University Defence Research Collaboration (UDRC)
Signal Processing in a Networked Battlespace

L_WP5: Low Complexity Algorithms and Efficient Implementation

WP Leaders: Stephan Weiss (University of Strathclyde), Ian Proudler (Loughborough University)
Researcher: K.Thompson (University of Strathclyde) PhD Student: J. Corr (University of Strathclyde)

Brief Summary of Last 3 Months (Oct – Dec 2013)

- Currently investigating the efficient implementation of the SBR2 algorithm [1] and new Sequential Matrix Diagonalisation (SMD).
- FPGA Development Training - recently completed going through the supplied Xilinx FPGA Primer training material
- Reviewing field of Graphical Models (esp. Bayesian Networks) for use in distributed processing applications. (e.g. distributed beamforming)
- Delivered a lunchtime seminar to Strathclyde's CeSIP research group on using Gaussian Process (GP) models for Bayesian probabilistic modelling.
- Established links with counterpart WP6 in Edinburgh/Heriot-Watt Consortium (follow-up on recommendation from [dstl])
- Attended UDRC Theme Day on Source Separation (Edinburgh 31st Oct)
- Attended ISP conference & UDRC II Launch (London 2-4th Dec)
- Attended Academic Symposium 2013 at Texas Instruments (Freiburg 4-5th Nov)

WP5.1 Data Reduction – Efficient Implementation of PEVD Methods (SBR2, SMD)

- Matlab Toolbox of optimised software code is in development (code optimised through using 'Profile' feature to find bottlenecks).
- Both SBR2 and SMD algorithms minimise the energy placed in off-diagonal elements. Maximum Element SMD (MESMD) (lower computational cost version) has been implemented using absolute value (not column norms). MESMD completes diagonalisation at faster rate than SBR2.

$$\text{Measure of Diagonalisation} = \frac{\sum(\|\text{off-diag elements}\|^2)}{\sum(\|\text{all elements}\|^2)}$$

- Approximate methods for SBR2 exploit the growing sparsity of the polynomial matrix upon repeated shift and rotation operations (zero-padding). These 'trim' methods are aimed at limiting the growth of the overall matrix to trade-off efficiency against accuracy.

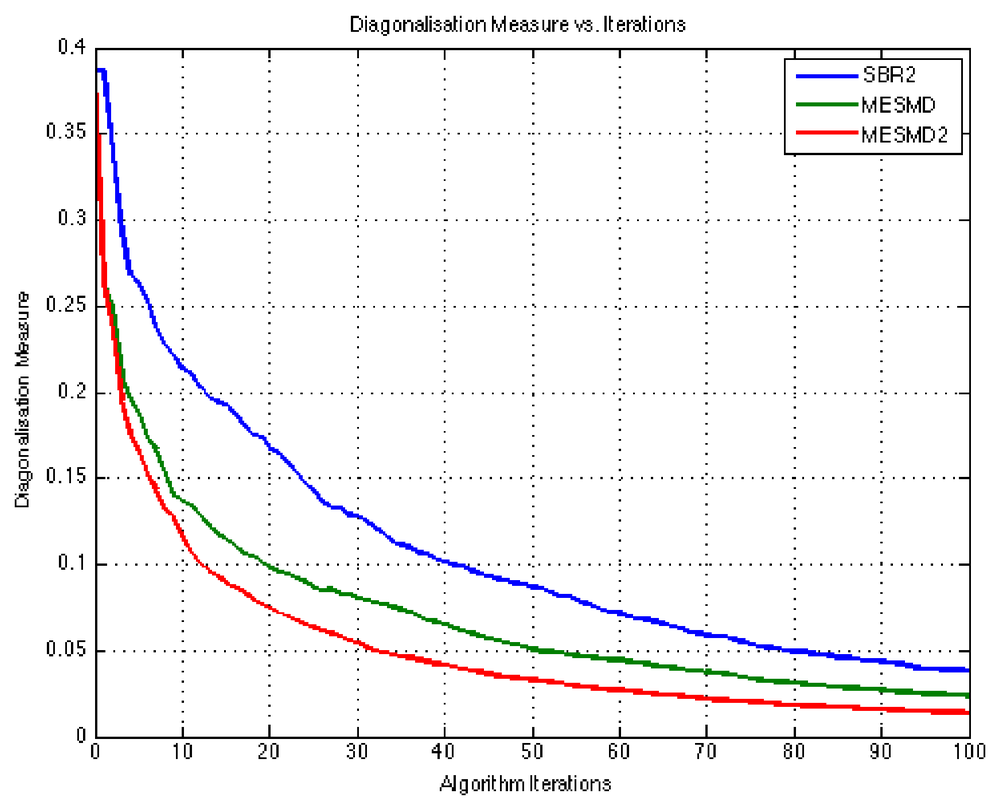
WP5.2 Hardware Realisations

Using implementation of SBR2 algorithms on FPGA hardware as a case study, see previous implementation [2]. Latest FPGA Hardware Development tools assessed, especially new auto-coding options for more efficient system design flow. Options:

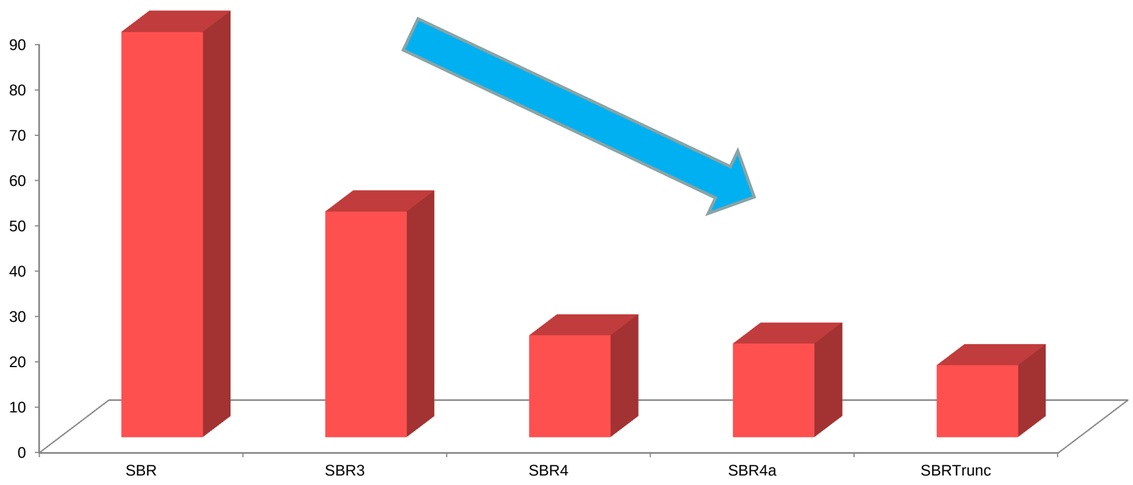
- Manual coding of VHDL/Verilog (Slow, but flexible)
- ISE (Xilinx) allowed Simulink design creation using embedded Matlab function blocks for auto-VHDL generation
- Mathworks – Matlab Coder (C auto-code) ,HDL Coder (VHDL auto-code), Fixed Point Designer (floating to fixed-point adaptation)
- C to VHDL tools – Xilinx Vivado, Mentor Graphics Catapult C
- C/C++ code also used as basis for GPU coding (OpenCL/CUDA) and DSP (TI Code Composer) – thus allowing performance comparisons.

Overall Strategy:
1) Optimised Matlab Toolboxes (use MEX (C) subroutines)
2) Optimised C code through Matlab Coder & Manual Tweaking
3) Use Matlab & C to produce VHDL, CUDA/OpenCL

SBR2 vs. MESMD



SBR2 Execution Time



SBR2 Implementation	Modification
SBR	Original code
SBR3	Only Search 1st half of the PH matrix
SBR4	Optimised column-wise masking algorithm
SBR4a	Modified shifting algorithm
SBRTrunc	Truncation of unnecessary fully zero lags

Future Activities

- Complete optimisation of SBR2 and SMD algorithms in Matlab, transfer to C/C++, build hardware implementation.
- Integration of SBR2 with 'oversampled sub-band decomposition' to retain coherence between sub-bands but reduce spectral dynamics.
- Organise Mathworks meeting (Professor Robert Stewart) – welcome any requests/recommendations for toolboxes?
- With C.Clemente develop probabilistic classifier for WP4 Micro-Doppler data. Contrast performance with current SVM-based approach.
- Develop Bayesian Network matlab examples – use as case studies for GPU implementation.
- Organise Show & Tell Event for 9th April 2014 – at Strathclyde CMT

References

[1] J G McWhirter, P Baxter, T Cooper, S Redif and J Foster. An EVD Algorithm for Para-Hermitian Polynomial Matrices. IEEE Trans Signal Processing, Vol 55, No 6 (May 2007).
[2] Kasap S, Redif S, Novel Field-Programmable Gate Array Architecture for Computing the Eigenvalue Decomposition of Para-Hermitian Polynomial Matrices, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, (2013)

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) and Dstl