Comments to Reviewers

L. Minah Yang, Alyson Fox, Geoffrey Sanders

May 28, 2020

We would like to thank both reviewers for your time and effort. First, we would like to make a few comments on the major aspects of the revisions:

- **Column partitioned Householder QR factorization algorithm (HQR) has been added to reflect on mixed precision settings of GPU tensor cores units.**
  We added the analysis of the level-3 BLAS variant of HQR since it is the standard HQR implementation in many libraries and can be effortlessly adapted to utilize the block Fused Multiply-Add operations (bFMAs) of NVIDIA TensorCore units. The WY representation of [1] is used instead of the compact, storage-efficient version of [8] since the former is discussed in both [3, 4], which we often refer to in the text.

- **We have added the mixed precision setting of NVIDIA's TensorCore bFMAs.**
  This setting is more relevant and practical since these hardware units are already in use, and result in fewer low precision errors than the inner product mixed precision setting we had introduced. While we do not discuss the speed-up advantages in depth, we do refer to the speed benchmarks for these operations that already exist and may be of interest to the readers.

- **Section 5 (Applications) has been removed from this manuscript.**
  Given the length of the first submission and the additional materials introduced as explained above, we have removed the applications section from this text. However, we plan on presenting our work on using mixed precision arithmetic in graph clustering in a future work, where we will include other graph problems.

Next, we would like to address concerns voiced by both referees:

- **Missing references to relevant and prior works.**
  Referee #1: The paper also suffers a complete lack of acknowledgment of prior work in this area and of the current state-of-the-art.
  - **lines 157-159: citation missing at the end of this sentence.**
    Citations for probabilistic rounding error analyses have been added ([5, 7]) to line A.
  - **lines 163-165: Be more specific about what rounding error analysis framework was established in the textbook [13]. As far as I know, the textbook does not establish any new error analysis framework, nor is it limited to analyses using a single precision.**
    We did not intend to imply that [4] establishes new error analysis framework. In section 3, we have included the standard rounding error analyses for HQR and its level-3 BLAS variant and have appropriately attributed and redirected readers to a main source, [4]. The textbook ([4]) does refer to mixed precision iterative refinement, but the rounding error analysis for HQR(Section 19.3) and aggregated Householder transformations (Section 19.5) both assume a uniform precision setting.

– **Section 2.2: Prior work should be discussed and cited here.**
In section 2.2, we define MP Setting 2.3 (line C), which is the mixed precision inner product that mimics TensorCore bFMAs but in a level-2 BLAS operation. Later in section 4.1 (line D), we discuss the specifics of NVIDIA TensorCore bFMAs, and reference work from [2].

Referee #2: We have added the suggested references for the following topics.

– We have referenced [2] for their work on mixed precision matrix products and LU decomposition in section 4.1 (line D).
– References to ([5, 7]) on probabilistic rounding error analyses have been added to to line A.
– Work on proving that the faithfulness of Algorithm 1 on simulating half precision arithmetic have been referenced ([6]) in line E.

- **Inconsistencies in notation and adhering to standard notation:**
  – **Referee #1:** Hyphens have been removed from "low-", "mixed-", and "high-precision".
  – **Referee #2:** We changed the notation for γ for k accumulated rounding errors in precision type q from $\gamma_q^{(k)}$ to $\gamma_k^{(q)}$ to match the standard notation.
- **P2L44-45: "QR factorization is known to provide a backward stable solution to the linear least squares problem ...'**
We meant to motivate the need for mixed precision QR factorization algorithms since mixed precision is an active area of research. However, we have removed this sentence as it was confusing and did not serve the purpose of the paper.

We now address unique concerns from each referee separately.

Requests from Referee #1

1. **The primary problem that I see is that the authors have not convinced me that there is any novelty in what they call a "new framework" for doing mixed-precision floating point error analysis.**
I need help addressing this.
2. **line 39: What is meant by "exact products"?**
The documentation for TensorCore bFMAs says "full precision products", which in our context is equivalent to a product in exact arithmetic. We have added a full explanation to clarify this in section 2 (line F).
3. **Pages 2-6 contain standard introductory textbook material and can be significantly shortened.**
While this material is standard, intermediate results from these analyses are necessary for the rounding error analyses for the mixed precision variant of the algorithms in section 4. We are aware that this adds significant length to this paper and have tried to shorten it while keeping it accessible for readers who may not be familiar with rounding error analyses. Modifying the existing analyses for various mixed precision settings is precisely our major contribution, which is presented in Section 4.

Requests from Referee #2

1. **Choice of mixed precision assumptions**
   - **Relation with GPU tensor core units**
     We have added a mixed precision setting that addresses this specific hardware. This is first mentioned in line G of section 2.2, and further explored in section 4.2 (line D).
   - **Assumptions on storage precision types**
     In line H of section 4.2, we explain that intermediate matrix products should be stored in the higher precision and the low precision output is used only for the final result of the block matrix product in order to gain the highest accuracy. In the analysis for a mixed precision variant of the level-3 HQR (BQR) in section 4.2, this translates to introducing only $\mathcal{O}(n/r)$ low precision rounding errors for forming the QR factorization for an m-by-n sized matrix whose columns are partitioned in groups of r. Also, we reference the readers to [2] for a full analysis of matrix products.
   - **Distinction between Lemma 2.4 and Corollary 2.5**
     We only define the exact product variant of Corollary 2.5 in MP Setting 2.3. With adding the TensorCore bFMAs, we felt that we didn't need two different types of mixed precision inner products.
2. **Rounding error analysis framework**
3. **Conclusions from the mixed precision HQR analysis (section 3)**
4. **Conclusion from HQR vs TSQR comparison (section 4)**
5. **Mislabel of forward and backward errors**
6. **P1L15: "standard algorithms may no longer be numerically stable when using half precision"**
7. **P2L49: fp16 should be removed, bfloat should be bfloat16.**
8. **P5L125: Rewording is needed to clarify "k represents the number of FLOPs"**
9. **Title suggestion:**
10. **P1L20: what does "weight" refer to in this context?**
11. **P2L57 "can successfully" $\rightarrow$ "can be succcessfully".**
12. **P5L134: $\gamma_p^{(d+2)}$ has not been defined yet.**
13. **P12L329: the middle term should be $(1+\delta_w)(x_1-\sigma-\Delta\sigma)$, rather than $(1+\delta_w)(\sigma+\Delta\sigma)$. Moreover, the last equality is only true because no cancellation can happen, since x_1 and σ have the same sign: this should be commented on.**
14. **Equations (4.6) and (4.7): isn't the $\sqrt{n}$ factor on the wrong equation?**
15. **P18L527: "for the a meaningful".**

16. **P18L531: as mentioned above, the $(L+1)/2^L$ factor is reversed**

17. **P20L599: I find it very strange that the backward error depends on the condition number of the matrix! Is it rather the forward error that is being plotted?**

18. **Section 5: given the relatively theoretical nature of this article, section 5 felt slightly out of place to me. Given that the article is quite long, perhaps the authors could consider including section 4 in another piece of work?**

References

[1] C. Bischof and C. Van Loan, *The WY Representation for Products of Householder Matrices*, SIAM Journal on Scientific and Statistical Computing, 8 (1987), pp. s2–s13, https://doi.org/10.1137/0908009.

[2] P. Blanchard, N. J. Higham, F. Lopez, T. Mary, and S. Pranesh, *Mixed Precision Block Fused Multiply-Add : Error Analysis and Application to GPU Tensor Cores*, (2019).

[3] G. H. Golub and C. F. Van Loan, *Matrix computations*, JHU press, 4 ed., 2013.

[4] N. J. Higham, *Accuracy and Stability of Numerical Methods*, 2002, https://doi.org/10.2307/2669725.

[5] N. J. Higham and T. Mary, *A New Approach to Probabilistic Rounding Error Analysis*, SIAM Journal on Scientific Computing, 41 (2019), pp. A2815–A2835, https://doi.org/10.1137/18M1226312, https://epubs.siam.org/doi/10.1137/18M1226312.

[6] N. J. Higham and S. Pranesh, *Simulating Low Precision Floating-Point Arithmetic*, SIAM Journal on Scientific Computing, 41 (2019), pp. C585–C602, https://doi.org/10.1137/19M1251308, https://epubs.siam.org/doi/10.1137/19M1251308.

[7] I. C. F. Ipsen and H. Zhou, *Probabilistic Error Analysis for Inner Products*, (2019), http://arxiv.org/abs/1906.10465, https://arxiv.org/abs/1906.10465.

[8] R. Schreiber and C. Van Loan, *A Storage-Efficient WY Representation for Products of Householder Transformations*, SIAM Journal on Scientific and Statistical Computing, 10 (1989), pp. 53–57, https://doi.org/10.1137/0910005.