# Collect, Commit, Expand: an Efficient Strategy for Column Subset Selection on Extremely Wide Matrices

*Robin Armstrong, Anil Damle*

Abstract

The column subset selection problem (CSSP) appears in a remarkably wide range of applications. For example, point selection problems that arise in model order reduction [5], computational chemistry [7], spectral clustering [8], low-rank approximation [6, 13], and Gaussian process regression [15] can all be treated as instances of CSSP. Given a matrix $A \in \mathbb{R}^{m \times n}$ and a target rank $k \leq \min\{m, n\}$, CSSP seeks to find a set of $k$ columns from $A$ that are "highly linearly independent." A more formal statement, using the framework of rank-revealing QR factorizations [4, 11, 12], is that algorithms for CSSP produce an index set $S \in [n]^k$ satisfying

$$\sigma_{\min}(A(:, S)) \geq \frac{\max_{J \in [n]^k} \sigma_{\min}(A(:, J))}{q(n,\, k)} \tag{1}$$

for some low-degree bivariate polynomial $q$. The Golub-Businger algorithm [3], which uses alternating column pivots and Householder reflections to compute a column-pivoted QR (CPQR) factorization $A\Pi = QR$, is widely used for this problem. After running this algorithm, choosing $A(:, S) = A\Pi(:, 1 : k)$ results in an $S$ which does not provably satisfy (1), but which nearly always brings $\sigma_{\min}(A(:, S))$ reasonably close to its maximum.

We seek to address a computational bottleneck in the Golub-Businger algorithm that results from sequential application of Householder reflections with level-2 BLAS. Most existing solutions to this problem involve reducing the number of rows manipulated with BLAS-2. For example, the CPQR factorization routine in LAPACK reflects only as many rows as are needed to determine a small block of pivot columns, deferring the full Householder reflection to a later application with BLAS-3 [16]. There also exists a large class of randomized algorithms that apply standard CPQR routines to sketched matrices with far fewer rows [6, 10, 14, 17]. We, however, are interested in problems where the difficulty arises not from the number of rows, but from the number of columns. For example, spectral clustering generates instances of CSSP where each row represents a cluster and each column represents a data point [8], meaning $m$ may be several orders of magnitude smaller than $n$. In these applications, reducing the number of rows being manipulated with BLAS-2 does not address the main bottleneck.

We will demonstrate a new CPQR-based column selection algorithm that effectively mitigates the BLAS-2 bottleneck for matrices with far more columns than rows. Our algorithm divides columns into a "tracked" set, where residual column norms are recorded, and an "untracked" set, where only overall norms are recorded. Pivot columns are selected in blocks, and each block is selected using a three-step strategy:

1. A "collect" step assembles a small number of candidate columns from the tracked set, and forms a conventional CPQR factorization of the candidates.
2. A "commit" step uses the CPQR factors to identify a set of provably "safe" pivots from among the candidates, and updates *only* the tracked columns according to the new pivots.
3. An "expand" step moves a small number of columns from "untracked" to "tracked," setting up for a new round of candidates to be chosen in the next block.

We call this algorithm CCEQR ("Collect-Commit-Expand QR").

| $n$ | GEQP3 Runtime (s) | CCEQR Runtime (s) |
|---|---|---|
| $10^2$ | $1.9 \times 10^{-5}$ | $8.1 \times 10^{-5}$ |
| $10^3$ | $2.7 \times 10^{-4}$ | $3.7 \times 10^{-4}$ |
| $10^4$ | $1.8 \times 10^{-3}$ | $7.5 \times 10^{-4}$ |
| $10^5$ | $1.8 \times 10^{-2}$ | $3.7 \times 10^{-3}$ |
| $10^6$ | $4.0 \times 10^{-1}$ | $4.5 \times 10^{-2}$ |

Figure 1: Average runtimes of GEQP3 and CCEQR (over 20 trials) on matrices of size $20 \times n$, for increasing $n$. Test matrices were generated from a spectral clustering problem, and correspond to Laplacian embeddings of $n$ data points drawn from a 20-component Gaussian mixture model.

CCEQR is fully deterministic, and unlike CPQR-based column selection algorithms which distribute the column load across several parallel processors [1, 2, 9], it provably selects the same basis columns as the Golub-Businger algorithm (assuming no ties between residual column norms). Using test problems from domains such as computational chemistry, model order reduction, and spectral clustering, we will demonstrate that CCEQR can run several times faster than the standard LAPACK routine (GEQP3) for matrices with an unbalanced column norm distribution. For example, Figure 1 shows that CCEQR can run as much as 10 times faster than GEQP3 for certain spectral clustering problems. We will also show that CCEQR and GEQP3 have essentially the same runtime for large unstructured problems, such as Gaussian random test matrices.

## References


[1] Christian H. Bischof. A parallel QR factorization algorithm with controlled local pivoting. *SIAM Journal on Scientific and Statistical Computing*, 12(1):36–57, 1991.

[2] Christian H. Bischof and Per Christian Hansen. Structure-preserving and rank-revealing QR-factorizations. *SIAM Journal on Scientific and Statistical Computing*, 12(6):1332–1350, 1991.

[3] Peter Businger and Gene H. Golub. Linear least squares solutions by Householder transformations. *Numerische Mathematik*, 7:269 – 276, 1965.

[4] Shivkumar Chandrasekaran and Ilse C. F. Ipsen. On rank-revealing factorisations. *SIAM Journal on Matrix Analysis and Applications*, 15(2):592–622, 1994.

[5] Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[6] H. Cheng, Z. Gimbutas, P. G. Martinsson, and V. Rokhlin. On the compression of low rank matrices. *SIAM Journal on Scientific Computing*, 26(4):1389–1404, 2005.

[7] Anil Damle, Lin Lin, and Lexing Ying. Compressed representation of Kohn-Sham orbitals via selected columns of the density matrix. *J Chem Theory Comput*, 14:1463–1469, 2015.

[8] Anil Damle, Victor Minden, and Lexing Ying. Simple, direct and efficient multi-way spectral clustering. *Information and Inference: A Journal of the IMA*, 8(1):181–203, 06 2018.

[9] James W. Demmel, Laura Grigori, Ming Gu, and Hua Xiang. Communication avoiding rank revealing QR factorization with column pivoting. *SIAM Journal on Matrix Analysis and Applications*, 36(1):55–89, 2015.

[10] Jed A. Duersch and Ming Gu. Randomized QR with column pivoting. *SIAM Journal on Scientific Computing*, 39(4):C263–C291, January 2017.

[11] Ming Gu and Stanley C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal on Scientific Computing*, 17(4):848–869, 1996.

[12] Y.P. Hong and C.-T. Pan. Rank-revealing QR factorizations and the singular value decomposition. *Mathematics of Computation*, 58(197):213 – 232, 1992.

[13] Michael W. Mahoney and Petros Drineas. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.

[14] Per-Gunnar Martinsson, Gregorio Quintana Ortí, Nathan Heavner, and Robert van de Geijn. Householder QR factorization with randomization for column pivoting (HQRRP). *SIAM Journal on Scientific Computing*, 39(2):C96–C115, 2017.

[15] Victor Minden, Anil Damle, Kenneth L. Ho, and Lexing Ying. Fast spatial Gaussian process maximum likelihood estimation via skeletonization factorizations. *Multiscale Modeling & Simulation*, 15(4):1584–1611, 2017.

[16] Gregorio Quintana-Ortí, Xiaobai Sun, and Christian H. Bischof. A BLAS-3 version of the QR factorization with column pivoting. *SIAM Journal on Scientific Computing*, 19(5):1486–1494, 1998.

[17] Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, 2008.