

Orthogonalization Techniques for a Set of Vectors

HPC for numerical methods and data analysis

Student : Tara Fjellman

Fall 2024

1 Introduction

The goal of this project is to study three different algorithms for the orthogonalization of a set of vectors. Those algorithms are Classical Gram-Schmidt (CGS), Cholesky-QR and TSQR. They are described in the Lecture Notes provided for the course and in the slides from Week 3 and Week 4. Such an algorithm will be needed in the second project where you will need to orthogonalize a set of vectors in parallel. Hence this project should allow you to identify an orthogonalization algorithm that is numerically stable and scales well in parallel. The goal of this report is to explore whether this can be done in a model which includes inertia, and is simple enough to be studied extensively.

Consider a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, where $m \gg n$. The goal is to compute the thin QR factorization of \mathbf{A} , that is $\mathbf{A} = \mathbf{Q}\mathbf{R}$, where $\mathbf{Q} \in \mathbb{R}^{m \times n}$ is orthogonal and $\mathbf{R} \in \mathbb{R}^{n \times n}$ is upper triangular. The matrix \mathbf{A} is available at once. In this project, the three algorithms allow to compute the QR factorization of \mathbf{A} and thus orthogonalize n vectors stored as columns of \mathbf{A} . Typically the number of vectors is in between 50 and a few hundreds, while m can be much larger.

2 Algorithms

2.1 Classical Gram-Schmidt

2.2 Cholesky-QR

2.3 TSQR

3 Parallelisation

3.1 Classical Gram-Schmidt

3.2 Cholesky-QR

3.3 TSQR

4 Experimental procedure

For a theoretical investigation, the bounds on the loss of orthogonality provided in the lectures should be used. For the numerical investigation, you should provide plots measuring the loss of orthogonality as $\|I - Q^T Q\|_2$ and the condition number of the basis Q . When it is possible, these quantities, along with the condition number of the input vectors, should be provided at each iteration of the algorithm when a new vector is orthogonalized. However this will not be possible for Cholesky-QR or TSQR for example. In this case you should present only the results obtained at the end of the algorithm.

This investigation should be done on at least two different matrices. One of these matrices will be generated by uniformly discretizing a parametric function, also used in [1], that we denote $C \in \mathbb{R}^{m \times n}$. For all floating (and thus rational) numbers $0 \leq x, \mu \leq 1$, the function is defined as

$$f(x, \mu) = \frac{\sin(10(\mu + x))}{\cos(100(\mu - x)) + 1.1}$$

and the associated matrix is

$$C \in \mathbb{R}^{m \times n}, \quad C(i, j) = f\left(\frac{i-1}{m-1}, \frac{j-1}{n-1}\right), i \in 1, \dots, m, j \in 1, \dots, n$$

You could use for example $m = 50000$ and $n = 600$. Other matrices could be obtained for example from the SuiteSparse Matrix Collection, <https://sparse.tamu.edu/about>. You can pick any square matrix, and then select only the first columns. The dimensions of the test matrices should be chosen such that you can compute the required quantities.

Your report should state the theoretical bounds for the loss of orthogonality of the three algorithms. You should explain why these measures are important. For each matrix you should plot or put in a table the loss of orthogonality/condition number. You should explain these plots, reference them, and make connection with the theory.

5 Numerical Stability Investigation

6 Runtime Investigation

7 Conclusion

Aknowledgements

A Runtime Estimation