quad i = 1\ldots I \ , \tag{8}$$

identifies the vector $\hat{\mathbf{s}}_k^{(i-1)}[\tau]$ with maximum norm, which is transferred onto the zero-lag matrix and subsequently eliminated. A total of ML column norms of the parahermitian matrix have to be calculated. Each norm requires a squaring of elements, but the square root operation can be omitted as only a comparison of norms but no explicit values are required. Thus, with each column vector having length M, the norm computation is $\mathcal{O}(M^2L)$ followed by a search over $\mathcal{O}(ML)$ elements.

TABLE I
Cost comparsion of SMD search methods.

method	norm calc.	comparisons	total
SMD	$\mathcal{O}(M^2L)$	$\mathcal{O}(ML)$	$\mathcal{O}(M^2L)$
ME-SMD	$\mathcal{O}(0)$	$\mathcal{O}(M^2L)$	$\mathcal{O}(M^2L)$
MSME-SMD	$\mathcal{O}(0)$	$\mathcal{O}(M^3L)$	$\mathcal{O}(M^3L)$

B. Maximum Element / ME-SMD Algorithm

Introduced as a simplification to the SMD search in Sec. III-A, the maximum-element SMD (ME-SMD) algorithm [2] replaces the l_2 norm in (8) by the l_∞ norm. Thus in each iteration, the maximum element can be identified without any explicit norm calculation but requires a search over an enlarged set of $\mathcal{O}(M^2L)$ elements. The energy that is transferred in a single step by ME-SMD is always smaller or equal to that eliminated by the original SMD version, but the algorithm was designed with the expectation of a lower computational complexity.

C. Multiple Shift Maximum Element / MSME-SMD Algorithm

The multiple shift maximum element (MSME) search method, used in the MSME-SMD algorithm [16], initiates every iteration by scanning the entire parahermitian matrix for its maximum off-diagonal element similar to ME-SMD, employing the l_∞ instead of the l_2 norm in (8). However, MSME-SMD does not transfer just one-column into the zero-lag, but will perform a total of $(M-1)$ column shifts to increase the energy transfer in the second step of each iteration. This is achieved by $(M-1)$ searches over increasingly limited search spaces such that previously identified and shifted maxima are not undone by later shifts [16].

This approach requires no norm evaluations but the complexity of the search is $\mathcal{O}(M^3L)$ because each iteration involves searching M^2L elements a total of $M-1$ times, for asymptotic analysis $M-1$ is simplified to M. An overall cost comparison of the three search methods is provided in Tab. I, with a total search cost order provided on the basis that one comparison for the maximum search is about as expensive as one multiply-accumulate operation.

IV. Results

A. Simulation Set-Up and Performance Metrics

To assess the proposed search algorithms, we consider an ensemble of 10^3 random $M \times M$ parahermitian matrices $\boldsymbol{R}(z)$ of order $2L-1$, for $M = 2, 4, \ldots 20$, & $L = 50, 100, \ldots 500$. Each instance of $\boldsymbol{R}(z)$ is generated as $\boldsymbol{R}(z) = \boldsymbol{A}(z)\tilde{\boldsymbol{A}}(z)$, where $\boldsymbol{A}(z) \in \mathbb{C}^{M\times M}$ is a random polynomial matrix of order L with independent and identically distributed zero mean and unit variance complex Gaussian entries.

Execution time is used to measure the computational complexity of the search methods in Matlab 2014a with the following system specification: Ubuntu 14.04 on a Dell Precision T3610 with Intel® Xeon® E5-1607V2 3.00 GHz x 4 cores and 8 GB RAM.

In addition to execution time we have also investigated the proportion of the off-diagonal energy that is brought onto the zero-lag by each of these search algorithms for the same ensemble but M is restricted to $4, 10$ & 20. The proportion of shifted energy, $E_{\text{shift}}^{(m,l)}$, is averaged over the ensemble and calculated as

$$E_{\text{shift}}^{(m,l)} = \frac{\sum_{k=1}^{M} \|\hat{\mathbf{s}}_k^{(m,l)'}[0]\|_2^2}{\sum_{\tau} \sum_{k=1}^{M} \|\hat{\mathbf{s}}_k^{(m,l)}[\tau]\|_2^2} \quad , \tag{9}$$

where $\hat{\mathbf{s}}_k^{(m,l)}[\tau]$ is the modified column vector from (8). The numerator in (9) is the the off-diagonal energy brought onto the zero-lag and the denominator is the off-diagonal energy in the entire parahermitian matrix. The algorithm that shifts most energy onto the zero-lag consequently produces the highest $E_{\text{shift}}^{(m,l)}$.

The final test measures diagonalisation, the remaining normalised off-diagonal energy after i iterations,

$$E_{\text{norm}}^{(i)} = \frac{\sum_{\tau} \sum_{k=1}^{M} \|\hat{\mathbf{s}}_k^{(i)}[\tau]\|_2^2}{\sum_{\tau} \|\mathbf{R}[\tau]\|_{\text{F}}^2} \quad , \tag{10}$$

where $\mathbf{R}[\tau]$ is the initial parahermitian matrix and $\|\cdot\|_{\text{F}}$ the Frobenius norm. Unlike $E_{\text{shift}}^{(m,l)}$, the value for $E_{\text{norm}}^{(i)}$ should ideally be minimised. The matrix dimension M is restricted as above and the initial L is 6.

B. Real Time Complexity

The $\mathcal{O}(\cdot)$ notation essentially only shows the shape of the time complexity of these search methods. It is also good to show the real time complexity as there could be hidden constants that dramatically affect real time performance [20], and it is difficult to relate the complexity order for MAC operations required for norm calculations with comparisons required for the maximum search.

The real time complexity for the column norm search is given in Fig. 1; this agrees with the complexity analysis of $\mathcal{O}(M^2L)$ with the linear increase with L and shallow but polynomial increase with M. Although thought of as low cost, the real time performance of the maximum element search in Fig. 2 is very similar to that of the column norm approach both with the trends shown with matrix dimensions and the real time performance. Fig. 3 shows the the real time performance of the MSME search. The linear increase with number of lags and quite a steep polynomial increase with the matrix dimension agrees with the complexity analysis of $\mathcal{O}(M^3L)$. Comparing Figs. 1, 2 & 3 it is clear to see that the MSME search is significantly slower for larger matrix sizes than the other two.

C. Energy Transfer

Another important metric for these search algorithms is the amount of energy they bring onto the zero-lag at each iteration. This section will investigate how this varies with matrix dimensions for the various search methods.

The energy transfer for the column norm, maximum element and MSME methods are shown in Fig. 4. The most striking

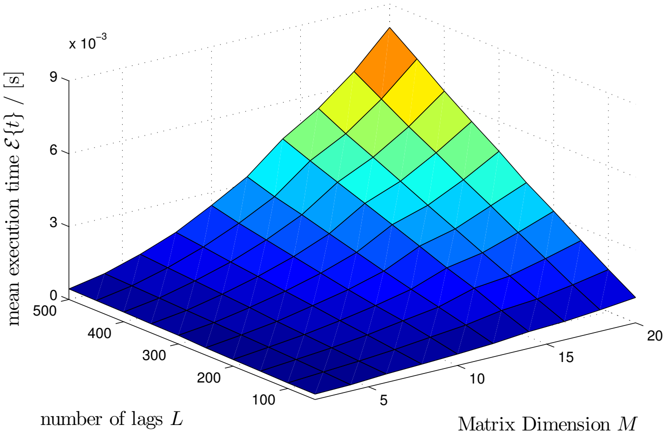

Fig. 1. Column Norm search time for varying matrix size.

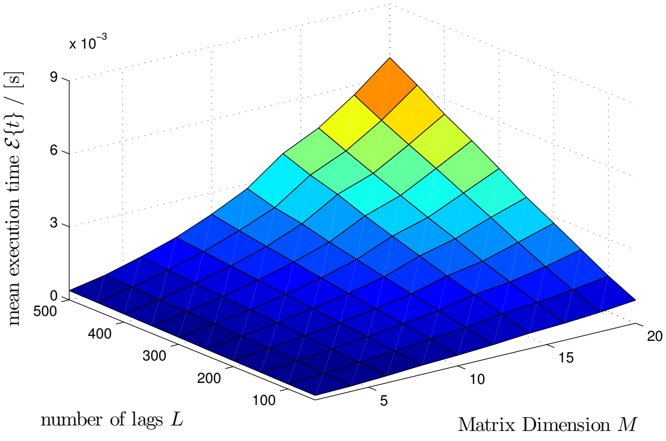

Fig. 2. Maximum Element search time for varying matrix size.

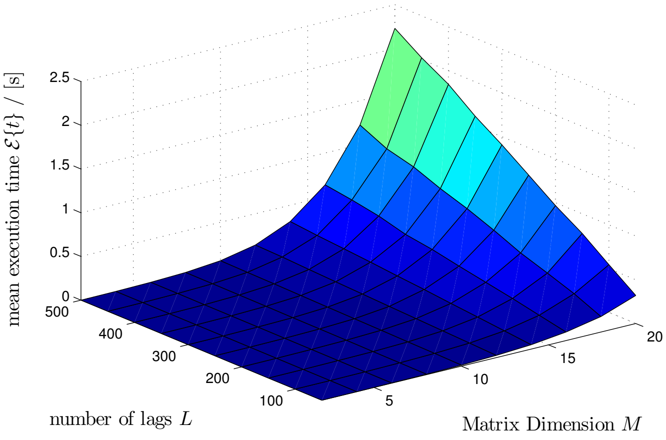

Fig. 3. MSME search time for varying matrix size.

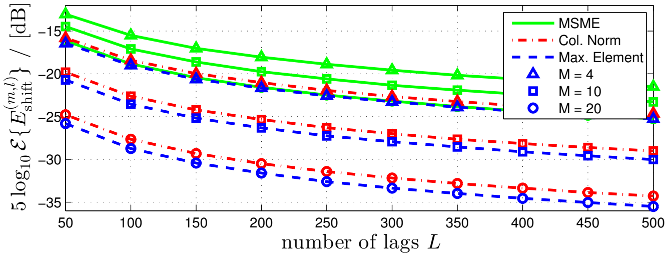

Fig. 4. Search energies for varying matrix size and search algorithm.

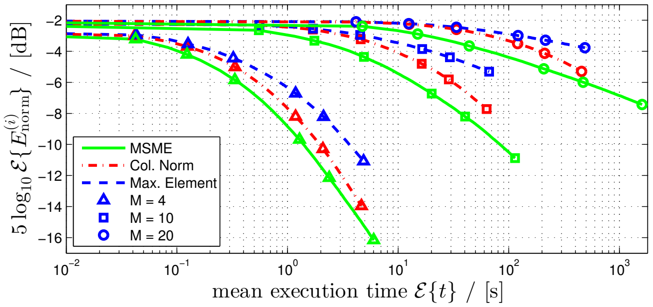

Fig. 5. Reduction in off-diagonal energy vs. mean execution time over 100 algorithm iterations.

difference is that for the MSME approach the energy transfer does not degrade as dramatically with the matrix dimension M, this is because as M increases the number of elements brought onto the zero-lag also increases. Comparing only the column norm and maximum element searches we see that the overall trends are very similar and that they degrade at a similar rate. Crucially, overall the column norm approach does indeed tend to transfer more energy than the maximum element method.

D. Real Time Convergence

While the previous two sections have focussed only on the search step of the SMD algorithms, here we show how they converge in real time over $I = 100$ iterations of each algorithm. Fig. 5 shows a real time convergence example for when $M = 4, 10 \ \& \ 20$, here we can see that despite its higher computational cost the fastest converging algorithm is the MSME-based implementation and the column norm approach converges faster than the algorithm using the maximum element search.

V. Conclusion

The complexities and energy transfer associated with three of the search algorithms used in the SMD family of PEVD algorithms have been investigated in detail. The complexities range from the simplistic maximum element search right up to the more complex multiple shift maximum element search. The MSME search tends to obtain the greatest amount of energy at any iteration however the multiple shifts cause its complexity to rise significantly with matrix dimensions. The maximum element and column norm searches have a similarly low complexity however this comes at the cost of lower energy transfer. From the results presented the maximum element version does not appear to have any significant benefit over the column norm based search however the column norm search will generally bring more energy onto the zero-lag and hence converge faster. Despite it's significantly higher cost, the MSME search approach has been shown to converge faster in real time than the others for the experiments shown.

References

[1] S. Redif, J. McWhirter, and S. Weiss. Design of FIR paraunitary filter banks for subband coding using a polynomial eigenvalue decomposition. *IEEE Transactions on Signal Processing*, 59(11):5253–5264, November 2011.

[2] S. Redif, S. Weiss, and J. McWhirter. Sequential matrix diagonalization algorithms for polynomial EVD of parahermitian matrices. *IEEE Transactions on Signal Processing*, 63(1):81–89, January 2015.

[3] S. Weiss, C. H. Ta, and C. Liu. A Wiener filter approach to the design of filter bank based single-carrier precoding and equalisation. In *IEEE International Symposium on Power Line Communications and Its Applications*, pages 493–498, Pisa, Italy, March 26-28 2007.

[4] M. Davies, S. Lambotharan, and J. McWhirter. Broadband MIMO beamforming using spatial-temporal filters and polynomial matrix decomposition. In *15th International Conference on Digital Signal Processing*, pages 579–582, Cardiff, UK, July 2007.

[5] C. H. Ta and S. Weiss. A Design of Precoding and Equalisation for Broadband MIMO Systems. In *15th International Conference on Digital Signal Processing*, pages 571–574, Cardiff, UK, July 2007.

[6] M. Alrmah, S. Weiss, and S. Lambotharan. An extension of the MUSIC algorithm to broadband scenarios using polynomial eigenvalue decomposition. In *19th European Signal Processing Conference*, pages 629–633, Barcelona, Spain, August 2011.

[7] S. Redif, J. McWhirter, P. Baxter, and T. Cooper. Robust broadband adaptive beamforming via polynomial eigenvalues. In *OCEANS*, pages 1–6, Boston, MA, September 2006.

[8] C. L. Koh, S. Redif, and S. Weiss. Broadband GSC beamformer with spatial and temporal decorrelation. In *17th European Signal Processing Conference*, pages 889–893, Glasgow, Scotland, August 2009.

[9] P. Vouras and T. Tran. Robust transmit nulling in wideband arrays. *IEEE Transactions on Signal Processing*, 62(14):3706–3719, July 2014.

[10] S. Weiss, S. Bendoukha, A. Alzin, F. Coutts, I. Proudler, and J. Chambers. MVDR broadband beamforming using polynomial matrix techniques. In *European Signal Processing Conference*, Nice, France, September 2015.

[11] I. Gohberg, P. Lancaster, and L. Rodman. *Matrix Polynomials*. Academic Press, New York, 1982.

[12] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, 1993.

[13] 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*, 55(5):2158–2169, May 2007.

[14] S. Icart and P. Comon. Some properties of Laurent polynomial matrices. In *9th IMA Conference on Mathematics in Signal Processing*, Birmingham, UK, December 2012.

[15] A. Tkacenko. Approximate eigenvalue decomposition of para-hermitian systems through successive FIR paraunitary transformations. In *IEEE International Conference on Acoustics Speech and Signal Processing*, pages 4074–4077, Dallas, TX, Mar. 2010.

[16] J. Corr, K. Thompson, S. Weiss, J. McWhirter, S. Redif, and I. Proudler. Multiple shift maximum element sequential matrix diagonalisation for parahermitian matrices. In *IEEE Workshop on Statistical Signal Processing*, pages 312–315, Gold Coast, Australia, June 2014.

[17] J. Corr, K. Thompson, S. Weiss, J. McWhirter, and I. Proudler. Maximum energy sequential matrix diagonalisation for parahermitian matrices. In *48th Asilomar Conference on Signals, Systems and Computers*, pages 470–474, Pacific Grove, CA, USA, November 2014.

[18] J. Corr, K. Thompson, S. Weiss, J. G. McWhirter, and I. K. Proudler. Causality-Constrained multiple shift sequential matrix diagonalisation for parahermitian matrices. In *22nd European Signal Processing Conference*, pages 1277–1281, Lisbon, Portugal, September 2014.

[19] J. Corr, K. Thompson, S. Weiss, I. K. Proudler, and J. G. McWhirter. Impact of source model matrix conditioning on PEVD algorithms. In *2nd IET Conference on Intelligent Signal Processing*, London, England, December 2015.

[20] A. Aho, and J. Ullman. *Foundations of computer science*. W.H. Freeman, New York, 1994.