Floating Point Operations in Matrix-Vector Calculus

(Version 1.3)

Raphael Hunger

Technical Report
2007

Technische Universität München
Associate Institute for Signal Processing
Univ.-Prof. Dr.-Ing. Wolfgang Utschick

History

Version 1.00: October 2005

- Initial version

Version 1.01: 2006

- Rewrite of sesquilinear form with a reduced amount of FLOPs
- Several Typos fixed concerning the number of FLOPS required for the Cholesky decomposition

Version 1.2: November 2006

- Conditions for the existence of the standard $\boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ Cholesky decomposition specified (positive definiteness)
- Outer product version of $\boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ Cholesky decomposition removed
- FLOPs required in Gaxpy version of $\boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ Cholesky decomposition updated
- $\boldsymbol{L}_1\boldsymbol{D}\boldsymbol{L}_1^{\mathrm{H}}$ Cholesky decomposition added
- Matrix-matrix product $\boldsymbol{L}\boldsymbol{C}$ added with $\boldsymbol{L}$ triangular
- Matrix-matrix product $\boldsymbol{L}^{-1}\boldsymbol{C}$ added with $\boldsymbol{L}$ triangular and $\boldsymbol{L}^{-1}$ not known a priori
- Inverse $\boldsymbol{L}_1^{-1}$ of a lower triangular matrix with ones on the main diagonal added

Version 1.3: September 2007

- First globally accessible document version

ToDo: (unknown when)

- QR-Decomposition
- LR-Decomposition

Please report any bug and suggestion to *hunger@tum.de*

Contents

1. Introduction

For the design of efficient und low-complexity algorithms in many signal-processing tasks, a detailed analysis of the required number of *floating-point operations* (FLOPs) is often inevitable. Most frequently, matrix operations are involved, such as matrix-matrix products and inverses of matrices. Structures like Hermiteness or triangularity for example can be exploited to reduce the number of needed FLOPs and will be discussed here. In this technical report, we derive expressions for the number of multiplications and summations that a majority of signal processing algorithms in mobile communications bring with them.

Acknowledgments:

The author would like to thank Dipl.-Ing. David A. Schmidt and Dipl.-Ing. Guido Dietl for the fruitful discussions on this topic.

2. Flop Counting

In this chapter, we offer expressions for the number of *complex* multiplications and summations required for several matrix-vector operations. A *floating-point operation* (FLOP) is assumed to be either a complex multiplication *or* a complex summation here, despite the fact that a complex multiplication requires 4 real multiplications and 2 real summations whereas a complex summations constists of only 2 real summations, making a multiplication more expensive than a summation. However, we count each operation as one FLOP.

Throughout this report, we assume $\alpha \in \mathbb{C}$ to be a scalar, the vectors $\boldsymbol{a} \in \mathbb{C}^N$, $\boldsymbol{b} \in \mathbb{C}^N$, and $\boldsymbol{c} \in \mathbb{C}^M$ to have dimension N, N, and M, respectively. The matrices $\boldsymbol{A} \in \mathbb{C}^{M\times N}$, $\boldsymbol{B} \in \mathbb{C}^{N\times N}$, and $\boldsymbol{C} \in \mathbb{C}^{N\times L}$ are assumed to have no special structure, whereas $\boldsymbol{R} = \boldsymbol{R}^{\mathrm{H}} \in \mathbb{C}^{N\times N}$ is Hermitian and $\boldsymbol{D} = \mathbf{diag}\{d_\ell\}_{\ell=1}^{N} \in \mathbb{C}^{N\times N}$ is diagonal. $\boldsymbol{L}$ is a lower triangular $N \times N$ matrix, $\boldsymbol{e}_n$ denotes the unit vector with a 1 in the n-th row and zeros elsewhere. Its dimensionality is chosen such that the respective matrix-vector product exists. Finally, $[\boldsymbol{A}]_{a,b}$ denotes the element in the a-th row and b-th column of $\boldsymbol{A}$, $[\boldsymbol{A}]_{a:b,c:d}$ selects the submatrix of $\boldsymbol{A}$ consisting of rows a to b and columns c to d. $\mathbf{0}_{a\times b}$ is the $a \times b$ zero matrix. Transposition, Hermitian transposition, conjugate, and real-part operator are denoted by $(\cdot)^{\mathrm{T}}$, $(\cdot)^{\mathrm{H}}$, $(\cdot)^*$, and $\Re\{\cdot\}$, respectively, and require no FLOP.

2.1 Matrix Products

Frequently arising matrix products and the amount of FLOPs required for their computation will be discussed in this section.

2.1.1 Scalar-Vector Multiplication $\alpha\boldsymbol{a}$

A simple multiplication $\alpha\boldsymbol{a}$ of a vector $\boldsymbol{a}$ with a scalar α requires N multiplications and no summation.

2.1.2 Scalar-Matrix Multiplication $\alpha\boldsymbol{A}$

Extending the result from Subsection 2.1.1 to a scalar matrix multiplication $\alpha\boldsymbol{A}$ requires NM multiplications and again no summation.

2.1.3 Inner Product $\boldsymbol{a}^{\mathrm{H}}\boldsymbol{b}$ of Two Vectors

An inner product $\boldsymbol{a}^{\mathrm{H}}\boldsymbol{b}$ requires N multiplications and $N-1$ summations, i.e., $2N-1$ FLOPs.

2.1.4 Outer Product $\boldsymbol{a}\boldsymbol{c}^{\mathrm{H}}$ of Two Vectors

An outer product $\boldsymbol{a}\boldsymbol{c}^{\mathrm{H}}$ requires NM multiplications and no summation.

2.1.5 Matrix-Vector Product $\boldsymbol{Ab}$

Computing $\boldsymbol{Ab}$ corresponds to applying the inner product rule $\boldsymbol{a}_i^{\mathrm{H}}\boldsymbol{b}$ from Subsection 2.1.3 M times. Obviously, $1 \leq i \leq M$ and $\boldsymbol{a}_i^{\mathrm{H}}$ represents the i-th row of $\boldsymbol{A}$. Hence, its computation costs MN multiplications and $M(N-1)$ summations, i.e., $2MN - M$ FLOPs.

2.1.6 Matrix-Matrix Product $\boldsymbol{AC}$

Repeated application of the matrix-vector rule $\boldsymbol{Ac}_i$ from Subsection 2.1.5 with $\boldsymbol{c}_i$ being the i-th column of $\boldsymbol{C}$ yields the overall matrix-matrix product $\boldsymbol{AC}$. Since $1 \leq i \leq L$, the matrix-matrix product has the L-fold complexity of the matrix-vector product. Thus, it needs MNL multiplications and $ML(N-1)$ summations, altogether $2MNL - ML$ FLOPs.

2.1.7 Matrix Diagonal Matrix Product $\boldsymbol{AD}$

If the right hand side matrix $\boldsymbol{D}$ of the matrix product $\boldsymbol{AD}$ is diagonal, the computational load reduces to M multiplications for each of the N columns of $\boldsymbol{A}$, since the n-th column of $\boldsymbol{A}$ is scaled by the n-th main diagonal element of $\boldsymbol{D}$. Thus, MN multiplications in total are required for the computation of $\boldsymbol{AD}$, no summations are needed.

2.1.8 Matrix-Matrix Product $\boldsymbol{LD}$

When multiplying a lower triangular matrix $\boldsymbol{L}$ by a diagonal matrix $\boldsymbol{D}$, column n of the matrix product requires $N - n + 1$ multiplications and no summations. With $n = 1, \ldots, N$, we get $\frac{1}{2}N^2 + \frac{1}{2}N$ multiplications.

2.1.9 Matrix-Matrix Product $\boldsymbol{L}_1\boldsymbol{D}$

When multiplying a lower triangular matrix $\boldsymbol{L}_1$ *with ones on the main diagonal* by a diagonal matrix $\boldsymbol{D}$, column n of the matrix product requires $N - n$ multiplications and no summations. With $n = 1, \ldots, N$, we get $\frac{1}{2}N^2 - \frac{1}{2}N$ multiplications.

2.1.10 Matrix-Matrix Product $\boldsymbol{LC}$ with $\boldsymbol{L}$ Lower Triangular

Computing the product of a lower triangular matrix $\boldsymbol{L} \in \mathbb{C}^{N\times N}$ and $\boldsymbol{C} \in \mathbb{C}^{N\times L}$ is done columnwise. The nth element in each column of $\boldsymbol{LC}$ requires n multiplications and $n-1$ summations, so the complete column needs $\sum_{n=1}^{N} n = \frac{N^2}{2} + \frac{N}{2}$ multiplications and $\sum_{n=1}^{N}(n-1) = \frac{N^2}{2} - \frac{N}{2}$ summations. The complete matrix-matrix product is obtained from computing L columns. We have $\frac{N^2L}{2} + \frac{NL}{2}$ multiplications and $\frac{N^2L}{2} - \frac{NL}{2}$ summations, yielding a total amount of N^2L FLOPs.

2.1.11 Gram $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$ of $\boldsymbol{A}$

In contrast to the general matrix product from Subsection 2.1.6, we can make use of the Hermitian structure of the product $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A} \in \mathbb{C}^{N\times N}$. Hence, the strictly lower triangular part of $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$ need not be computed, since it corresponds to the Hermitian of the strictly upper triangular part. For this reason, we have to compute only the N main diagonal entries of $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$ and the $\frac{N^2-N}{2}$ upper off-diagonal elements, so only $\frac{N^2+N}{2}$ different entries have to be evaluated. Each element requires an inner product step from Subsection 2.1.3 costing M multiplications and $M-1$ summations. Therefore, $\frac{1}{2}MN(N+1)$ multiplications and $\frac{1}{2}(M-1)N(N+1)$ summations are needed, making up a total amount of $MN^2 + MN - \frac{N^2}{2} - \frac{N}{2}$ FLOPs.

2.1.12 Squared Frobenius Norm $\|\boldsymbol{A}\|_\mathrm{F}^2 = \mathrm{tr}(\boldsymbol{A}^\mathrm{H}\boldsymbol{A})$

The squared *Hilbert-Schmidt* norm $\|\boldsymbol{A}\|_\mathrm{F}^2$ follows from summing up the MN squared entries from $\boldsymbol{A}$. We therefore have MN multiplications and $MN-1$ summations, yielding a total of $2MN-1$ FLOPs.

2.1.13 Sesquilinear Form $\boldsymbol{c}^\mathrm{H}\boldsymbol{A}\boldsymbol{b}$

The sesquilinear form $\boldsymbol{c}^\mathrm{H}\boldsymbol{A}\boldsymbol{b}$ should be evaluated by computing the matrix-vector product $\boldsymbol{A}\boldsymbol{b}$ in a first step and then multiplying with the row vector $\boldsymbol{c}^\mathrm{H}$ from the left hand side. The matrix vector product requires MN multiplications and $M(N-1)$ summations, whereas the inner product needs M multiplications and $M-1$ summations. Altogether, $M(N+1)$ multiplications and $MN-1$ summations have to be computed for the sesquilinear form $\boldsymbol{c}^\mathrm{H}\boldsymbol{A}\boldsymbol{b}$, yielding a total number of $2MN+M-1$ flops.

2.1.14 Hermitian Form $\boldsymbol{a}^\mathrm{H}\boldsymbol{R}\boldsymbol{a}$

With the Hermitian matrix $\boldsymbol{R}=\boldsymbol{R}^\mathrm{H}$, the product $\boldsymbol{a}^\mathrm{H}\boldsymbol{R}\boldsymbol{a}$ can be expressed as

$$\begin{aligned} \boldsymbol{a}^\mathrm{H}\boldsymbol{R}\boldsymbol{a} &= \sum_{m=1}^{N}\sum_{n=1}^{N} \boldsymbol{a}^\mathrm{H}\boldsymbol{e}_m\boldsymbol{e}_m^\mathrm{T}\boldsymbol{R}\boldsymbol{e}_n\boldsymbol{e}_n^\mathrm{T}\boldsymbol{a} \\ &= \sum_{m=1}^{N}\sum_{n=1}^{N} a_m^* a_n r_{m,n} \\ &= \sum_{m=1}^{N} |a_m|^2 r_{m,m} + 2\sum_{m=1}^{N-1}\sum_{n=m+1}^{N} \Re\{a_m^* a_n r_{m,n}\}, \end{aligned} \tag{2.1}$$

with $a_m = [\boldsymbol{a}]_{m,1}$, and $r_{m,n} = [\boldsymbol{R}]_{m,n}$. The first sum accumulates the weighted main diagonal entries and requires $2N$ multiplications and $N-1$ summations.[1] The second part of (2.1) accumulates all weighted off-diagonal entries from $\boldsymbol{A}$. The last two summations sum up $\frac{N(N-1)}{2}$ terms[2]. Consequently, the second part of (2.1) requires $\frac{N(N-1)}{2}-1$ summations and $N(N-1)$ products[3]. Finally, the two parts have to be added accounting for an additional summation and yielding an overall amount of N^2+N products and $\frac{1}{2}N^2+\frac{1}{2}N-1$ summations, corresponding to $\frac{3}{2}N^2+\frac{3}{2}N-1$ FLOPs[4].

2.1.15 Gram $\boldsymbol{L}^\mathrm{H}\boldsymbol{L}$ of a Lower Triangular Matrix $\boldsymbol{L}$

During the computation of the inverse of a positive definite matrix, the Gram matrix of a lower triangular matrix occurs when Cholesky decomposition is applied. Again, we make use of the Hermitian structure of the Gram $\boldsymbol{L}^\mathrm{H}\boldsymbol{L}$, so only the main diagonal entries and the upper right off-diagonal entries of the product have to be evaluated. The a-th main-diagonal entry can be expressed

[1] We do not exploit the fact that only real-valued summands are accumulated as we only account for complex flops.

[2] $\sum_{m=1}^{N-1}\sum_{n=m+1}^{N} 1 = \sum_{m=1}^{N-1}(N-m) = N(N-1) - \sum_{m=1}^{N-1} m = N(N-1) - \frac{N(N-1)}{2} = \frac{N(N-1)}{2}$. We made use of (A1) in the Appendix for the computation of the last sum accumulating subsequent integers.

[3] The scaling with the factor 2 does not require a FLOP, as it can be implemented by a simple bit shift.

[4] Clearly, if $N=1$, we have to subtract one summation from the calculation since no off-diagonal entries exist.

as

$$[\boldsymbol{L}^{\mathrm{H}}\boldsymbol{L}]_{a,a} = \sum_{n=a}^{N} |\ell_{n,a}|^2, \tag{2.2}$$

with $\ell_{n,a} = [\boldsymbol{L}]_{n,a}$, requiring $N-a+1$ multiplications and $N-a$ summations. Hence, all main diagonal elements need $\sum_{n=1}^{N}(N-n+1) = \frac{1}{2}N^2 + \frac{1}{2}N$ multiplications and $\sum_{n=1}^{N}(N-n) = \frac{1}{2}N^2 - \frac{1}{2}N$ summations.
The upper right off-diagonal entry $[\boldsymbol{L}^{\mathrm{H}}\boldsymbol{L}]_{a,b}$ in row a and column b with $a < b$ reads as

$$[\boldsymbol{L}^{\mathrm{H}}\boldsymbol{L}]_{a,b} = \sum_{n=b}^{N} \ell_{n,a}^{*}\ell_{n,b}, \tag{2.3}$$

again accounting for $N-b+1$ multiplications and $N-b$ summations. These two expressions have to be summed up over all $1 \leq a \leq N-1$ and $a+1 \leq b \leq N$, and for the number of multiplications, we find

$$\begin{aligned}
\sum_{a=1}^{N-1}\sum_{b=a+1}^{N}(N-b+1) &= \sum_{a=1}^{N-1}\left[(N-a)(N+1) - \sum_{b=a+1}^{N} b\right] \\
&= \sum_{a=1}^{N-1}\left[N^2+N-a(N+1) - \frac{N(N+1)-a(a+1)}{2}\right] \\
&= \sum_{a=1}^{N-1}\left[\frac{N^2+N}{2} + \frac{a^2}{2} - a\left(N+\frac{1}{2}\right)\right] \\
&= \frac{(N-1)(N+1)N}{2} + \frac{(N-1)N(2N-1)}{2\cdot 6} - \left(N+\frac{1}{2}\right)\frac{N(N-1)}{2} \\
&= \frac{1}{6}N^3 - \frac{1}{6}N.
\end{aligned} \tag{2.4}$$

Again, we made use of (A1) for the sum of subsequent integers and (A2) for the sum of subsequent squared integers. For the number of summations, we evaluate

$$\sum_{a=1}^{N-1}\sum_{b=a+1}^{N}(N-b) = \frac{1}{6}N^3 - \frac{1}{2}N^2 + \frac{1}{3}N. \tag{2.5}$$

Computing all necessary elements of the Gram $\boldsymbol{L}^{\mathrm{H}}\boldsymbol{L}$ thereby requires $\frac{1}{6}N^3 + \frac{1}{2}N^2 + \frac{1}{3}N$ multiplications and $\frac{1}{6}N^3 - \frac{1}{6}N$ summations. Altogether, $\frac{1}{3}N^3 + \frac{1}{2}N^2 + \frac{1}{6}N$ FLOPs result. The same result of course holds for the Gram of two upper triangular matrices.

2.2 Decompositions

2.2.1 Cholesky Decomposition $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ (Gaxpy Version)

Instead of computing the inverse of a *positive definite* matrix $\boldsymbol{R}$ directly, it is more efficient to start with the Cholesky decomposition $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ and then invert the lower triangular matrix $\boldsymbol{L}$ and compute its Gram. In this section, we count the number of FLOPs necessary for the Cholesky decomposition.

The implementation of the Generalized $\boldsymbol{Ax}$ plus $\boldsymbol{y}$ (Gaxpy) version of the Cholesky decomposition, which overwrites the lower triangular part of the positive definite matrix $\boldsymbol{R}$ is listed in Algorithm 2.1, see [1]. Note that $\boldsymbol{R}$ needs to be positive definite for the $\boldsymbol{LL}^{\mathrm{H}}$ decomposition!

Algorithm 2.1 Algorithm for the Gaxpy version of the Cholesky decomposition.

```
1: $[\boldsymbol{R}]_{1:N,1} = \dfrac{\overbrace{[\boldsymbol{R}]_{1:N,1}}^{\in\mathbb{C}^N}}{\sqrt{[\boldsymbol{R}]_{1,1}}}$
2: for $n = 2$ to $N$ do
3:     $[\boldsymbol{R}]_{n:N,n} = \underbrace{[\boldsymbol{R}]_{n:N,n}}_{\in\mathbb{C}^{N-n+1}} - \underbrace{[\boldsymbol{R}]_{n:N,1:n-1}}_{\in\mathbb{C}^{(N-n+1)\times(n-1)}} \underbrace{[\boldsymbol{R}]^{\mathrm{H}}_{n,1:n-1}}_{\in\mathbb{C}^{(n-1)}}$
4:     $[\boldsymbol{R}]_{n:N,n} = \dfrac{\overbrace{[\boldsymbol{R}]_{n:N,n}}^{\in\mathbb{C}^{N-n+1}}}{\sqrt{[\boldsymbol{R}]_{n,n}}}$
5: end for
6: $\boldsymbol{L} = \mathrm{tril}(\boldsymbol{R})$        {lower triangular part of overwritten $\boldsymbol{R}$}
```

The computation of the first column of $\boldsymbol{L}$ in Line 1 of Algorithm 2.1 requires $N-1$ multiplications[5], a single square-root operation, and no summations. Column $n > 1$ takes a matrix vector product of dimension $(N-n+1)\times(n-1)$ which is subtracted from another $(N-n+1)$-dimensional vector involving $N-n+1$ summations, see Line 3. Finally, $N-n$ multiplications[6] and a single square-root operation are necessary in Line 4. In short, row n with $1 < n \leq N$ needs $-n^2 + n(N+1) - 1$ multiplications, $-n^2 + n(N+2) - N - 1$ summations (see Subsection 2.1.5), and one square root operation, which we classify as an additional FLOP. Summing up the multiplications for rows $2 \leq n \leq N$, we obtain

$$\begin{aligned}\sum_{n=2}^{N}(-n^2 + n(N+1) - 1) &= (N+1)\frac{N(N+1)-2}{2} - \frac{N(N+1)(2N+1)-6}{6} - (N-1)\\ &= \frac{N^3+2N^2-N}{2} - \frac{2N^3+3N^2+N}{6} - (N-1)\\ &= \frac{1}{6}N^3 + \frac{1}{2}N^2 - \frac{5}{3}N + 1.\end{aligned} \tag{2.6}$$

The number of summations for rows $2 \leq n \leq N$ reads as

$$\begin{aligned}\sum_{n=2}^{N}(-n^2 + n(N+2) - N - 1) &= -(N+1)(N-1) + (N+2)\frac{N(N+1)-2}{2}\\ &\quad - \frac{N(N+1)(2N+1)-6}{6}\\ &= -N^2 + 1 + \frac{N^3+3N^2-4}{2} - \frac{2N^3+3N^2+N-6}{6}\\ &= \frac{1}{6}N^3 - \frac{1}{6}N,\end{aligned} \tag{2.7}$$

[5] The first element need not be computed twice, since the result of the division is the square root of the denominator.

[6] Again, the first element need not be computed twice, since the result of the division is the square root of the denominator.

Algorithm 2.2 Algorithm for the Cholesky decomposition $\boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^{\mathrm{H}}$.

```
1: $[\boldsymbol{R}]_{2:N,1} = \frac{\overbrace{[\boldsymbol{R}]_{2:N,1}}^{\in\mathbb{C}^{N-1}}}{[\boldsymbol{R}]_{1,1}}$
2: for $n = 2$ to $N$ do
3:     for $i = 1$ to $n - 1$ do
4:         $[\boldsymbol{v}]_i = \begin{cases} [\boldsymbol{R}]_{1,n} & \text{if } i = 1 \\ [\boldsymbol{R}]_{i,i}[\boldsymbol{R}]^*_{n,i} & \text{if } i \neq 1 \end{cases}$
5:     end for
6:     $[\boldsymbol{v}]_n = [\boldsymbol{R}]_{n,n} - \underbrace{[\boldsymbol{R}]_{n,1:n-1}}_{\in\mathbb{C}^{1\times n-1}}\underbrace{[\boldsymbol{v}]_{1:n-1}}_{\in\mathbb{C}^{n-1}}$
7:     $[\boldsymbol{R}]_{n,n} = [\boldsymbol{v}]_n$
8:     $[\boldsymbol{R}]_{n+1:N,n} = \frac{\overbrace{[\boldsymbol{R}]_{n+1:N,n}}^{\in\mathbb{C}^{N-n}} - \overbrace{[\boldsymbol{R}]_{n+1:N,1:n-1}}^{\in\mathbb{C}^{(N-n)\times(n-1)}}\overbrace{[\boldsymbol{v}]_{1:n-1}}^{\in\mathbb{C}^{n-1}}}{[\boldsymbol{v}]_n}$
9: end for
10: $\boldsymbol{D} = \mathrm{diag}(\mathrm{diag}(\boldsymbol{R}))$ (return diagonal $\boldsymbol{D}$)
11: $\boldsymbol{L}_1 = \mathrm{tril}(\boldsymbol{R})$ with ones on the main diagonal
```

and finally, $N-1$ square-root operations are needed for the $N-1$ rows. Including the $N-1$ multiplications for column $n = 1$ and the additional square root operation, $\frac{1}{6}N^3 + \frac{1}{2}N^2 - \frac{2}{3}N$ multiplications, $\frac{1}{6}N^3 - \frac{1}{6}N$ summations, and N square-root operations occur, $\frac{1}{3}N^3 + \frac{1}{2}N^2 + \frac{1}{6}N$ FLOPs in total.

2.2.2 Cholesky Decomposition $\boldsymbol{R} = \boldsymbol{L}_1\boldsymbol{D}\boldsymbol{L}_1^{\mathrm{H}}$

The main advantage of the $\boldsymbol{L}_1\boldsymbol{D}\boldsymbol{L}_1^{\mathrm{H}}$ decomposition compared to the standard $\boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ decomposition is that no square root operations are needed, which may require more than one FLOP depending on the given hardware platform. Another benefit of the $\boldsymbol{L}_1\boldsymbol{D}\boldsymbol{L}_1^{\mathrm{H}}$ decomposition is that it does not require a positive definite matrix $\boldsymbol{R}$, the only two conditions for the unique existence are that $\boldsymbol{R}$ is Hermitian and all but the last principle minor (*i.e., the determinant*) of $\boldsymbol{R}$ need to be different from zero [2]. Hence, $\boldsymbol{R}$ may also be rank deficient to a certain degree. If $\boldsymbol{R}$ is not positive semidefinite, then $\boldsymbol{D}$ may contain negative main diagonal entries.

The outcome of the decomposition is a lower triangular matrix $\boldsymbol{L}_1$ with ones on the main diagonal and a diagonal matrix $\boldsymbol{D}$.

Algorithm 2.2 overwrites the strictly lower left part of the matrix $\boldsymbol{R}$ with the strictly lower part of $\boldsymbol{L}_1$ (*i.e.,* without the ones on the main diagonal) and overwrites the main diagonal of $\boldsymbol{R}$ with the main diagonal of $\boldsymbol{D}$. It is taken from [1] and slightly modified, such that is also applicable to complex matrices (see the conjugate in Line 4) and no *existing* scalar should be re-computed (see case distinction in Line 4 for $i = 1$).

Line 1 needs $N-1$ multiplications. Lines 3 to 5 require $n-2$ multiplications and are executed for $n = 2, \ldots, N$, yielding $\sum_{n=2}^{N}(n-2) = \frac{N^2-3N+2}{2}$ multiplications. Line 6 takes $n-1$ multiplications and $n-1$ summations, again with $n = 2, \ldots, N$, yielding $\sum_{n=2}^{N}(n-1) = \frac{N^2-N}{2}$ multiplications and the same amount of summations. Line 7 does not require any FLOP. In Line 8, the matrix-vector product needs $(N-n)(n-1)$ multiplications, and additional $N-n$ multiplications arise when the complete numerator is divided by the denominator. Hence, we have $Nn - n^2$ multiplications. For $n = 2, \dots, N$, we get $\sum_{n=2}^{N}(Nn - n^2) = \frac{1}{6}N^3 - \frac{7}{6}N + 1$ multiplications. The number of summations in Line 8 is $(N - n)(n - 2)$ for the matrix vector product and $N - n$ for the subtraction in the numerator. Together, we have $-n^2 + n(N + 1) - N$ summations. With $n = 2, \dots, N$, we get $\sum_{n=2}^{N}[-n^2 + n(N + 1) - N)] = \frac{1}{6}N^3 - \frac{1}{2}N^2 + \frac{1}{3}N$ summations.

Summing up, this algorithm requires $\frac{1}{6}N^3 + N^2 - \frac{13}{6}N + 1$ multiplications, and $\frac{1}{6}N^3 - \frac{1}{6}N$ summations, yielding a total amount of $\frac{1}{3}N^3 + N^2 - \frac{7}{3}N + 1$ FLOPs. (Note that this formula is also valid for $N = 1$.)

2.3 Inverses of Matrices

2.3.1 Inverse $\boldsymbol{L}^{-1}$ of a Lower Triangular Matrix $\boldsymbol{L}$

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_N] = \boldsymbol{L}^{-1}$ denote the inverse of a lower triangular matrix $\boldsymbol{L}$. Then, $\boldsymbol{X}$ is again lower triangular which means that $[\boldsymbol{X}]_{b,n} = 0$ for $b < n$. The following equation holds:

$$\boldsymbol{L}\boldsymbol{x}_n = \boldsymbol{e}_n. \tag{2.8}$$

Via *forward substitution*, above system can easily be solved. Row b ($n \leq b \leq N$) from (2.8) can be expressed as

$$\sum_{a=n}^{b} \ell_{b,a} x_{a,n} = \delta_{b,n}, \tag{2.9}$$

with $\delta_{b,n}$ denoting the *Kronecker* delta which vanishes for $b \neq n$, and $x_{a,n} = [\boldsymbol{X}]_{a,n} = [\boldsymbol{x}_n]_{a,1}$. Starting from $b = 1$, the $x_{b,n}$ are computed successively, and we find

$$x_{b,n} = -\frac{1}{\ell_{b,b}} \left[\sum_{a=n}^{b-1} \ell_{b,a} x_{a,n} - \delta_{b,n} \right], \tag{2.10}$$

with all $x_{a,n}, n \leq a \leq b - 1$ having been computed in previous steps. Hence, if $n = b$, $x_{n,n} = \frac{1}{\ell_{n,n}}$ and a single multiplication[7] is required, no summations are needed. For $b > n$, $b - n + 1$ multiplications and $b - n - 1$ summations are required, as the *Kronecker*-delta vanishes. All main diagonal entries can be computed by means of N multiplications The lower left off-diagonal entries

[7] Actually, it is a division rather than a multiplication.

require

$$\begin{aligned}\sum_{n=1}^{N-1}\sum_{b=n+1}^{N}(b-n+1) &= \sum_{n=1}^{N-1}\left[(1-n)(N-n)+\sum_{b=n+1}^{N} b\right]\\ &= \sum_{n=1}^{N-1}\left[N+n^2-n(N+1)+\frac{N^2+N-n^2-n}{2}\right]\\ &= \sum_{n=1}^{N-1}\left[\frac{N^2}{2}+\frac{3N}{2}+\frac{n^2}{2}-n(N+\frac{3}{2})\right] \qquad (2.11)\\ &= (N-1)\frac{N}{2}(N+3)+\frac{(N-1)N(2N-1)}{2\cdot 6}\\ &\quad -(N+\frac{3}{2})\frac{(N-1)N}{2}\\ &= \frac{1}{6}N^3+\frac{1}{2}N^2-\frac{2}{3}N\end{aligned}$$

multiplications, and

$$\sum_{n=1}^{N-1}\sum_{b=n+1}^{N}(b-n-1) = \frac{1}{6}N^3-\frac{1}{2}N^2+\frac{1}{3}N \qquad (2.12)$$

summations. Including the N multiplications for the main-diagonal entries, $\frac{1}{6}N^3+\frac{1}{2}N^2+\frac{1}{3}N$ multiplications and $\frac{1}{6}N^3-\frac{1}{2}N^2+\frac{1}{3}N$ summations have to be implemented, yielding a total amount of $\frac{1}{3}N^3+\frac{2}{3}N$ FLOPs.

2.3.2 Inverse $\boldsymbol{L}_1^{-1}$ of a Lower Triangular Matrix $\boldsymbol{L}_1$ with Ones on the Main Diagonal

The inverse of a lower triangular matrix $\boldsymbol{L}_1$ turns out to require N^2 FLOPs less than the inverse of $\boldsymbol{L}$ with arbitrary nonzero diagonal elements. Let $\boldsymbol{X}$ denote the inverse of $\boldsymbol{L}_1$. Clearly, $\boldsymbol{X}$ is again a lower triangular matrix with ones on the main diagonal. We can exploit this fact in order to compute only the unknown entries.

The mth row and nth column of the system of equations $\boldsymbol{L}_1\boldsymbol{X}=\mathbf{I}_N$ with $m \geq n+1$ reads as[8]

$$l_{m,n}+\sum_{\substack{i=n+1\\ i\geq m-1}}^{m-1} l_{m,i}x_{i,n}+x_{m,n}=0,$$

or, equivalently,

$$x_{m,n}=-\left[l_{m,n}+\sum_{\substack{i=n+1\\ i\geq m-1}}^{m-1} l_{m,i}x_{i,n}\right].$$

Hence, $\boldsymbol{X}$ is computed via forward substitution. To compute $x_{m,n}$, we need $m-n-1$ multiplications and $m-n-1$ summations. Remember that $m \geq n+1$. The total number of multiplications/summations is obtained from

$$\sum_{n=1}^{N-1}\sum_{m=n+1}^{N}(m-n-1)=\frac{1}{6}N^3-\frac{1}{2}N^2+\frac{1}{3}N. \qquad (2.13)$$

[8] We only have to consider $m \geq n+1$, since the equations resulting from $m < n+1$ are automatically fulfilled due to the structure of $\boldsymbol{L}_1$ and $\boldsymbol{X}$.

Summing up, $\frac{1}{3}N^3 - N^2 + \frac{2}{3}N$ FLOPs are needed.

2.3.3 Inverse $\boldsymbol{R}^{-1}$ of a Positive Definite Matrix $\boldsymbol{R}$

The inverse of a matrix can for example be computed via Gaussian-elimination [1]. However, this approach is computationally expensive and does not exploit the Hermitian structure of $\boldsymbol{R}$. Instead, it is more efficient to start with the Cholesky decomposition of $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ (see Subsection 2.2.1), invert the lower triangular matrix $\boldsymbol{L}$ (see Subsection 2.3.1), and then build the Gram $\boldsymbol{L}^{-\mathrm{H}}\boldsymbol{L}^{-1}$ of $\boldsymbol{L}^{-1}$ (see Subsection 2.1.15). Summing up the respective number of operations, this procedure requires $\frac{1}{2}N^3+\frac{3}{2}N^2$ multiplications, $\frac{1}{2}N^3-\frac{1}{2}N^2$ summations, and N square-root operations, which yields a total amount of $N^3 + N^2 + N$ FLOPs.

2.4 Solving Systems of Equations

2.4.1 Product $\boldsymbol{L}^{-1}\boldsymbol{C}$ with $\boldsymbol{L}^{-1}$ not known *a priori.*

A naive way of computing the solution $\boldsymbol{X} = \boldsymbol{L}^{-1}\boldsymbol{C}$ of the equation $\boldsymbol{L}\boldsymbol{X} = \boldsymbol{C}$ is to find $\boldsymbol{L}^{-1}$ first and afterwards multiply it by $\boldsymbol{C}$. This approach needs $N^2(L + \frac{1}{3}N) + \frac{2}{3}N$ FLOPs as shown in Sections 2.3.1 and 2.1.10. However, doing so is very expensive since we are not interested in the inverse of $\boldsymbol{L}$ in general. Hence, there must be a computationally cheaper variant. Again, *forward substitution* plays a key role.

It is easy to see, that $\boldsymbol{X}$ can be computed column-wise. Let $x_{b,a} = [\boldsymbol{X}]_{b,a}$, $\ell_{b,a} = [\boldsymbol{L}]_{b,a}$, and $c_{b,c} = [\boldsymbol{C}]_{b,a}$. Then, from $\boldsymbol{L}\boldsymbol{X} = \boldsymbol{C}$, we get for the element $x_{b,a}$ in row b and column a of $\boldsymbol{X}$:

$$x_{b,a} = -\frac{1}{\ell_{b,b}}\left[\sum_{i=1}^{b-1}\ell_{b,i}x_{i,a} - c_{b,a}\right]. \tag{2.14}$$

Its computation requires b multiplications and $b - 1$ summations. A complete column of $\boldsymbol{X}$ can therefore the computed with $\sum_{b=1}^{N} b = \frac{N^2}{2}+\frac{N}{2}$ multiplications and $\sum_{b=1}^{N}(b-1) = \frac{N^2}{2}-\frac{N}{2}$ summations. The complete matrix $\boldsymbol{X}$ with L columns thus needs N^2L FLOPs, so the *forward substitution* saves $\frac{1}{3}N^3 + \frac{2}{3}N$ FLOPs compared to the direction inversion of $\boldsymbol{L}$ and a subsequent matrix matrix product. Interestingly, computing $\boldsymbol{L}^{-1}\boldsymbol{C}$ with $\boldsymbol{L}^{-1}$ unknown is as expensive as computing $\boldsymbol{L}\boldsymbol{C}$, see Section 2.1.10.

3. Overview

$\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{B} \in \mathbb{C}^{N \times N}$, and $\boldsymbol{C} \in \mathbb{C}^{N \times L}$ are arbitrary matrices. $\boldsymbol{D} \in \mathbb{C}^{N \times N}$ is a diagonal matrix, $\boldsymbol{L} \in \mathbb{C}^{N \times N}$ is lower triangular, $\boldsymbol{L}_1 \in \mathbb{C}^{N \times N}$ is lower triangular with ones on the main diagonal, $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{C}^N$, $\boldsymbol{c} \in \mathbb{C}^M$, and $\boldsymbol{R} \in \mathbb{C}^{N \times N}$ is positive definite.

Expression	Description	products	summations	FLOPs
$\alpha \boldsymbol{a}$	Vector Scaling	N		N
$\alpha \boldsymbol{A}$	Matrix Scaling	MN		MN
$\boldsymbol{a}^{\mathrm{H}} \boldsymbol{b}$	Inner Product	N	$N-1$	$2N-1$
$\boldsymbol{a} \boldsymbol{c}^{\mathrm{H}}$	Outer Product	MN		MN
$\boldsymbol{A} \boldsymbol{b}$	Matrix Vector Prod.	MN	$M(N-1)$	$2MN-M$
$\boldsymbol{A} \boldsymbol{C}$	Matrix Matrix Prod.	MNL	$ML(N-1)$	$2MNL-ML$
$\boldsymbol{A} \boldsymbol{D}$	Diagonal Matrix Prod.	MN		MN
$\boldsymbol{L} \boldsymbol{D}$	Matrix-Matrix Prod.	$\frac{1}{2}N^2 + \frac{1}{2}N$	0	$\frac{1}{2}N^2 + \frac{1}{2}N$
$\boldsymbol{L}_1 \boldsymbol{D}$	Matrix-Matrix Prod.	$\frac{1}{2}N^2 - \frac{1}{2}N$	0	$\frac{1}{2}N^2 - \frac{1}{2}N$
$\boldsymbol{L} \boldsymbol{C}$	Matrix Product	$\frac{N^2 L}{2} + \frac{NL}{2}$	$\frac{N^2 L}{2} - \frac{NL}{2}$	$N^2 L$
$\boldsymbol{A}^{\mathrm{H}} \boldsymbol{A}$	Gram	$\frac{MN(N+1)}{2}$	$\frac{(M-1)N(N+1)}{2}$	$MN^2+N(M-\frac{N}{2})-\frac{N}{2}$
$\lVert \boldsymbol{A} \rVert_{\mathrm{F}}^2$	Frobenius Norm	MN	$MN-1$	$2MN-1$
$\boldsymbol{c}^{\mathrm{H}} \boldsymbol{A} \boldsymbol{b}$	Sesquilinear Form	$M(N+1)$	$MN-1$	$2MN+M-1$
$\boldsymbol{a}^{\mathrm{H}} \boldsymbol{R} \boldsymbol{a}$	Hermitian Form	N^2+N	$\frac{N^2}{2}+\frac{N}{2}-1$	$\frac{3}{2}N^2+\frac{3}{2}N-1$
$\boldsymbol{L}^{\mathrm{H}} \boldsymbol{L}$	Gram of Triangular	$\frac{N^3}{6}+\frac{N^2}{2}+\frac{N}{3}$	$\frac{N^3}{6}-\frac{N}{6}$	$\frac{1}{3}N^3+\frac{1}{2}N^2+\frac{1}{6}N$
$\boldsymbol{L}$	Cholesky $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{H}}$ (Gaxpy version)	$\frac{N^3}{6}+\frac{N^2}{2}-\frac{2}{3}N$	$\frac{N^3}{6}-\frac{N}{6}$	$\frac{1}{3}N^3+\frac{1}{2}N^2+\frac{1}{6}N$ (N roots included)
$\boldsymbol{L}, \boldsymbol{D}$	Cholesky $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^{\mathrm{H}}$	$\frac{N^3}{6}+N^2-\frac{13N}{6}+1$	$\frac{N^3}{6}-\frac{N}{6}$	$\frac{1}{3}N^3+N^2-\frac{7}{3}N+1$
$\boldsymbol{L}^{-1}$	Inverse of Triangular	$\frac{N^3}{6}+\frac{N^2}{2}+\frac{N}{3}$	$\frac{N^3}{6}-\frac{N^2}{2}+\frac{N}{3}$	$\frac{1}{3}N^3+\frac{2}{3}N$
$\boldsymbol{L}_1^{-1}$	Inverse of Triangular with ones on main diag.	$\frac{N^3}{6}-\frac{N^2}{2}+\frac{N}{3}$	$\frac{N^3}{6}-\frac{N^2}{2}+\frac{N}{3}$	$\frac{1}{3}N^3-N^2+\frac{2}{3}N$
$\boldsymbol{R}^{-1}$	Inverse of Pos. Definite	$\frac{N^3}{2}+\frac{3N^2}{2}$	$\frac{N^3}{2}-\frac{N^2}{2}$	N^3+N^2+N (N roots included)
$\boldsymbol{L}^{-1} \boldsymbol{C}$	$\boldsymbol{L}^{-1}$ *unknown*	$\frac{N^2 L}{2} + \frac{NL}{2}$	$\frac{N^2 L}{2} - \frac{NL}{2}$	$N^2 L$

Appendix

A frequently occurring summation in FLOP counting is the sum of subsequent integers. By complete induction, we find

$$\sum_{n=1}^{N} n = \frac{N(N+1)}{2}. \tag{A1}$$

Above result can easily be verified by recognizing that the sum of the n-th and the $(N-n)$-th summand is equal to $N+1$, and we have $\frac{N}{2}$ such pairs.

Another sum of relevance is the sum of subsequent squared integers. Again, via complete induction, we find

$$\sum_{n=1}^{N} n^2 = \frac{N(N+1)(2N+1)}{6}. \tag{A2}$$

Bibliography

[1] G. H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1991.

[2] Kh.D. Ikramov and N.V. Savel'eva, "Conditionally Definite Matrices," *Journal of Mathematical Sciences*, vol. 98, no. 1, pp. 1–50, 2000.