Speeding up Magnetic Resonance Image Acquisition by Bayesian Multi-Slice Adaptive Compressed Sensing — Supplemental Appendix

Matthias W. Seeger

Saarland University and Max Planck Institute for Informatics
Campus E1.4, 66123 Saarbrücken, Germany
mseeger@mmci.uni-saarland.de

This note contains supplemental material for [3].

Approximating Messages and Covariances

On the scale of interest here, Kalman smoothing discussed in [3, Sect. 3] cannot be implemented exactly. Readers familiar with approximate computation in Gaussian (Markov) random fields will appreciate the difficulties here: while $\boldsymbol{X}$ and $\boldsymbol{B}$ are structured, so that matrix-vector multiplications (MVMs) can be computed rapidly, the overall model does not have locally connected MRF structure[1]. Moreover, our problem does not come with a single static Gaussian MRF with close to stationary potentials, but with a sequence of $Q(\boldsymbol{u}|\boldsymbol{y})$ fitted to a posterior with significantly non-Gaussian statistics (edges in the image, *etc*).

In this section, we show how message passing and marginal covariance computations can be approximated by the Lanczos algorithm [1]. The underlying idea is PCA: if we can efficiently determine the $l \ll \tilde{n}$ smallest eigenvalues $\boldsymbol{\Lambda}$ and eigenvectors $\boldsymbol{U}$ of a precision matrix $\boldsymbol{A}$, the PCA approximation of $\boldsymbol{A}^{-1}$ is $\boldsymbol{U}\boldsymbol{\Lambda}^{-1}\boldsymbol{U}^T$, capturing most of the covariance across all matrices of rank l. Matrices $\boldsymbol{A}$ encountered here have close to linear spectral decay, so the Lanczos method can be used to approximate PCA [3, Sect. 5.1]. At the cost of k MVMs with $\boldsymbol{A}$, this algorithm results in $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{T}\boldsymbol{Q}^T$, $\boldsymbol{Q} \in \mathbb{R}^{\tilde{n} \times k}$ orthonormal (where $\tilde{n} = 2n$ for complex-valued data), $\boldsymbol{T} \in \mathbb{R}^{k \times k}$ tridiagonal, with extremal eigenvalues (largest and smallest) close to eigenvalues of $\boldsymbol{T}$. Corresponding approximate eigenvectors (Ritz vectors) can be determined from $\boldsymbol{Q}$ and an eigendecomposition of $\boldsymbol{T}$. Lanczos approximations $\boldsymbol{A}^{-1} \approx \boldsymbol{Q}\boldsymbol{T}^{-1}\boldsymbol{Q}^T$, $\log|\boldsymbol{A}| \approx \log|\boldsymbol{T}|$ are therefore closely related to PCA. For Lanczos, eigenvectors from both ends of the spectrum are used. Empirically, k is a small multiple of l. If desired, eigenvalue convergence can be tested within the Lanczos method, which could be stopped once a desired number of smallest eigenvalues have been obtained. In experiments presented here, we instead run Lanczos for a fixed number of iterations. As shown in [4], the Lanczos variance estimator $\boldsymbol{z}_{[k]} := \text{diag}^{-1}(\boldsymbol{B}\boldsymbol{Q}\boldsymbol{T}^{-1}\boldsymbol{Q}^T\boldsymbol{B}^T)$ has the property that $z_{[k],j} \leq z_{[k+1],j} \leq \ldots z_j$ for all j and k: it is monotonically increasing. Lanczos approximations underestimate covariance in general.

In order to efficiently implement Gaussian message passing, we will use Lanczos low rank approximations $\tilde{\boldsymbol{A}}_{t\to} = \boldsymbol{Q}_{t\to}\boldsymbol{T}_{t\to}\boldsymbol{Q}_{t\to}^T$, $\boldsymbol{Q}_{t\to} \in \mathbb{R}^{\tilde{n} \times k_m}$ (k_m Lanczos steps). This factorization is aligned with common "square root" Kalman filter implementations [5]. Recalling eq. 3 in [3], simple algebra results in

$$\mathcal{M}_{t\to} = \mathcal{M}(\tilde{\boldsymbol{A}}_{t\to}, \boldsymbol{\Gamma}_{t\to}^{-1}) = \boldsymbol{Q}_{t\to}\left(\boldsymbol{T}_{t\to}^{-1} + \boldsymbol{Q}_{t\to}^T\boldsymbol{\Gamma}_{t\to}\boldsymbol{Q}_{t\to}\right)^{-1}\boldsymbol{Q}_{t\to}^T = \boldsymbol{V}_{t\to}\boldsymbol{V}_{t\to}^T, \quad \boldsymbol{V}_{t\to} \in \mathbb{R}^{\tilde{n} \times k_m},$$

computed from $\boldsymbol{Q}_{t\to}, \boldsymbol{T}_{t\to}$ in $O(n\,k_m^2)$ using the Cholesky decomposition of $\boldsymbol{T}_{t\to}^{-1} + \boldsymbol{Q}_{t\to}^T\boldsymbol{\Gamma}_{t\to}\boldsymbol{Q}_{t\to}$ (based on which $\log|\tilde{\boldsymbol{A}}_{t\to} + \boldsymbol{\Gamma}_{t\to}^{-1}|$ is computed as well, as part of the $\log|\boldsymbol{A}|$ estimate). Kalman

---

[1]In fact, prior and likelihood potentials both have MRF structure, but in different domains (image and Fourier), which are densely and nonlocally coupled.

filtering is approximated by iterating between message matrix computations and Lanczos runs to determine $\boldsymbol{Q}_{(t+1)\rightarrow}$, $\boldsymbol{T}_{(t+1)\rightarrow}$ for the precision matrix $\tilde{\boldsymbol{A}}_{(t+1)\rightarrow} = \boldsymbol{A}_{t+1} + \boldsymbol{V}_{t\rightarrow}\boldsymbol{V}_{t\rightarrow}^T$. A $\tilde{\boldsymbol{A}}_{(t+1)\rightarrow}$-MVM costs one $\boldsymbol{A}_{t+1}$-MVM plus $O(n\,k_m)$. Messages passed from one node to the next are of size $\tilde{n} \cdot k_m$. Once all messages are computed, node variances (or design scores) are approximated by running Lanczos on $\tilde{\boldsymbol{A}}_t = \boldsymbol{A}_t + \boldsymbol{V}_{(t-1)\rightarrow}\boldsymbol{V}_{(t-1)\rightarrow}^T + \boldsymbol{V}_{\leftarrow(t+1)}\boldsymbol{V}_{\leftarrow(t+1)}^T$ for k_c iterations. Pair variances $\mathrm{Var}_Q[\boldsymbol{s}_{t\rightarrow}|\boldsymbol{y}]$ are estimated by running Lanczos on vectors of size $2\tilde{n}$ (say for $k_c/2$ iterations; the precision matrix is given in [3, Sect. 3]. Again, generic single slice code can be used to run all these operations, simply extending $\boldsymbol{A}_t$-MVM code by appending low rank message matrix MVMs.

What about parallelization? Message passing along a chain is a serial operation. In two-filter smoothing, the forward and backward pass can be run in parallel. To avoid communication, each pass is computed on a single processor, sending messages to others dedicated to marginal covariance computation (which can start once left and right messages are available for a node). Global criterion evaluation requires collecting $\log|\boldsymbol{A}|$ parts from messages and node covariances. k_c should be larger than k_m, because node covariance computations can be done independently in parallel, and resulting Lanczos representations do not have to be communicated. Note that the Lanczos algorithm is more difficult to run than linear conjugate gradients. Beside the MVM cost, it runs up $O(n\,k)$ storage and $O(n\,k^2)$ extra computation: the dense $\boldsymbol{Q}$ has to be stored, and each new column has to be orthogonalized against the others (due to intrinsic numerical problems, orthogonality is rapidly lost without re-orthogonalization). Here, and also to speed up MVMs with design matrices $\boldsymbol{X}_t$, parallelization on the fine-grained level of commodity graphics hardware should be used. Deflation of new columns to $\boldsymbol{Q}$ is trivial to parallelize, and FFT implementations for graphics hardware are publicly available.

Details for Experimental Setup

In this section, we provide further details for the experimental setup in [3, Sect. 5.2]. We use sagittal head scan data of resolution 64×64 in-plane, 32 slices, acquired on a Siemens 3T scanner (phase encode direction anterior-posterior). Preprocessing is done as in [4]. All designs below contain the 8 lowest-frequency encodes. A low-frequency phase map is obtained from this data by inverse FFT, then divided out. Phase compensation is commonplace in MRI. In the context of sparse reconstruction, strong phase contributions reduce sparsity, and are best filtered out beforehand. Note that our simple compensation method is purely postprocessing: no additional phase encodes have to be acquired.

The sparsity prior is constructed as in [4, 2], then lifted to complex values as detailed in [3]. $\boldsymbol{B}_{\mathrm{orig}}$ consists of two parts: an orthonormal wavelet transform, and the 2D image gradient transform. Corresponding Laplace parameters are τ_a (wavelet) and τ_r (gradient). The imaginary part penalties of strength τ_i and between-slices penalties of strength τ_c are described in [3].

References

[1] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996.
[2] M. Lustig, D. Donoho, and J. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magn. Reson. Med.*, 85(6):1182–1195, 2007.
[3] M. Seeger. Speeding up magnetic resonance image acquisition by Bayesian multi-slice adaptive compressed sensing. In *Advances in Neural Information Processing Systems 22*, 2010. In print.
[4] M. Seeger, H. Nickisch, R. Pohmann, and B. Schölkopf. Bayesian experimental design of magnetic resonance imaging sequences. In *NIPS 21*, pages 1441–1448, 2009.
[5] D. Treebushny and H. Madsen. On the construction of a reduced rank square-root Kalman filter for efficient uncertainty propagation. *Future Gener. Comput. Syst.*, 21(7):1047–1055, 2005.