Provable Lattice Reduction in Nearly the Time of QR Factorization

Yang Zhou[1,2,3] Zhedong Wang[4] Mingsheng Wang[2]

[1]State Key Laboratory of Information Security
[2]Institute of Information Engineering, Chinese Academy of Sciences
[3]School of Cyber Security, University of Chinese Academy of Sciences
[4]Shanghai Jiao Tong University

zhouyang2000@iie.ac.cn
wzdstill@sjtu.edu.cn
wangmingsheng@iie.ac.cn

Abstract. We propose a deterministic lattice-reduction algorithm with bit complexity matching QR factorization up to logarithmic factors, and an implementation that uses fewer instructions than `flatter` (CRYPTO 2023) in our benchmarks. The key technical innovations are iterative precision management, a finer-grained potential analysis, and a new representation for bases and transformations, all integrated into a recursive framework. We analyze the asymptotic behavior of our algorithm, and show that, for fixed reduction and output accuracy, the running time is $\widetilde{O}\big(n^{\omega}\big(1+\log \kappa_{\mathrm{col}}(\mathbf{B})\big)\big)$, where $\mathbf{B} \in \mathbb{Z}^{n\times n}$ is a nonsingular lattice basis and κ_{col} is the optimally column-scaled spectral condition number. This gives a finer complexity than prior art, by allowing a tighter condition number κ_{col}. Besides, through a new observation on the recursive structure, we demonstrate that the algorithm is free from heuristic assumptions. We compare our implementation with `flatter` on q-ary, NTRU-like, scaled q-ary, and Coppersmith inputs. Our implementation uses fewer instructions in all 153 executions. The median ratios of `flatter`'s instruction count to ours are 1.55 for ordinary q-ary inputs and 1.86 for NTRU-like inputs.

1 Introduction

A lattice generated by linearly independent vectors is the set of their integer linear combinations. The same lattice has many bases. Lattice reduction changes the basis while preserving the lattice and seeks short, nearly orthogonal basis vectors. Lattice reduction is a fundamental tool in computational number theory and cryptanalysis: many cryptographic and number-theoretic problems are solved by constructing an appropriate lattice and retrieving one of its short vectors. Examples include knapsack problems [14], breaking linear congruential generators [7], Coppersmith's attack against RSA moduli [3], factoring polynomials over $\mathbb{Z}[X]$ [18], finding integer relations [9], and solving simultaneous Diophantine approximation problems [17].

The celebrated LLL algorithm [18] achieves a $2^{O(n)}$-approximation factor for lattice reduction in time $\widetilde{O}(n^5\beta^2)$[1] for a lattice with input basis $\mathbf{B} \in \mathbb{Z}^{n\times n}$ and $\beta = \max\{1, \lceil \log_2 \max_i \|\mathbf{B}_i\|_2 \rceil\}$, where $\mathbf{B}_i$ is column i of the input. Even though this algorithm is polynomial time, the exponents are so large that it quickly becomes infeasible in practice for even moderately sized lattices.

To improve the practical efficiency of the original LLL algorithm, Nguyen and Stehlé proposed the L^2 algorithm [20], which improves the dependence on β by carefully managing precision for a runtime of $\widetilde{O}(n^4\beta(\beta+n))$ and was subsequently implemented in `fplll` [25]. Although this algorithm remains relatively fast for dimensions up to a few hundred, the running time once again becomes a bottleneck once this range is exceeded. Subsequently, the works of [16, 19, 21] proposed algorithms with a recursive structure to improve the dependence of running time on n and β. However, these algorithms have practical drawbacks, such as a shortness guarantee limited to the first vector of the returned basis [19] and a lack of mature implementations.

Kirchner, Espitau, and Fouque [15] gave an algorithm with a recursive structure that decreases the working precision as the lattice basis is reduced. The algorithm achieves a running time of $\widetilde{O}(n^\omega C)$, where $C > \log(\|\mathbf{B}\|_2\|\mathbf{B}^{-1}\|_2)$, and shows superior performance in tests conducted on lattice bases with high dimensions and high precision. However, their results are based on a very strong heuristic assumption about linear regressions of logarithmic Gram–Schmidt norms related to the Geometric Series Assumption (GSA). Regrettably, this heuristic does not hold for a broad family of lattices [22, Section 2.4] that are prevalent in cryptanalysis, including the NTRU cryptosystem [4], Coppersmith lattices for solving low-degree polynomials modulo integers [3, 12], and partial-information integer factorization [13]. Consequently, the algorithm faces considerable obstacles in practical deployment.

To deal with these classes of cryptanalytically relevant lattices, Ryan and Heninger [22] proposed a new recursive variant of the LLL algorithm called `flatter`, which relies on weaker heuristic assumptions, thereby circumventing the application barriers caused by the strong heuristic assumptions in the algorithm of [15]. Meanwhile, the algorithm also outperforms `fplll` and the algorithm of [15] in practical running time across a broad range of lattice families.

Even though `flatter` has made important progress in terms of applicability and practical efficiency, it still relies on heuristic iteration bounds for its fast running time and for controlling accumulated errors during iterated compression [22, Section 4.3]. From a macro perspective, the fact that lattice basis reduction algorithms rely on heuristic assumptions does not mean that their algorithmic analysis lacks mathematical foundation. However, the practical efficiency and applicability of these algorithms often depend on the strength of the heuristic assumptions themselves. More importantly, heuristic analysis cannot answer the key question of "what algorithmic performance can be guaranteed at least in the worst case." Therefore, *how to achieve high practical efficiency while*

[1] Throughout, $\widetilde{O}$ suppresses subpolynomial factors in the dimension and precision parameters.

maintaining provable guarantees (i.e., without relying on heuristic assumptions) is an important scientific problem in lattice basis reduction.

On the other hand, turning back to LLL-type algorithms, they use Gram–Schmidt orthogonalization to divide the basis-reduction problem into projected subproblems. In the corresponding QR factorization $\mathbf{B} = \mathbf{QR}$, each contiguous diagonal block of $\mathbf{R}$ represents one such subproblem. Classical algorithms process these subproblems through size reductions and neighboring swaps. Recursive algorithms reduce whole projected blocks and batch integer basis changes into matrix operations. QR factorization is a standard matrix method for computing this orthogonalization. Its cost is therefore a natural complexity baseline for lattice reduction: namely, the complexity of LLL-type lattice reduction algorithms consistently exceeds that of QR factorization.

From the perspective of asymptotic algorithmic complexity, the closer the complexity of an LLL-type algorithm approaches that of QR factorization, the superior its asymptotic performance. Among existing algorithms, the two recursive algorithms [15, 22] mentioned above achieve asymptotic complexity closest to that of QR factorization. Their bounds are $\widetilde{O}(n^\omega C)$ and $\widetilde{O}(n^\omega (C+n))$, respectively, where $C > \log(\|\mathbf{B}\|_2 \|\mathbf{B}^{-1}\|_2)$. It is worth noting that the matrix multiplication and QR routines used here have an $O(n^\omega)$ arithmetic cost in the basis dimension n. The factor C is a precision-related quantity that captures the logarithmic dependence on the condition number of the input basis.

However, the condition numbers adopted by the two aforementioned algorithms are suboptimal, resulting in a corresponding factor C that is not as tight as it could be. For the comparison below, define the spectral condition number $\kappa_2(\mathbf{B}) = \|\mathbf{B}\|_2 \|\mathbf{B}^{-1}\|_2$, which is sensitive to column scaling. Consequently, this measure can be large when the input basis is unbalanced (the column norms are highly unbalanced). For instance, letting

$$\mathbf{B} = \begin{pmatrix} 1 & 2^{16} \\ 0 & 2^{16} \end{pmatrix}$$

we have $\log_2 \kappa_2(\mathbf{B}) \geq 16$. In fact, rigorous perturbation analyses of QR factorization [2, 23] suggest that the true measure governing precision should be $\kappa_{\mathrm{col}}(\mathbf{B})$ (the definition is given below). The reason is that (i) κ_{col} is invariant under column scaling, such that it remains small even for unbalanced input bases, e.g., $\kappa_{\mathrm{col}}(\mathbf{B}) \leq 1 + \sqrt{2}$ in this example, and (ii) $\kappa_{\mathrm{col}}(\mathbf{B}) \leq \kappa_2(\mathbf{B})$. Especially for highly unbalanced bases, the gap between the two is particularly pronounced. Although the perturbation analyses of [2, 23] relate the required precision to column-scaling-invariant condition measures $\kappa_{\mathrm{col}}(\mathbf{B})$, they do not establish a lattice-reduction running-time bound matching QR factorization up to logarithmic factors. This leads to the following natural question: *is it possible to construct a lattice reduction algorithm whose asymptotic complexity matches* $\widetilde{O}(n^\omega(1 + \log \kappa_{\mathrm{col}}(\mathbf{B})))$?

1.1 Our results

In this work, we give a new provable lattice-reduction algorithm with asymptotic complexity of $\widetilde{O}\left(n^{\omega}(1+\log \kappa_{\mathrm{col}}(\mathbf{B}))\right)$, where $\mathbf{B}$ is a nonsingular lattice basis, $\kappa_{\mathrm{col}}(\mathbf{B})$ is the optimally scaled condition number defined as

$$\kappa_{\mathrm{col}}(\mathbf{B}) := \inf_{\substack{\mathbf{D} \text{ positive} \\ \text{diagonal}}} \kappa_2(\mathbf{BD}).$$

It can be summarized by the following simplified theorem:

Theorem 1 (Main theorem, simplified). *Let $n \geq 1$ and let $\mathbf{B} \in \mathbb{Z}^{n \times n}$ be a nonsingular input basis under the input conventions of Section 2.2. Let $\omega \in (2,3]$ be a fixed admissible exponent of matrix multiplication. Then there exists a provable lattice reduction algorithm that outputs a basis $\widetilde{\mathbf{C}}$ approximating $\mathbf{C} = \mathbf{BU}$ (i.e., a unimodular transform of the basis $\mathbf{B}$, $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$). The running time of the algorithm is*

$$\widetilde{O}\left(n^{\omega}(1+\log \kappa_{\mathrm{col}}(\mathbf{B}))\right).$$

It should be noted that, similar to the finite-precision approach of [15, 22], our algorithm returns a numerical approximation to the reduced basis. Consistent with [22], the degree of approximation here can be measured by using a parameter $\varepsilon_{\mathrm{out}}$, which ensures that $\|\widetilde{\mathbf{C}}\mathbf{x} - \mathbf{C}\mathbf{x}\|_2 \leq \varepsilon_{\mathrm{out}}\|\mathbf{C}\mathbf{x}\|_2$ for every $\mathbf{x} \neq 0$ (refer to definition 1 for details).

As discussed above, for unbalanced input basis $\mathbf{B}$, we achieve a finer complexity than that of [15, 22]. For upper-triangular, size-reduced input bases $\mathbf{B}$, as is common in cryptanalytic attacks, the asymptotic complexity of our algorithm is $\widetilde{O}\left(n^{\omega}(\log \|\mathbf{B}\|_{\max} + n)\right)$, with quasi-linear dependence on the input precision.

Beyond the running time, the quality of the approximate lattice basis produced by the algorithm is equally important. The following theorem states the properties satisfied by the basis output by our algorithm.

Theorem 2 (Output quality). *Let $0 < \delta \leq \frac{1}{4}$ be a constant, $a_{\mathrm{LLL}} = \frac{1}{2}\log_2(4/3) + \delta$. Let $\mathbf{C} = \mathbf{BU} = [\mathbf{c}_1, \ldots, \mathbf{c}_n]$ be the exact basis in Theorem 1, $\mathbf{C} = \mathbf{QR}$ be the QR factorization with positive diagonal of $\mathbf{C}$, and $\mathbf{c}_i^*$ be the Gram–Schmidt vectors corresponding to $\mathbf{c}_i$. Let $\lambda_i(\boldsymbol{\Lambda})$ be the ith successive minimum of $\boldsymbol{\Lambda} = \mathbf{B}\mathbb{Z}^n$. Then, we have*

$$\begin{aligned} \max_{1 \leq i \leq j \leq n} \frac{\|\mathbf{c}_i^*\|_2}{\|\mathbf{c}_j^*\|_2} &\leq 2^{(a_{\mathrm{LLL}}+\delta)(n-1)}, \\ \|\mathbf{c}_i^*\|_2 &\leq 2^{(a_{\mathrm{LLL}}+\delta)(n-1)}\lambda_i(\boldsymbol{\Lambda}) \qquad (1 \leq i \leq n), \\ \kappa_{\mathrm{col}}(\mathbf{R}) &\leq 2^{O(n \log n)}. \end{aligned}$$

Our algorithm outputs an approximate basis, and as mentioned above, the degree of approximation between the approximate basis and the exact basis can be determined by the constant $\varepsilon_{\mathrm{out}}$. Therefore, when discussing the output

quality, we only need to focus on the vector lengths of the exact basis, because the vector lengths corresponding to the approximate basis differ from the former only by a multiplicative factor of $1 \pm \varepsilon_{\text{out}}$. In addition, the bound on $\kappa_{\text{col}}(\mathbf{R})$ controls the numerical stability of computations with the column-scaled output basis. For further details, please refer to the discussion in Section 2.3 on the precision of Babai's nearest plane algorithm.

The following table 1 provides an intuitive comparison with existing lattice reduction algorithms, focusing on asymptotic complexity, output basis quality and provability.

Technically, we develop the following new tools and techniques, which we believe are of independent interest:

1. Our algorithm uses a new criterion for lattice reduction distinct from prior work, which we call the `forward spread`. Our new definition is used analogously to the Lovász condition of traditional LLL reduction and the `drop` criterion of `flatter`. This criterion is more natural within our algorithmic framework, particularly due to its compatibility with the condition number κ_{col}.
2. We introduce finer-grained potential analyses. By tracking progress through the recursive middle blocks, we prove a minimum potential decrease for every failed round, giving an iteration bound without heuristic assumptions.
3. In order to use a tighter condition number κ_{col}, we propose a new representation for bases and transformations. Our representation separates column scales from numerical operands and stores transformations in the scaled representation, so the working precision depends on `forward spread`, whose initial bound is controlled by the column-scaling-invariant $\kappa_{\text{col}}(\mathbf{B})$.

Table 1. Lattice-reduction bounds for full-rank square integer bases. The quality column gives a bound on the first vector of the output basis. "Provable" indicates whether the stated complexity bound holds without heuristic assumptions.

Reference	Bit complexity	First-vector bound	Provable
[18]	$\widetilde{O}(n^5\beta^2)$	$2^{(n-1)/2}\lambda_1(\boldsymbol{\Lambda})$	Yes
[20]	$\widetilde{O}(n^4\beta(\beta+n))$	$2^{O(n)}\lambda_1(\boldsymbol{\Lambda})$	Yes
[16]	$\widetilde{O}(n^3\max\{n,\beta\}^2)$	$2^{O(n\log n)}\det(\boldsymbol{\Lambda})^{1/n}$	Yes
[21]	$\widetilde{O}(n^5\beta)$	$2^{O(n)}\lambda_1(\boldsymbol{\Lambda})$	Yes
[19]	$\widetilde{O}(n^4\beta)$	$(4/3)^{n/4+O(1)}\det(\boldsymbol{\Lambda})^{1/n}$	Yes
[15]	$\widetilde{O}(n^\omega C)$	$2^{O(n)}\lambda_1(\boldsymbol{\Lambda})$	No
[22]	$\widetilde{O}(n^\omega(C+n))$	$2^{O(n)}\lambda_1(\boldsymbol{\Lambda})$	No
Ours	$\widetilde{O}(n^\omega(1+\log\kappa_{\text{col}}(\mathbf{B})))$	$2^{O(n)}\lambda_1(\boldsymbol{\Lambda})$	Yes

Here $\beta = \max\{1, \lceil\log_2 \max_i \|\mathbf{B}_i\|_2\rceil\}$, where $\mathbf{B}_i$ is column i of the input. This comparison uses quasi-linear integer multiplication and in-memory inputs, with arithmetic and input-access conventions given in Section 2.2.

Algorithm 1 `flatter`'s REDUCE sketch, adapted from [22, Algorithm 1]

Input: Basis $\mathbf{B}$ of rank n and reduction parameter α

while $\mathrm{drop}(\mathbf{B}) > \alpha n$ **do**
  for J = middle half, left half, right half **do**
    $\mathbf{U}_J \leftarrow \textsc{Reduce}(\mathbf{B}_J, \alpha')$ with $\alpha' < \alpha$
    $\mathbf{B} \leftarrow \textsc{Compress}(\mathbf{B}\,\mathrm{emb}_J(\mathbf{U}_J))$
return accumulated unimodular transformation

We also implement our algorithm and benchmark it on q-ary and NTRU-like lattices, two common families in cryptanalysis. Our implementation uses fewer instructions than `flatter` across the tested parameter sets, with median instruction-count ratios of 1.55 and 1.86, respectively. We further construct a scaled q-ary family with fixed κ_{col} across the tested exponent spans. On tested instances from this family, the instruction-count reduction exceeds a factor of 100, illustrating the practical benefit of separating column scales from numerical precision. Our implementation is available in the repository. Appendix E describes the test instances, measurement methodology, and results.

1.2 Technical overview

The `flatter` framework. Algorithm 1 sketches the main algorithm in [22]. Here $\mathbf{B}_J$ is the projected basis on J, and emb_J extends its update by the identity outside J.

Compression and drop. Algorithm 1 controls precision through compression and the drop criterion. Compression uses Reduce-by-Rounding's column scaling [23] to close upward gaps between independently reducible profile blocks. For the QR profile entries $\ell_i = \log_2 R_{ii}$, drop is the total length of the union of downward intervals $[\ell_{i+1}, \ell_i]$ with $\ell_{i+1} < \ell_i$. Closing the gaps leaves a profile range of $\mathrm{drop}(\mathbf{B}) + O(n)$, giving a compressed integer matrix $\widehat{\mathbf{B}}$ with

$$\log \kappa_2(\widehat{\mathbf{B}}) = \mathrm{drop}(\mathbf{B}) + O(n),$$
$$\log_2 \|\widehat{\mathbf{B}}\|_{\max} = O(\mathrm{drop}(\mathbf{B}) + n + \log(1/\eta))$$

at similarity tolerance $0 < \eta < 1$. Thus drop controls the compression output size and subsequent arithmetic cost. For example, take $\mathbf{B} = \mathrm{diag}(2^t, 1)$ with $t \gg 1$. Then $\mathrm{drop}(\mathbf{B}) = t$, yet $\kappa_{\mathrm{col}}(\mathbf{B}) = 1$. The compression size bound therefore does not directly yield a bound in terms of κ_{col}.

The two heuristic assumptions. Heuristic 1 of [22, Section 4.3] assumes that each round removes a constant fraction of the remaining potential decrease. Heuristic 2 assumes a potential decrease of $\Omega(n^3(\alpha(n) - \alpha(n/2)))$ in every round except the last two, where $\alpha(n)$ is the order-n reduction parameter. This lower bound lets the amortized analysis charge the cost of repeated rounds to total potential decrease, even near the stopping threshold [22, Lemmas 17–18]. Together with

Heuristic 1, it gives a logarithmic round bound for controlling the accumulated error.

Our approach. Our approach combines a new **scaled representation** for bases and transformations with a provable lower bound on potential decrease. The representation separates column scales from operands used in updates and return products. This lets us express the complexity in terms of the column-scaling-invariant condition number $\kappa_{\mathrm{col}}(\mathbf{B})$. Choosing a block crossing the boundary between the two halves at the right recursion depth forces potential decrease after every failed stopping test, removing the heuristic assumptions from the analysis.

Initialization. We use a new criterion called `forward spread`: $\mathrm{fspread}(\mathbf{R}) = \max_{i \le j} R_{ii}/R_{jj}$, and denote $\sigma = \mathrm{lfspread}(\mathbf{R}) = \log_2 \mathrm{fspread}(\mathbf{R})$. Unlike drop, σ records the largest forward decrease rather than the total length of the downward ranges. For example, the profile $(t, 0, 3t, 2t)$ has drop $2t$, yet $\sigma = t$.

Column order still matters: reversing the two columns in the diagonal example makes the `forward spread` equal to one. We sort columns by estimates of their Euclidean norms, obtaining $\mathbf{B}^{\mathrm{sort}} = \mathbf{BP}$ for a permutation $\mathbf{P}$. Based on this, we bound the lfspread of the exact QR factor of $\mathbf{B}^{\mathrm{sort}}$ by $O(1+\log \kappa_{\mathrm{col}}(\mathbf{B}))$. To compute such a factor, we double the working precision until QR tests verify the condition and error bounds, without prior knowledge of the condition number. We further prove that the computed $\mathbf{R}^{\mathrm{init}}$ is the R-factor of a basis similar to $\mathbf{B}^{\mathrm{sort}}$ at the prescribed tolerance and satisfies

$$\mathrm{lfspread}(\mathbf{R}^{\mathrm{init}}) = O(1 + \log \kappa_{\mathrm{col}}(\mathbf{B})).$$

Scaled representation. To turn the initial lfspread bound into a precision bound, we keep column scales separate throughout the reduction. We store $\mathbf{R}_{\mathrm{sc}} = \mathbf{R}\mathbf{E}^{-1}$, where $\mathbf{E}$ is diagonal with power-of-two entries approximating the QR diagonal. An integer update $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$ returning scale $\mathbf{E}'$ is represented by the return matrix

$$\mathbf{M} = \mathbf{E}\mathbf{U}(\mathbf{E}')^{-1}, \qquad \mathbf{R}\mathbf{U}(\mathbf{E}')^{-1} = \mathbf{R}_{\mathrm{sc}}\mathbf{M}.$$

Return matrices compose because intermediate scales cancel. Their entries lie on a grid with (i,j)th spacing E_{ii}/E'_{jj}.

Rounding a return product to its grid gives a matrix representing an integer update. We show that the correction between the exact and decoded updates is integral and differs from the identity by less than one on and below the diagonal. The correction is therefore upper unitriangular, preserving the update's unimodularity and the updated basis's exact QR diagonal.

Working precision and accumulated error. To bound the precision needed for these scaled operations, we introduce GridSeysen, which adapts Seysen reduction [24] to the scaled representation and gives the norm bound

$$\log_2 \max\{\|\mathbf{R}_{\mathrm{sc}}\|_1, \|\mathbf{R}_{\mathrm{sc}}^{-1}\|_1\} = O((1+\sigma)\log(2n) + \log^2(2n)).$$

The input and output factors also bound each return matrix and its inverse. Errors of order 2^{-p} are amplified by products of these norms, so $p = \widetilde{O}(1+\sigma)$ bits suffice, apart from accuracy terms. Return products are evaluated from right to left, bounding each accumulated product through its input and output factors.

The potential-decrease bound (1.1), together with potential monotonicity under sufficiently small numerical perturbations, bounds the number of rounds. Summable local error budgets incur only logarithmic precision overhead in this bound and preserve the stopping margins. Basis similarity [22] then transfers the output-quality guarantees to **BU**. A final scaled multiplication returns its numerical approximation.

Provable potential decrease. We use the balanced merge schedule of [22, Algorithm 4], splitting blocks into halves differing in size by at most one dimension. The left and right sides are reduced before a merge begins. We measure progress using the potential of the uncompressed QR profile,

$$\mathrm{Pot}(\mathbf{R}) := \sum_{j=1}^{n-1} \log_2 \prod_{i \le j} R_{ii}.$$

For root order $n_{\mathrm{root}} \ge 2$, put $\gamma = \delta/(2\lceil \log_2 n_{\mathrm{root}} \rceil)$ and $a_n = a_{\mathrm{LLL}} + \gamma \lceil \log_2 n \rceil$. Here n is the current block order, so children have stronger targets.

The middle-block recursion contains smaller blocks crossing the original boundary. Following its first recursive MERGE child lets us choose the size of a crossing block. Write b for the number of columns on its smaller side. Let $\Delta\,\mathrm{Pot}$ be the potential decrease over the round, σ_{out} its output lfspread, and $\sigma_{\mathrm{left}}, \sigma_{\mathrm{mid}}, \sigma_{\mathrm{right}}$ the local bounds for the left side, selected crossing block, and right side, respectively. We show the centered-block inequality

$$\sigma_{\mathrm{out}} \le \sigma_{\mathrm{left}} + \sigma_{\mathrm{mid}} + \sigma_{\mathrm{right}} + \frac{2\Delta\,\mathrm{Pot}}{b^2}.$$

For a fixed excess over the local bounds, the forced decrease grows as b^2. A larger crossing block also has a larger lfspread bound. We choose $b = \Theta(\max\{1, \gamma n\})$, with a sufficiently small constant when γn is large. This leaves an excess of $\Omega(\gamma n)$ from the stronger child targets whenever the stopping test fails. Theorem 4 proves

$$\Delta\,\mathrm{Pot} = \Omega\big(n\gamma \max\{1, (n\gamma)^2\}\big), \qquad n^2 \max\{1, \sigma_{\mathrm{out}}\} = O(\gamma^{-3}\Delta\,\mathrm{Pot}). \tag{1.1}$$

According to these bounds, a failed root round decreases potential by $\Omega(n^3/\log^3 n)$ for fixed δ. This replaces the use of Heuristic 2 of [22]. The second bound charges the next round's lfspread-dependent cost to this decrease, analogous to the drop bound of [22, Lemma 4]. The two-dimensional updates never decrease the minimum of any suffix of the QR profile. This bounds the total available potential decrease by $\frac{1}{2} n_{\mathrm{root}}(n_{\mathrm{root}} - 1)\,\mathrm{lfspread}(\mathbf{R}^{\mathrm{init}})$. The decreases telescope across overlapping calls, and the geometric decrease in block sizes controls the recursive costs. This amortization replaces the need for the logarithmic round bound from Heuristic 1 in [22].

2 Preliminaries

2.1 Notation

We use Higham's floating-point terminology [11].

- *QR factors and computed values.* We consider nonsingular square lattice bases. An exact QR factorization is $\mathbf{A} = \mathbf{QR}$, where $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$ and $\mathbf{R}$ is upper triangular with $R_{ii} > 0$. The notation fl_p denotes floating-point computation at p-bit precision. We omit the subscript when the precision is clear from context. $\mathrm{computeR}_p(\mathbf{A})$ denotes the stored triangular output of the fixed QR routine, with nonnegative diagonal entries. A tilde marks a computed quantity, and the symbol without it denotes the corresponding mathematical quantity.
- *Dimensions and intervals.* Put $[n] = \{1, \ldots, n\}$. An unsubscripted n is the order of the current matrix or call. For a named index block A, put $n_A = |A|$. The root dimension is n_{root}. Matrix intervals are nonempty and contiguous unless stated otherwise. The symbol $\mathbf{A}_i$ denotes column i.
- *Norms and entrywise operations.* For matrices, $\|\cdot\|_2$ is the spectral norm and $\|\cdot\|_1$ is the induced 1-norm. For vectors, they are the Euclidean and ℓ_1 norms. The Frobenius norm $\|\cdot\|_F$ is the square root of the sum of squared entries. The norm $\|\cdot\|_S$ is the sum of absolute entries, and $\|\cdot\|_{\max}$ is their maximum. Absolute values and matrix rounding are entrywise. The notation $\lfloor x \rceil$ means nearest-integer rounding with half ties toward $+\infty$.
- *Lattices and asymptotic bounds.* For a nonsingular matrix $\mathbf{A}$, put $\mathcal{L}(\mathbf{A}) = \mathbf{A}\mathbb{Z}^n$. The ith successive minimum $\lambda_i(\mathcal{L}(\mathbf{A}))$ is the smallest radius of a ball centered at zero containing i independent lattice vectors, and $\det(\mathcal{L}(\mathbf{A}))$ is the fundamental volume. We fix a stable matrix multiplication algorithm using $O(n^\omega)$ arithmetic operations, with $\omega \in (2, 3]$. All logarithms are to base two.

2.2 Input representation and arithmetic

We use the standard word RAM model with multiple-precision arithmetic, as in [15, Section 2.1]. Exponents are stored separately and processed exactly. A scalar operation at p-bit precision costs $\widetilde{O}(p)$ bit operations.

The input is an integer basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$ stored in memory, with direct access to each nonzero entry's sign, exponent $\lfloor \log_2 |B_{ij}| \rfloor$, and leading binary digits. Truncating to the first t significand bits costs $\widetilde{O}(t)$ and gives relative error at most 2^{1-t}.

Internal and output entries, including those of the exact integer matrix $\mathbf{U}$, use the dyadic form $z2^e$ with $z, e \in \mathbb{Z}$. The stated costs include input access and output construction in this representation. Appendix F.2 gives the encoding and bit-cost details.

2.3 Lattice bases and quality measures

For a positive-diagonal triangular matrix $\mathbf{R}$ of order n, define its *forward spread* and *logarithmic forward spread* by

$$\operatorname{fspread}(\mathbf{R}) := \max_{i \le j} \frac{R_{ii}}{R_{jj}}, \qquad \operatorname{lfspread}(\mathbf{R}) := \log_2 \operatorname{fspread}(\mathbf{R}).$$

Its *log profile* (or QR profile) is $(\log_2 R_{11}, \dots, \log_2 R_{nn})$, and its potential is

$$\operatorname{Pot}(\mathbf{R}) := \sum_{i=1}^{n-1} (n-i) \log_2 R_{ii}.$$

We compare computed and exact bases using the similarity of [22, Definition 4], with exact inequalities.

Definition 1 (Basis similarity). *For nonsingular $\mathbf{C}, \widetilde{\mathbf{C}} \in \mathbb{R}^{n \times n}$ and $\varepsilon \ge 0$, write $\mathbf{C} \approx_\varepsilon \widetilde{\mathbf{C}}$ if some orthogonal $\mathbf{O}$ satisfies*

$$\|(\mathbf{C} - \mathbf{O}\widetilde{\mathbf{C}})\mathbf{x}\|_2 \le \varepsilon \|\mathbf{C}\mathbf{x}\|_2 \qquad (\mathbf{x} \in \mathbb{R}^n).$$

Lemma 1 (Properties of similarity). *Similarity is invariant under a common invertible right multiplier, and*

$$\mathbf{A} \approx_\varepsilon \mathbf{B}, \quad \mathbf{B} \approx_\eta \mathbf{C} \quad \Longrightarrow \quad \mathbf{A} \approx_{\varepsilon+\eta+\varepsilon\eta} \mathbf{C}. \tag{2.1}$$

If $\mathbf{C} \approx_\varepsilon \widetilde{\mathbf{C}}$ with $\varepsilon < 1$, then $\widetilde{\mathbf{C}} \approx_{\varepsilon/(1-\varepsilon)} \mathbf{C}$. Replacing $\mathbf{C}$ by $\widetilde{\mathbf{C}}$ changes corresponding vector lengths and Gram–Schmidt lengths by factors between $1-\varepsilon$ and $1+\varepsilon$.

Proof. Right invariance follows by substituting the transformed coefficients. After orthogonal alignment, the triangle inequality gives the length bounds, the reverse bound $\varepsilon/(1-\varepsilon)$, and the composition bound $\varepsilon + \eta(1+\varepsilon)$. For Gram–Schmidt lengths, minimize the vector-length bounds over $\mathbf{c}_i - \sum_{j<i} x_j \mathbf{c}_j$. □

Theorem 2 bounds Gram–Schmidt lengths $\|\mathbf{c}_i^*\|_2$. For $i = 1$, $\mathbf{c}_1^* = \mathbf{c}_1$, so it gives $\|\mathbf{c}_1\|_2 \le 2^{(a_{\mathrm{LLL}}+\delta)(n-1)} \lambda_1(\mathcal{L}(\mathbf{B}))$. For $i > 1$, the bound applies to the projected vector $\mathbf{c}_i^*$.

For this exact output basis, Babai's nearest-plane algorithm has approximation factor $\sqrt{n}\,\operatorname{fspread}(\mathbf{R})$ in exact arithmetic. Appendix F.3 gives the precision and input-accuracy requirements for achieving factor $(1+\tau)\sqrt{n}\,\operatorname{fspread}(\mathbf{R})$ with any $0 < \tau \le 1$ in finite precision.

2.4 Column sorting

Theorem 3 (Column sorting bounds forward spread). *Let $\mathbf{C} = \mathbf{B}\mathbf{P}$ be a column permutation of a lattice basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$, with positive-diagonal R-factor $\mathbf{R}$. If, for some $\varepsilon \ge 0$,*

$$\|\mathbf{C}_i\|_2 \le (1+\varepsilon)\|\mathbf{C}_j\|_2 \qquad (1 \le i \le j \le n),$$

then its forward spread satisfies

$$\operatorname{fspread}(\mathbf{R}) \le (1+\varepsilon)\kappa_{\mathrm{col}}(\mathbf{B}).$$

The permutation also preserves the lattice and κ_{col}.

Proof. Column permutation preserves the lattice and κ_{col}. For any positive diagonal $\mathbf{D}$, the R-factor of $\mathbf{CD}$ is $\mathbf{RD}$. Column ordering and $R_{ii} \le \|\mathbf{C}_i\|_2$ give

$$\frac{R_{ii}}{R_{jj}} \le (1+\varepsilon)\frac{\|\mathbf{C}_j\|_2}{R_{jj}} \le (1+\varepsilon)\kappa_2(\mathbf{CD}) \qquad (i \le j),$$

since $D_{jj}\|\mathbf{C}_j\|_2 \le \|\mathbf{CD}\|_2$ and $(R_{jj}D_{jj})^{-1} \le \|(\mathbf{CD})^{-1}\|_2$. Taking the infimum over $\mathbf{D}$ and the maximum over $i \le j$ proves the claim. □

For initialization, sort by nondecreasing squared norm estimates $\frac{1}{2}\|\mathbf{B}_j\|_2^2 \le s_j \le \|\mathbf{B}_j\|_2^2$, breaking ties by the original index. After power-of-two column scaling, truncate entry magnitudes and sum their squares with downward rounding at $O(\log(2n))$-bit precision, then rescale to obtain s_j. The cost is $\widetilde{O}(n^2)$ bit operations, including sorting. Theorem 3 then applies with $1+\varepsilon = \sqrt{2}$, giving initial forward spread at most $\sqrt{2}\,\kappa_{\mathrm{col}}(\mathbf{B})$.

3 Recursive reduction with the scaled representation

The recursion acts on a positive-diagonal upper-triangular factor $\mathbf{R}$. Write $\mathbf{R}_I$ for its principal block on an interval I and $n_{\mathrm{root}} \ge 2$ for the input order.

REDUCE reduces an arbitrary block, while MERGE reduces the union of two adjacent blocks that are already reduced. Algorithm 2 sketches MERGE. Its middle-block call reduces across the boundary between the input blocks. Its left- and right-block calls restore reduction within each side.

In the sketch, GRIDSEYSEN bounds the scaled factor and its inverse to control working precision. REDUCETWO supplies the two-dimensional reduction. Each listed update is extended by the identity to the current block and passed to APPLY, which updates the stored factor and records the transformation for return composition. In the stopping test, SPREADBOUND upper-bounds the current lfspread, and $a_n^{\mathrm{stop}}(n-1)$ is the target (Section 3.2). The round limit $L(n, \bar{\sigma}_{\mathrm{entry}})$ is fixed from the entry bound on lfspread (Section 3.3). Appendix D gives the full algorithm.

3.1 Scaled representation of bases and transformations

We store basis factors and integer transformations, called *updates*, using diagonal column scales. A *power-of-two scale* is a diagonal matrix $\mathbf{E} = \operatorname{diag}(2^{e_1}, \dots, 2^{e_n}), \quad e_i \in \mathbb{Z}$.

The algorithm represents a basis with positive-diagonal R-factor $\mathbf{R}$ by the scaled pair

$$(\mathbf{R}_{\mathrm{sc}}, \mathbf{E}), \qquad \mathbf{R}_{\mathrm{sc}} = \mathbf{R}\mathbf{E}^{-1}.$$

We store the computed scaled factor at working precision and keep the integer exponents e_i separately and exactly.

Let $\mathbf{E}_0, \mathbf{E}_1$ be the scales before and after an integer update $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$. We represent $\mathbf{U}$ by the *return matrix* from $\mathbf{E}_0$ to $\mathbf{E}_1$:

$$\mathbf{M} = \mathbf{E}_0 \mathbf{U} \mathbf{E}_1^{-1}, \qquad (\mathbf{R}\mathbf{E}_0^{-1})\mathbf{M} = (\mathbf{R}\mathbf{U})\mathbf{E}_1^{-1}.$$

The second identity gives the updated basis at scale $\mathbf{E}_1$, ready for QR. Since $\mathbf{U} = \mathbf{E}_0^{-1}\mathbf{M}\mathbf{E}_1$ is integral, M_{ij} lies on the grid $((E_0)_{ii}/(E_1)_{jj})\mathbb{Z}$.

For example, take $t \in \mathbb{Z}_{\geq 0}$, $\mathbf{E}_0 = \mathbf{E}_1 = \mathrm{diag}(1, 2^t)$, and

$$\mathbf{R} = \begin{pmatrix} 1 & 2^t \\ 0 & 2^t \end{pmatrix}, \qquad \mathbf{U} = \begin{pmatrix} 1 & -2^t \\ 0 & 1 \end{pmatrix}.$$

The update gives $\mathbf{R}\mathbf{U} = \mathbf{E}_1$. Its scaled operands $\mathbf{R}_{\mathrm{sc}} = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$ and $\mathbf{M} = \left(\begin{smallmatrix} 1 & -1 \\ 0 & 1 \end{smallmatrix}\right)$ are independent of t, which is stored only in the scales.

For successive updates $\mathbf{U}_i$ with scales $\mathbf{E}_{i-1}, \mathbf{E}_i$, the corresponding return matrices compose by cancellation of intermediate scales:

$$\mathbf{M}_1 \cdots \mathbf{M}_s = \mathbf{E}_0 (\mathbf{U}_1 \cdots \mathbf{U}_s) \mathbf{E}_s^{-1}.$$

For a child block $J \subseteq A$, let $\mathbf{E}_J$ be the restriction of $\mathbf{E}_A$ to J. Extend $\mathbf{M}_J$ by the identity outside J to obtain $\mathbf{M}_A$. Replace the J block of $\mathbf{E}_A$ by $\mathbf{E}'_J$ to obtain $\mathbf{E}'_A$. Write $\mathrm{emb}_J(\mathbf{M}_J, \mathbf{E}'_J) := (\mathbf{M}_A, \mathbf{E}'_A)$. This pair represents $\mathbf{E}_J^{-1}\mathbf{M}_J\mathbf{E}'_J$ on J and the identity outside, preserving unimodularity.

Algorithm 2 MERGE(I, J) (sketch)

Input: Adjacent nonempty intervals I, J with balanced sizes ($\big||I| - |J|\big| \leq 1$) and reduced blocks
$K_{\mathrm{L}}, K_{\mathrm{R}}$ are the left and right halves of an interval K.
For $|K| > 1$, $|K_{\mathrm{L}}| = \lfloor |K|/2 \rfloor$. For $|K| = 1$, set $K_{\mathrm{L}} = K_{\mathrm{R}} = K$.
Output: Unimodular update reducing $I \cup J$, in the scaled representation

$A \leftarrow I \cup J$, $n \leftarrow |A|$
Apply GRIDSEYSEN(A)
if $n = 2$ **then**
  Apply REDUCETWO(A, γ)
else
  $\bar{\sigma}_{\mathrm{entry}} \leftarrow$ SPREADBOUND(A)
  for $t = 1, \ldots, L(n, \bar{\sigma}_{\mathrm{entry}})$ **do**
    Apply the update from MERGE$(I_{\mathrm{R}}, J_{\mathrm{L}})$
    if $|I| > 1$ **then** apply the update from MERGE$(I_{\mathrm{L}}, I_{\mathrm{R}})$
    if $|J| > 1$ **then** apply the update from MERGE$(J_{\mathrm{L}}, J_{\mathrm{R}})$
    Apply GRIDSEYSEN(A)
    if SPREADBOUND$(A) \leq a_n^{\mathrm{stop}}(n-1)$ **then break**
  if no test succeeded **then fail with** $\perp$
Apply GRIDSEYSEN(A)
return composed update in the scaled representation

To keep the scaled diagonal near one, define the canonical power-of-two scale of a positive-diagonal triangular matrix $\mathbf{R}$ by

$$\operatorname{scale}(\mathbf{R}) := \operatorname{diag}\bigl(2^{\lfloor \log_2 R_{11} \rfloor}, \dots, 2^{\lfloor \log_2 R_{nn} \rfloor}\bigr).$$

We allow slack for rounding when a scale is selected or retained.

Definition 2 (Admissible scale). *A power-of-two scale* $\mathbf{E}$ *is admissible for* $\mathbf{R}$ *when*

$$E_{ii}/2 \le R_{ii} < 4E_{ii} \qquad (1 \le i \le n).$$

Section 3.4 bounds the norms of the scaled factors and their inverses in terms of lfspread. These bounds also control $\mathbf{M}$ and $\mathbf{M}^{-1}$ (Lemma 11). We use these norm bounds to choose the working precision for basis updates and return products.

We round computed products of return matrices back to their grids by *grid decoding*, denoted by GridDecode. At the precision specified in Appendix C.1, the decoded matrix represents a unimodular integer update. Applying this update to the basis gives the same exact QR diagonal as applying the exact product of the original updates.

3.2 Reduction targets and profile monotonicity

Each parent has a slightly weaker target than its children, leaving room for merging and rounding error. Fix an absolute dyadic constant $c \in (0, 1/1024]$, small enough for every absolute inequality imposed below.

For a root order $n_{\mathrm{root}} \ge 2$, set

$$h_{\mathrm{root}} = \lceil \log_2 n_{\mathrm{root}} \rceil, \qquad \gamma = \frac{\delta}{2h_{\mathrm{root}}},$$
$$a_n = a_{\mathrm{LLL}} + \gamma \lceil \log_2 n \rceil \quad (1 \le n \le n_{\mathrm{root}}).$$

Consequently, for $2 \le n \le n_{\mathrm{root}}$, $a_n - a_{\lceil n/2 \rceil} = \gamma$ and $a_{n_{\mathrm{root}}} = a_{\mathrm{LLL}} + \delta/2$. Let $\mathcal{T}(I)$ consist of I and its recursive halves, stopping at singletons. A factor is *profile-reduced on* I when

$$\operatorname{lfspread}(\mathbf{R}_J) \le \bigl(a_{|J|} - c\gamma\bigr)\,(|J| - 1) \qquad \bigl(J \in \mathcal{T}(I)\bigr). \tag{3.1}$$

Adjacent nonempty intervals are *balanced* when their sizes differ by at most one.

Stopping criterion. A merge returns when its lfspread meets the target with a numerical margin. For an interval of order $n \ge 1$, put $\varepsilon_{\mathrm{stop}} := \min\{c, c^2\gamma \max\{1, n-1\}\}$. For $n \ge 2$, fix a dyadic stopping coefficient a_n^{stop} with

$$a_n - 3c\gamma \le a_n^{\mathrm{stop}} \le a_n - 2c\gamma.$$

The stopping test uses $\bar{\sigma} = \textsc{SpreadBound}(I)$, required to satisfy

$$\sigma \le \bar{\sigma} \le \sigma + \varepsilon_{\mathrm{stop}}, \qquad \sigma = \operatorname{lfspread}(\mathbf{R}_I), \tag{3.2}$$

where $\mathbf{R}$ is the current exact factor from this call's supplied input. The test accepts when $\bar{\sigma} \le a_n^{\mathrm{stop}}(n-1)$. The construction in (D.4) obtains this bound from the stored factor and its scale.

Definition 3 (Profile monotonicity). *Let $\widetilde{\mathbf{R}}$ be positive-diagonal upper triangular of order n and let $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$. For $0 \le \varepsilon < 1$, the pair $(\widetilde{\mathbf{R}}, \mathbf{U})$ is* profile-monotone at tolerance ε *if, for every positive-diagonal upper-triangular $\mathbf{R}$ with $\mathbf{R} \approx_\varepsilon \widetilde{\mathbf{R}}$, the transition to the positive-diagonal R-factor $\mathbf{R}'$ of $\mathbf{RU}$ satisfies the following profile inequalities. Writing $x_i = \log_2 R_{ii}$ and $y_i = \log_2 R'_{ii}$, we require*

$$\sum_{i \le k} y_i \le \sum_{i \le k} x_i, \qquad \max_{i \le k} y_i \le \max_{i \le k} x_i, \qquad \min_{i \ge k} y_i \ge \min_{i \ge k} x_i \tag{3.3}$$

for every $1 \le k \le n$.

These inequalities make both lfspread and the potential nonincreasing. Each exact-arithmetic LLL update satisfies them [18].

Definition 4 (Exact QR chain). *Fix a positive-diagonal triangular input $\mathbf{R}_0$ and ordered updates $\mathbf{V}_1, \mathbf{V}_2, \ldots \in \mathrm{GL}_n(\mathbb{Z})$. Put $\mathbf{U}_0 = \mathbf{I}$ and $\mathbf{U}_t = \mathbf{V}_1 \cdots \mathbf{V}_t$. The exact QR chain from $\mathbf{R}_0$ consists of the positive-diagonal factors defined by*

$$\mathbf{R}_0 \mathbf{U}_t = \mathbf{Q}_t \mathbf{R}_t, \qquad \mathbf{Q}_t^T \mathbf{Q}_t = \mathbf{I}.$$

The update from $\mathbf{R}_s$ to $\mathbf{R}_t$ is $\mathbf{U}_s^{-1}\mathbf{U}_t = \mathbf{V}_{s+1} \cdots \mathbf{V}_t$.

3.3 Potential decrease in a merge round

To bound the number and cost of merge rounds, we measure their potential decrease on exact factors from a common input.

The potential analysis below assumes (3.3) for every exact QR transition. These inequalities bound the total potential decrease in terms of the initial lfspread.

Lemma 2 (Potential budget and termination). *Let $\mathbf{R}_0, \mathbf{R}_1, \ldots$ be an exact QR chain of order $n \ge 3$ whose transitions satisfy the profile inequalities (3.3), with $\mathrm{lfspread}(\mathbf{R}_0) \le \bar{\sigma}$ for some $\bar{\sigma} \ge 0$. Its total potential decrease satisfies*

$$\mathrm{Pot}(\mathbf{R}_0) - \mathrm{Pot}(\mathbf{R}_t) \le \frac{n(n-1)}{2}\bar{\sigma}, \tag{3.4}$$

for every $t \ge 0$ for which $\mathbf{R}_t$ is defined. Suppose each transition is a completed round followed by a stopping test. The call returns on success. For some $G > 0$, assume every failed round satisfies

$$\mathrm{Pot}(\mathbf{R}_{t-1}) - \mathrm{Pot}(\mathbf{R}_t) \ge G.$$

Then at most

$$\left\lceil \frac{n(n-1)\max\{1, \bar{\sigma}\}}{2G} \right\rceil$$

tests fail.

Proof. Write $x_i^{(k)} = \log_2(R_k)_{ii}$ and $x_i^{\min} = \min_{j \geq i} x_j^{(0)}$. The suffix-minimum bound in (3.3) gives $x_i^{(t)} \geq x_i^{\min}$, and the initial lfspread bound gives $x_i^{(0)} - x_i^{\min} \leq \bar{\sigma}$. Therefore

$$\mathrm{Pot}(\mathbf{R}_0) - \mathrm{Pot}(\mathbf{R}_t) \leq \sum_{i<n} (n-i)(x_i^{(0)} - x_i^{\min}) \leq \frac{n(n-1)}{2}\bar{\sigma},$$

which proves (3.4). Each failed round consumes at least G of this budget, proving the failure bound. □

After the side blocks meet their stronger targets, any remaining violation must cross their boundary. The next lemma bounds the resulting lfspread by the three block bounds and the total potential decrease.

Lemma 3 (Forward spread from potential decrease). *Consider exact QR transitions $\mathbf{R}^{\mathrm{in}} \to \mathbf{R}^{\mathrm{mid}} \to \mathbf{R}^{\mathrm{out}}$ of order $n = n_{\mathrm{L}} + n_{\mathrm{R}}$ with successive updates $\mathbf{U}_1, \mathbf{U}_2 \in \mathrm{GL}_n(\mathbb{Z})$. Assume both transitions satisfy the profile inequalities (3.3). Choose $1 \leq b_{\mathrm{L}} \leq n_{\mathrm{L}}$ and $1 \leq b_{\mathrm{R}} \leq n_{\mathrm{R}}$, and set*

$$I_{\mathrm{mid}} = [n_{\mathrm{L}} - b_{\mathrm{L}} + 1, n_{\mathrm{L}} + b_{\mathrm{R}}].$$

Suppose $R_{ii}^{\mathrm{mid}} = R_{ii}^{\mathrm{in}}$ for $i \notin I_{\mathrm{mid}}$. Let $\bar{\sigma}_{\mathrm{left}}, \bar{\sigma}_{\mathrm{mid}}, \bar{\sigma}_{\mathrm{right}} \geq 0$ satisfy

$$\begin{aligned} \max\Big\{\mathrm{lfspread}(\mathbf{R}^{\mathrm{in}}_{[1,n_{\mathrm{L}}]}), \mathrm{lfspread}(\mathbf{R}^{\mathrm{out}}_{[1,n_{\mathrm{L}}]})\Big\} &\leq \bar{\sigma}_{\mathrm{left}}, \\ \mathrm{lfspread}(\mathbf{R}^{\mathrm{mid}}_{I_{\mathrm{mid}}}) &\leq \bar{\sigma}_{\mathrm{mid}}, \\ \max\Big\{\mathrm{lfspread}(\mathbf{R}^{\mathrm{in}}_{[n_{\mathrm{L}}+1,n]}), \mathrm{lfspread}(\mathbf{R}^{\mathrm{out}}_{[n_{\mathrm{L}}+1,n]})\Big\} &\leq \bar{\sigma}_{\mathrm{right}}. \end{aligned}$$

Then

$$\mathrm{lfspread}(\mathbf{R}^{\mathrm{out}}) \leq \frac{2\big(\mathrm{Pot}(\mathbf{R}^{\mathrm{in}}) - \mathrm{Pot}(\mathbf{R}^{\mathrm{out}})\big)}{\min\{b_{\mathrm{L}}, b_{\mathrm{R}}\}^2} + \bar{\sigma}_{\mathrm{left}} + \bar{\sigma}_{\mathrm{mid}} + \bar{\sigma}_{\mathrm{right}}. \tag{3.5}$$

Proof. We bound the potential decrease using the gaps between the input and middle profiles on the two sides. Write $r_i^{\mathrm{in}} = \log_2 R_{ii}^{\mathrm{in}}$, $r_i^{\mathrm{mid}} = \log_2 R_{ii}^{\mathrm{mid}}$, and $r_i^{\mathrm{out}} = \log_2 R_{ii}^{\mathrm{out}}$. Unimodularity and the unchanged diagonal outside I_{mid} give $\sum_{i \in I_{\mathrm{mid}}} r_i^{\mathrm{in}} = \sum_{i \in I_{\mathrm{mid}}} r_i^{\mathrm{mid}}$. Put $\Delta\,\mathrm{Pot} = \mathrm{Pot}(\mathbf{R}^{\mathrm{in}}) - \mathrm{Pot}(\mathbf{R}^{\mathrm{out}})$. Put $b = \min\{b_{\mathrm{L}}, b_{\mathrm{R}}\}$, $i_{\mathrm{L}} = n_{\mathrm{L}} - b_{\mathrm{L}} + 1$, $i_{\mathrm{R}} = n_{\mathrm{L}} + b_{\mathrm{R}}$, and

$$\begin{aligned} G_{\mathrm{L}} &= \max\left\{\min_{i_{\mathrm{L}} \leq i \leq n_{\mathrm{L}}} r_i^{\mathrm{in}} - \max_{i_{\mathrm{L}} \leq i \leq n_{\mathrm{L}}} r_i^{\mathrm{mid}},\, 0\right\}, \\ G_{\mathrm{R}} &= \max\left\{\min_{n_{\mathrm{L}} < j \leq i_{\mathrm{R}}} r_j^{\mathrm{mid}} - \max_{n_{\mathrm{L}} < j \leq i_{\mathrm{R}}} r_j^{\mathrm{in}},\, 0\right\}. \end{aligned}$$

For $j < n$, let $\Delta_j^{\mathrm{cut}} = \sum_{i \leq j}(r_i^{\mathrm{in}} - r_i^{\mathrm{out}})$. The transition to $\mathbf{R}^{\mathrm{mid}}$ preserves the sum of log diagonals and leaves the diagonals outside I_{mid} unchanged. Together with profile monotonicity, this gives, for $1 \leq t \leq b$,

$$\Delta_{i_{\mathrm{L}}+t-1}^{\mathrm{cut}} \geq tG_{\mathrm{L}}, \qquad \Delta_{i_{\mathrm{R}}-t}^{\mathrm{cut}} \geq \sum_{i_{\mathrm{R}}-t < i \leq i_{\mathrm{R}}} (r_i^{\mathrm{mid}} - r_i^{\mathrm{in}}) \geq tG_{\mathrm{R}}.$$

The suffix sum equals the corresponding nonnegative prefix difference. The first $b-1$ cuts from each end are disjoint, and an additional cut contributes at least $b\max\{G_{\mathrm{L}}, G_{\mathrm{R}}\}$. Hence

$$\Delta\,\mathrm{Pot} = \sum_{j<n} \Delta_j^{\mathrm{cut}} \geq \frac{b^2}{2}(G_{\mathrm{L}} + G_{\mathrm{R}}).$$

To bound $\mathrm{lfspread}(\mathbf{R}^{\mathrm{out}})$, put

$$L_m = \max_{i_{\mathrm{L}} \leq i \leq n_{\mathrm{L}}} r_i^{\mathrm{mid}}, \qquad R_m = \min_{n_{\mathrm{L}} < j \leq i_{\mathrm{R}}} r_j^{\mathrm{mid}}.$$

The unchanged diagonals outside I_{mid} and the lfspread bounds for both sides give

$$\max_{i \leq n_{\mathrm{L}}} r_i^{\mathrm{mid}} \leq L_m + \bar{\sigma}_{\mathrm{left}} + G_{\mathrm{L}}, \qquad \min_{j > n_{\mathrm{L}}} r_j^{\mathrm{mid}} \geq R_m - \bar{\sigma}_{\mathrm{right}} - G_{\mathrm{R}}.$$

Since $L_m - R_m \leq \bar{\sigma}_{\mathrm{mid}}$, profile monotonicity bounds every final log-diagonal difference across n_{L} by $\bar{\sigma}_{\mathrm{left}} + \bar{\sigma}_{\mathrm{mid}} + \bar{\sigma}_{\mathrm{right}} + G_{\mathrm{L}} + G_{\mathrm{R}}$. Every difference within one side satisfies the corresponding side bound. Combining this bound with the potential estimate proves (3.5). □

A merge round of order n means one round of MERGE in Algorithm 2. Its *middle-block recursion path* starts with $I_{\mathrm{R}} \cup J_{\mathrm{L}}$ and repeatedly follows the first recursive MERGE child through order two. For an order-16 round split between indices 8 and 9, these blocks are $[5,12] \supset [7,10] \supset [8,9]$. The proof selects a block A among these existing calls. The updates along this path before its invocation are all diagonal-preserving GRIDSEYSEN updates. Applying them and the selected call's decoded return leaves the QR diagonal outside A and the sum of log diagonals on A unchanged.

Theorem 4 (Potential decrease in one merge round). *Let $\mathbf{R}^{\mathrm{before}}, \mathbf{R}^{\mathrm{after}}$ be positive-diagonal upper-triangular matrices of order $n = n_{\mathrm{L}} + n_{\mathrm{R}} \geq 3$, split after position n_{L}, and let $I = [1, n_{\mathrm{L}}]$, $J = [n_{\mathrm{L}}+1, n]$, and $0 < \gamma \leq a < 1$. Assume $|n_{\mathrm{L}} - n_{\mathrm{R}}| \leq 1$. Consider a merge round starting at $\mathbf{R}^{\mathrm{before}}$. Use this matrix as the common input for all exact QR factors below, with block updates embedded in the full matrix. Assume (3.3) for every recorded update, including those within descendant calls. The complete round update $\mathbf{U}_{\mathrm{round}}$ gives $\mathbf{R}^{\mathrm{after}}$ as the exact QR factor of $\mathbf{R}^{\mathrm{before}}\mathbf{U}_{\mathrm{round}}$. Assume:*

1. *the two side blocks at the start and end of the round satisfy*

$$\max\left\{\mathrm{lfspread}(\mathbf{R}^{\mathrm{before}}_{[1,n_{\mathrm{L}}]}), \mathrm{lfspread}(\mathbf{R}^{\mathrm{after}}_{[1,n_{\mathrm{L}}]})\right\} \leq (a-\gamma)(n_{\mathrm{L}}-1),$$
$$\max\left\{\mathrm{lfspread}(\mathbf{R}^{\mathrm{before}}_{[n_{\mathrm{L}}+1,n]}), \mathrm{lfspread}(\mathbf{R}^{\mathrm{after}}_{[n_{\mathrm{L}}+1,n]})\right\} \leq (a-\gamma)(n_{\mathrm{R}}-1).$$

2. *for every block A on the middle-block recursion path, its exact returned factor, measured from $\mathbf{R}^{\mathrm{before}}$, has logarithmic forward spread at most $a(|A|-1)$.*

3. the final factor satisfies

$$\operatorname{lfspread}(\mathbf{R}^{\text{after}}) > (a - 4c\gamma)(n-1).$$

Then

$$\begin{aligned} \operatorname{Pot}(\mathbf{R}^{\text{before}}) - \operatorname{Pot}(\mathbf{R}^{\text{after}}) &\geq cn\gamma \max\{1, (n\gamma)^2\}, \\ n^2 \operatorname{lfspread}(\mathbf{R}^{\text{after}}) &= O\big(\gamma^{-3}\big(\operatorname{Pot}(\mathbf{R}^{\text{before}}) - \operatorname{Pot}(\mathbf{R}^{\text{after}})\big)\big)\,. \end{aligned} \tag{3.6}$$

Proof. Put $\Delta \operatorname{Pot} = \operatorname{Pot}(\mathbf{R}^{\text{before}}) - \operatorname{Pot}(\mathbf{R}^{\text{after}})$. Along the middle-block recursion path, put $b_{\mathrm{L}} = |A \cap I|$ and $b_{\mathrm{R}} = |A \cap J|$. The balanced split ensures $|b_{\mathrm{L}} - b_{\mathrm{R}}| \leq 1$. If $\gamma(n-1) \geq 8$, choose the first A for which both sizes are at most $\gamma(n-1)/8$. The preceding block on the path has a side larger than $\gamma(n-1)/8$. The balanced split makes each side of A at least one quarter of that larger side. Therefore

$$\frac{\gamma(n-1)}{32} < b_{\mathrm{L}}, b_{\mathrm{R}} \leq \frac{\gamma(n-1)}{8}.$$

If $\gamma(n-1) < 8$, choose the order-two block formed by the two leaves adjacent to the boundary, so $b_{\mathrm{L}} = b_{\mathrm{R}} = 1$ and $n\gamma < 9$. In both cases, $b := \min\{b_{\mathrm{L}}, b_{\mathrm{R}}\} = \Theta(\max\{1, n\gamma\})$.

Apply Lemma 3 with $\mathbf{R}^{\text{in}} = \mathbf{R}^{\text{before}}$, $\mathbf{R}^{\text{out}} = \mathbf{R}^{\text{after}}$, and $I_{\text{mid}} = A$. Form $\mathbf{U}_{\text{mid}}$ from the updates preceding the call on A along this path, in ancestor-to-descendant order, followed by the update $\mathbf{U}_A$ returned by the call on A, embedding each update in the round's block. Let $\mathbf{R}^{\text{mid}}$ be the exact QR factor of $\mathbf{R}^{\text{before}}\mathbf{U}_{\text{mid}}$. Its A block is the exact returned factor in condition 2. The path construction gives the unchanged diagonal outside A.

Both transitions satisfy the profile inequalities: they consist of the updates already performed and the remaining operations of each pending call, with its diagonal-preserving return correction. Conditions 1 and 2 give the three lfspread bounds, whose sum is

$$\bar{\sigma}_{\text{sum}} = (n-2)(a-\gamma) + (b_{\mathrm{L}} + b_{\mathrm{R}} - 1)a.$$

Thus

$$\operatorname{lfspread}(\mathbf{R}^{\text{after}}) \leq 2\Delta \operatorname{Pot}/b^2 + \bar{\sigma}_{\text{sum}}.$$

Put $t = \gamma(n-1)$. The gap forced by the third hypothesis is

$$(a - 4c\gamma)(n-1) - \bar{\sigma}_{\text{sum}} = a(2 - b_{\mathrm{L}} - b_{\mathrm{R}}) + \gamma(n-2) - 4c\gamma(n-1).$$

In the large case, $b_{\mathrm{L}} + b_{\mathrm{R}} \leq t/4$ and $\gamma \leq a < 1$, so this is at least $(3/4 - 4c)t$. In the small case, $b_{\mathrm{L}} + b_{\mathrm{R}} = 2$ and $\gamma(n-2) \geq t/2$, so it is at least $(1/2 - 4c)t$. In either case the gap is $\Omega(n\gamma)$, uniformly in c, so the centered-block inequality gives

$$\Delta \operatorname{Pot} = \Omega(b^2 n\gamma) = \Omega\big(n\gamma \max\{1, (n\gamma)^2\}\big)\,.$$

The choice of c gives the first bound in (3.6). Finally, $n^2/b^2 = O(\gamma^{-2})$ and $n^3 = O(\gamma^{-3}\Delta \operatorname{Pot})$. Multiplying $\operatorname{lfspread}(\mathbf{R}^{\text{after}}) \leq 2\Delta \operatorname{Pot}/b^2 + O(n)$ by n^2 therefore proves the second bound in (3.6). □

For an interval I of order $n \geq 3$ and any upper bound $\bar{\sigma} \geq \operatorname{lfspread}(\mathbf{R}_I)$, define the guaranteed decrease and iteration limit by

$$G_n := cn\gamma \max\{1, (n\gamma)^2\},$$
$$L(n, \bar{\sigma}) := 1 + \left\lceil \frac{n(n-1)\max\{1, \bar{\sigma}\}}{2G_n} \right\rceil.$$

Theorem 4 gives at least G_n potential decrease per failed round. Lemma 2 then bounds the number of failed tests by $L(n, \bar{\sigma}) - 1$. The leading 1 counts the successful test.

3.4 Reducing scaled factors with GridSeysen

The operation GridSeysen adapts Seysen reduction [24] to the scaled representation, following Algorithm 7 of [15]. The prefix "Grid" refers to rounding update entries to the grids determined by the column scales. The operation bounds the scaled factor and its inverse by enforcing the following condition.

Definition 5 (Seysen reduction). *For $\eta \geq 0$, a nonsingular upper-triangular matrix $\mathbf{T}$ of order n is Seysen-reduced with parameter η if it has order one, or if the split after its first $\lfloor n/2 \rfloor$ indices satisfies*

$$\mathbf{T} = \begin{pmatrix} \mathbf{T}_{\text{left}} & \mathbf{T}_{\text{cross}} \\ 0 & \mathbf{T}_{\text{right}} \end{pmatrix}, \qquad \|\mathbf{T}_{\text{left}}^{-1}\mathbf{T}_{\text{cross}}\|_{\max} \leq \eta,$$

and both diagonal blocks satisfy the same condition. The pair $(\mathbf{R}, \mathbf{E})$ is Seysen-reduced with parameter η when $\mathbf{R}\mathbf{E}^{-1}$ is Seysen-reduced with that parameter.

Seysen reduction bounds the scaled factor and its inverse by a product over the recursion levels. For $\eta \geq 1/2$, put

$$S_\eta(n) = \eta^{\lceil \log_2 n \rceil} 2^{\left(\lceil \log_2 \frac{n}{2} \rceil + 1\right)}. \tag{3.7}$$

Lemma 4 (Norm bounds for Seysen reduction). *Every unit-diagonal upper-triangular matrix $\mathbf{W}$ of order n that is Seysen-reduced with parameter $\eta \geq 1/2$ satisfies*

$$\max\{\|\mathbf{W}\|_{\max}, \|\mathbf{W}^{-1}\|_{\max}\} \leq S_\eta(n).$$

Let $\mathbf{E}$ be admissible for a positive-diagonal triangular matrix $\mathbf{R}$ of order n. If $(\mathbf{R}, \mathbf{E})$ is Seysen-reduced with parameter $\eta \geq 1/2$, then

$$\max\{\|\mathbf{R}\mathbf{E}^{-1}\|_1, \|\mathbf{E}\mathbf{R}^{-1}\|_1\} = O(nS_\eta(n)), \qquad \kappa_{\text{col}}(\mathbf{R}) = O\big(n^2 S_\eta(n)^2\big).$$

For every contiguous interval $I \subseteq [n]$, put $\mathbf{R}_I = \mathbf{R}_{I,I}$ and $\mathbf{E}_I = \mathbf{E}_{I,I}$. Then

$$\max\{\|\mathbf{R}_I\mathbf{E}_I^{-1}\|_1, \|\mathbf{E}_I\mathbf{R}_I^{-1}\|_1\} = O(|I|S_\eta(n)).$$

Appendix F.1 proves the lemma. For working precision, take logarithms of these bounds. If $\mathbf{E}$ is admissible and $(\mathbf{R}, \mathbf{E})$ is Seysen-reduced with parameter $\eta = O(2^{\bar{\sigma}})$, the lemma gives

$$\log_2 \max\{1, \|\mathbf{R}\mathbf{E}^{-1}\|_1, \|\mathbf{E}\mathbf{R}^{-1}\|_1, \kappa_{\mathrm{col}}(\mathbf{R})\} = O((1+\bar{\sigma})\log(2n) + \log^2(2n)).$$

For the stored scaled factor $\mathbf{T} = \widetilde{\mathbf{R}}_{\mathrm{sc}}$ of order n, at precision p, GridSeysen (Algorithm 4) computes a return matrix and a stored scaled factor, retaining the input scale $\mathbf{E}$. The return matrix copies both child return matrices into its diagonal blocks and has a zero lower-left block. Only the new upper-right block is grid-decoded (Appendix C.3).

Write $\mathbf{R} = \mathbf{T}\mathbf{E}$ for the input factor. At a recursive split, write $\mathbf{E} = \mathrm{diag}(\mathbf{E}_L, \mathbf{E}_R)$. If $\mathbf{E}$ is admissible and $\mathrm{lfspread}(\mathbf{R}) \le \bar{\sigma}$, then $(E_L)_{ii}/(E_R)_{jj} < 8\,2^{\bar{\sigma}}$. For the exact scaled output, write $\mathbf{T}_L, \mathbf{T}_X, \mathbf{T}_R$ for the blocks at any recursive split. At the precision specified in Theorem 10,

$$\|\mathbf{T}_L^{-1}\mathbf{T}_X\|_{\max} \le \frac{1}{2}\max_{i,j}\frac{(E_L)_{ii}}{(E_R)_{jj}} + 1. \tag{3.8}$$

Thus the output is Seysen-reduced with parameter $C_{\mathrm{S}}2^{\bar{\sigma}}$, where we fix a sufficiently large absolute dyadic constant $C_{\mathrm{S}} \ge 8$ for the norm bounds.

3.5 Updates and return composition

Each Reduce or Merge call on A stores $(\widetilde{\mathbf{R}}_{\mathrm{sc}}, \mathbf{E}_A)$, an ordered list $\mathcal{H}$ of return matrices with their output scales, and an operation counter j. Its fixed input $\mathbf{R}_A^{\mathrm{in}}$ is the parent's stored unscaled block at entry. The root uses the initial QR factor $\mathbf{R}^{\mathrm{init}}$ and its canonical scale $\mathbf{E}_{\mathrm{init}}$. Local error starts at zero, with inherited error handled by the reference comparison in Section 3.6.

Apply accepts a local routine or a completed child return. Local routines use the stored pair. ReduceTwo also receives the current lfspread bound. For GridSeysen, Apply adopts the computed factor, whose error bound includes internal recursion and decoding (Theorem 10). For every other update, Apply multiplies the current stored factor by the return matrix and recomputes QR. A child return represents one exact integer update, so only this new multiplication and QR contribute to the parent's local error. The procedure records the return matrix and output scale. A call's Apply steps are its *direct operations*. A call *prefix* is an initial sequence of these steps.

Initialization and nonidentity ReduceTwo updates select the computed factor's canonical scale. Identity updates and GridSeysen retain their input scale. A child return replaces its block's scale. Theorem 12 proves admissibility throughout.

Operation and call errors. The call tolerance $\varepsilon_{\mathrm{call}}$ controls accumulated rounding error within one call, and $\varepsilon_{\mathrm{ref}}$ controls comparisons with the exact factors used for potential. Direct operations receive summable tolerances η_j^{dir}. Set

$$\varepsilon_{\mathrm{call}} := \frac{c^5\gamma^3}{n_{\mathrm{root}}^5}, \qquad \varepsilon_{\mathrm{ref}} := \frac{c^4\gamma^3}{n_{\mathrm{root}}^2}. \tag{3.9}$$

Here c is a sufficiently small absolute constant. In a call of order n, the jth direct update uses

$$\eta_j^{\mathrm{dir}} := \min\{\varepsilon_{\mathrm{call}}/(8j^2), c\varepsilon_{\mathrm{call}}/n^2, c\varepsilon_{\mathrm{stop}}\}.$$

For a call on A, write $\mathbf{V}_t$ for its tth recorded update. Let $\mathbf{R}_{A,t}$ be the tth factor in the exact QR chain from $\mathbf{R}_A^{\mathrm{in}}$. Write $\widetilde{\mathbf{R}}_{A,t}$ for the stored unscaled factor after t direct operations. Each operation compares its new stored factor with the exact update of the preceding stored factor. The local error estimates and the budget above give

$$\begin{aligned} \widetilde{\mathbf{R}}_{A,t-1}\mathbf{V}_t &\approx_{\eta_t^{\mathrm{dir}}} \widetilde{\mathbf{R}}_{A,t}, \\ \sum_t \log_2(1+\eta_t^{\mathrm{dir}}) &\le \log_2(1+\varepsilon_{\mathrm{call}}). \end{aligned} \tag{3.10}$$

Lemma 1 bounds the accumulated error by $\prod_{j=1}^{t}(1+\eta_j^{\mathrm{dir}}) - 1 \le \varepsilon_{\mathrm{call}}$. Hence

$$\mathbf{R}_{A,t} \approx_{\varepsilon_{\mathrm{call}}} \widetilde{\mathbf{R}}_{A,t}, \tag{3.11}$$

measured from the fixed call input.

Two-dimensional updates and monotonicity. ReduceTwo rounds the input as in [23, Algorithm 1], then applies Yap's planar reduction [27]. A norm test accepts the candidate update only if its decrease in the first QR diagonal ensures profile monotonicity at the required similarity tolerance. Otherwise the routine returns the identity. Appendix C.4 gives the construction and proves Lemma 5.

Lemma 5 (Planar reduction). *For every positive-diagonal upper-triangular* $\mathbf{R}$ *of order two and* $0 < \vartheta < 1$, ReduceTwo *returns an update* $\mathbf{U} \in \mathrm{GL}_2(\mathbb{Z})$ *such that the positive-diagonal R-factor* $\mathbf{R}'$ *of* $\mathbf{RU}$ *satisfies* $\mathrm{lfspread}(\mathbf{R}') \le a_{\mathrm{LLL}} + (1-2c)\vartheta$. *The pair* $(\mathbf{R}, \mathbf{U})$ *is profile-monotone at tolerance* $c^2\vartheta/64$. *A nonidentity update satisfies* $R'_{11} \le (1-c^2\vartheta)R_{11}$.

Suppose $\mathbf{R}$ *is supplied with an admissible scale* $\mathbf{E}_0$, $\mathrm{lfspread}(\mathbf{R}) \le \bar{\sigma}$, *and* $(\mathbf{R}, \mathbf{E}_0)$ *is Seysen-reduced with parameter* $C_{\mathrm{S}}2^{\bar{\sigma}}$. *There is a sufficient precision*

$$P_2(\bar{\sigma}, \vartheta) = \widetilde{O}(1 + \bar{\sigma} + \log(1/\vartheta)).$$

At every $p \ge P_2$, *the scaled implementation computes* $(\mathbf{M}, \mathbf{E}_1)$ *with* $\mathbf{U} = \mathbf{E}_0^{-1}\mathbf{M}\mathbf{E}_1$ *in* $\widetilde{O}(p)$ *bit operations. The scale* $\mathbf{E}_1$ *is admissible for* $\mathbf{R}'$, *and* $(\mathbf{R}', \mathbf{E}_1)$ *is Seysen-reduced with parameter* $C_{\mathrm{S}}2^{\bar{\sigma}}$.

Preserving the profile when composing returns. Let $\mathbf{E}_0$ be the call's input scale and let $\mathcal{H}$ be its nonempty history of return pairs $(\mathbf{M}_i, \mathbf{E}_i)$ in execution order, for $1 \le i \le s$. To return a single pair, ComposeReturns$(\mathbf{E}_0, \mathcal{H})$ evaluates the product $\mathbf{M}_1 \cdots \mathbf{M}_s$ from right to left. Each product accumulated from the right (a *suffix*) and its inverse are bounded through their input and output factors (Lemma 11). These bounds determine the precision for the next multiplication.

Each product is decoded using the same final scale $\mathbf{E}_s$. ComposeReturns computes

$$\begin{aligned}\mathbf{T}_s &= \mathbf{M}_s,\\ \mathbf{T}_i &= \mathrm{GridDecode}\big(\mathrm{fl}_{p_i}(\mathbf{M}_i\mathbf{T}_{i+1}), \mathbf{E}_{i-1}, \mathbf{E}_s\big) \quad (i = s-1, \dots, 1),\end{aligned} \tag{3.12}$$

where p_i is the precision used for that product. The call returns $(\mathbf{T}_1, \mathbf{E}_s)$. The represented integer return is $\mathbf{U} = \mathbf{E}_0^{-1}\mathbf{T}_1\mathbf{E}_s$.

Let $\mathbf{R}_{A,s}$ be the exact factor after the call's last direct operation, and let $\mathbf{R}_A^{\mathrm{out}}$ be the exact QR factor of $\mathbf{R}_A^{\mathrm{in}}\mathbf{U}$. The composed return satisfies

$$\mathbf{R}_A^{\mathrm{out}} = \mathbf{R}_{A,s}\mathbf{W}, \qquad \mathbf{W} \text{ is upper unitriangular over } \mathbb{Z}. \tag{3.13}$$

Each decoding step multiplies the integer update on the right by an integral upper-unitriangular correction (Lemma 12). The product of these corrections is $\mathbf{W}$. Thus multiplication by $\mathbf{W}$ preserves QR diagonals, lfspread, and potential for any fixed input. The stored factor's error bound (3.11) applies before decoding. Equation (D.6) bounds the effect of decoding on the scaled norms.

Approximate basis output. Write $\mathbf{B}^{\mathrm{sort}} = \mathbf{BP}$ for the column-ordered input. Let $(\mathbf{M}_{\mathrm{loc}}, \mathbf{E}_{\mathrm{out}})$ be the root's scaled return. The final multiplication is

$$\widetilde{\mathbf{C}} = \mathrm{fl}_{p_{\mathrm{out}}}\big((\mathbf{B}^{\mathrm{sort}}\mathbf{E}_{\mathrm{init}}^{-1})\mathbf{M}_{\mathrm{loc}}\big)\,\mathbf{E}_{\mathrm{out}}. \tag{3.14}$$

The product being rounded is $\mathbf{CE}_{\mathrm{out}}^{-1}$, with $\mathbf{C} = \mathbf{BU}$ and $\mathbf{U} = \mathbf{PE}_{\mathrm{init}}^{-1}\mathbf{M}_{\mathrm{loc}}\mathbf{E}_{\mathrm{out}}$. Choose p_{out} from the available operand and inverse-norm bounds to ensure

$$\mathbf{C} \approx_{\varepsilon_{\mathrm{out}}/(64n^2)} \widetilde{\mathbf{C}}. \tag{3.15}$$

Here the similarity inequality holds with $\mathbf{O} = \mathbf{I}$. The amortized analysis below bounds this precision and the resulting cost.

3.6 Amortized complexity and output guarantees

We first assign local work to calls and rounds, then bound its cost and charge failed rounds to potential decrease.

We account for the direct work of each Reduce call and each Merge round separately from recursive work, counting a base-case Merge as one round. Each child's first GridSeysen operation and final multiplication $\mathbf{M}_1\mathbf{T}_2$ in (3.12) use the entry norm bound supplied by the parent, so we charge them to the invoking parent call or round. Charge every other return product to the call or round that recorded its left operand.

Lemma 6. *Let p_{init} be the initialization precision and $p_{\max}$ the largest working precision. Suppose the exact QR transitions from $\mathbf{R}^{\mathrm{init}}$ satisfy (3.3), their scales are admissible, and stored and corresponding exact diagonals differ by at most a factor two. Scale exponents occupy $O(1)$ machine words, as do exponents of return matrices with $O(p_{\max})$-bit significands and norm at most $2^{\widetilde{O}(p_{\max})}$.*

At precision p, on operands of order $n' \le n$ with $O(p)$-bit significands and norms at most $2^{O(p)}$, each GridSeysen, Apply, *or return product with grid decoding costs $\widetilde{O}((n')^{\omega}p)$ bit operations. Preparing operands, testing* lfspread, *and updating scales and counters cost $\widetilde{O}((n')^2 p)$. These bounds include exponent processing and memory access.*

The proof is given in Appendix F.2.

Numerical guarantees. For comparison with the exact factors used in Section 3.3, write $\mathbf{R}^{\mathrm{in}}$ for a call's supplied input and $\mathbf{R}^{\mathrm{in}}_{\mathrm{ref}}$ for the corresponding exact input block. The same recorded update $\mathbf{U}$ defines both exact factors by

$$\mathbf{R}^{\mathrm{in}}\mathbf{U} = \mathbf{Q}\mathbf{R}, \qquad \mathbf{R}^{\mathrm{in}}_{\mathrm{ref}}\mathbf{U} = \mathbf{Q}_{\mathrm{ref}}\mathbf{R}_{\mathrm{ref}}. \tag{3.16}$$

The stored factor is $\widetilde{\mathbf{R}}$. At the outer call, the two inputs coincide. For each child, $\mathbf{R}^{\mathrm{in}}_{\mathrm{ref}}$ is the corresponding block of the parent's $\mathbf{R}_{\mathrm{ref}}$ when the child starts.

Assume the following seven numerical guarantees for the execution under analysis.

1. *Local errors and return composition.* Direct operations satisfy (3.10), which implies the call-error bound (3.11). Composing returns preserves the profile as in (3.13).
2. *Profile monotonicity.* Each ReduceTwo operation's supplied input and returned update form a profile-monotone pair at tolerance $\varepsilon_{\mathrm{ref}}$.
3. *Entry and return profiles.* A Reduce(A) output is profile-reduced on A. A Merge(I, J) input and output are profile-reduced on I and J. Its output also satisfies the inequality in (3.1) on the full block $I \cup J$. These exact factors are measured from the call's own supplied input.
4. *Stopping tests.* At every tested prefix, SpreadBound satisfies (3.2).
5. *Reference errors.* At every direct-operation prefix, the stored and exact factors satisfy

$$\mathbf{R}_{\mathrm{ref}} \approx_{\varepsilon_{\mathrm{ref}}} \widetilde{\mathbf{R}}, \tag{3.17}$$

$$\max_i \left| \log_2 \frac{R_{ii}}{(\mathbf{R}_{\mathrm{ref}})_{ii}} \right| \le -\log_2(1 - 2\varepsilon_{\mathrm{ref}}). \tag{3.18}$$

   Grid decoding preserves (3.18) at return.
6. *Output norms.* At the exit of an order-n' call with exact input lfspread value σ, its exact returned factor $\mathbf{R}^{\mathrm{out}}$ and returned scale $\mathbf{E}^{\mathrm{out}}$ satisfy

$$\begin{aligned} &\max\{\|\mathbf{R}^{\mathrm{out}}(\mathbf{E}^{\mathrm{out}})^{-1}\|_1, \|\mathbf{E}^{\mathrm{out}}(\mathbf{R}^{\mathrm{out}})^{-1}\|_1\} \\ &\quad \le 2^{O((1+\min\{\sigma,n'\})\log(2n')+\log^2(2n'))}. \end{aligned} \tag{3.19}$$

7. *Working precision.* For a Reduce call or Merge round of order n' and exact input lfspread value σ, each operation charged to it uses precision $p = \widetilde{O}(\max\{1, \sigma\})$, $O(p)$-bit significands, and operand norms at most $2^{O(p)}$, allowing polylogarithmic factors in the dimensions, entry lfspread,

and $1/\gamma$. Exponent arithmetic is accounted for by Lemma 6. For ReduceTwo, $p \geq P_2(\bar{\sigma}, \gamma)$ with the scaled-input bounds of Lemma 5. The outer call's first GridSeysen operation and final multiplication $\mathbf{M}_1\mathbf{T}_2$ may add $\widetilde{O}(\log(2K_{\text{entry}}))$ bits, where K_{entry} bounds its scaled input factor and inverse.

Theorem 11 proves these guarantees for Algorithm 6, with the tolerances in (3.9) and the precision choices of Appendix D.3.

The profile-monotonicity and reference-error guarantees transfer each ReduceTwo update's profile inequalities to its reference factor. GridSeysen and grid decoding preserve exact diagonals. Induction over calls therefore gives (3.3) for every reference transition.

Local costs. Each call or round is charged $O(1)$ operations. Lemmas 6 and 5 bound this local cost, including tests and updates to scales and counters, by $\widetilde{O}((n')^\omega \max\{1, \sigma\})$. The outer call adds $\widetilde{O}(n^\omega \log(2K_{\text{entry}}))$.

Using the numerical margins. The comparison error changes a logarithmic diagonal ratio by at most $-2\log_2(1 - 2\varepsilon_{\text{ref}}) < c\gamma$. For $n' \geq 2$, a returned order-n' block with lfspread at most $(a_{n'} - c\gamma)(n' - 1)$ has lfspread at most $a_{n'}(n' - 1)$ in the parent call. Singletons have lfspread $= 0$. The entry and return guarantees, with $a_{n'} \leq a_n - \gamma$ on either side, give the side and middle-block path hypotheses of Theorem 4. If the stopping test fails, then

$$\sigma \geq \bar{\sigma} - \varepsilon_{\text{stop}} > (a_n - 3c\gamma)(n - 1) - \varepsilon_{\text{stop}} \geq (a_n - 4c\gamma)(n - 1),$$

since $\varepsilon_{\text{stop}} \leq c^2\gamma(n - 1)$. Theorem 4 therefore gives

$$\Delta\,\text{Pot} := \text{Pot}(\mathbf{R}^{\text{before}}) - \text{Pot}(\mathbf{R}^{\text{after}}) \geq G_n, \tag{3.20}$$

$$n^2 \max\{1, \sigma\} = O(\gamma^{-3}\Delta\,\text{Pot}). \tag{3.21}$$

Here $\sigma > a_{\text{LLL}}(n - 1)$, so $\max\{1, \sigma\} = O(\sigma)$ with a constant independent of c. With $\bar{\sigma}_{\text{entry}}$ bounding the call's entry lfspread, Lemma 2 allows at most $L(n, \bar{\sigma}_{\text{entry}}) - 1$ failed tests. Success gives $\sigma \leq (a_n - 2c\gamma)(n - 1)$, leaving $c\gamma(n - 1)$ for later numerical comparisons.

Theorem 5 (Amortized cost under the numerical guarantees). *Under the seven numerical guarantees above, a call on A of order n terminates, and the sum of local weights $(n')^\omega \max\{1, \sigma\}$ over its* Reduce *calls and* Merge *rounds, including descendants, is*

$$\widetilde{O}\big(n^\omega \max\{1, \bar{\sigma}_{\text{entry}}\} + \gamma^{-3} n^{\omega-2}\big(\text{Pot}(\mathbf{R}_A^{\text{in}}) - \text{Pot}(\mathbf{R}_A^{\text{out}})\big)\big)\,,$$

where n' and σ are the order and entry logarithmic forward spread of each call or round, $\mathbf{R}_A^{\text{in}}$ is the supplied input, and $\mathbf{R}_A^{\text{out}}$ is its exact returned factor defined by (3.13). Also, $\bar{\sigma}_{\text{entry}} \geq \text{lfspread}(\mathbf{R}_A^{\text{in}})$. The bit cost adds $\widetilde{O}(n^\omega \log(2K_{\text{entry}}))$ to this bound.

Proof. Termination follows from the round bound by induction on call order, with MERGE before REDUCE at equal order. We first identify each block's decrease with its contribution to the parent potential, then charge repeated work to failed rounds and sum by depth. Use the outer call A's supplied input as the reference in (3.16). For a direct update on $B = [\ell + 1, \ell + n']$, indexed within its parent P, let $\mathbf{R}, \mathbf{R}'$ be the parent's reference factors before and after the update. Their diagonals agree outside B, and unimodularity gives $\det(\mathbf{R}_B) = \det(\mathbf{R}'_B)$. The parent and block weights differ by the constant $|P| - \ell - n'$, whose contribution vanishes by this determinant equality. Hence

$$\mathrm{Pot}(\mathbf{R}) - \mathrm{Pot}(\mathbf{R}') = \mathrm{Pot}(\mathbf{R}_B) - \mathrm{Pot}(\mathbf{R}'_B). \tag{3.22}$$

Write Δ_B for a round's potential decrease on its own call's chain, and Δ_B^{ref} for the same round's decrease in the fixed reference execution. Each endpoint potential has total weight $n'(n'-1)/2$, so the comparison bound gives

$$|\Delta_B - \Delta_B^{\mathrm{ref}}| \le n'(n'-1)[-\log_2(1 - 2\varepsilon_{\mathrm{ref}})] = O((n')^2 \varepsilon_{\mathrm{ref}}). \tag{3.23}$$

Represent each REDUCE call and each MERGE round by a node, and connect it to the node where each of its child calls starts. The resulting trees start at the outer call or after a failed round. Let $H_{\mathrm{M}}(n')$ and $H_{\mathrm{R}}(n')$ bound the sum of node orders raised to ω for a tree rooted at an order-n' MERGE or REDUCE, respectively. Then

$$\begin{aligned} &H_{\mathrm{M}}(n') \le O((n')^\omega) + 3H_{\mathrm{M}}(\lceil n'/2 \rceil), \\ &H_{\mathrm{R}}(n') \le O((n')^\omega) + H_{\mathrm{R}}(\lfloor n'/2 \rfloor) + H_{\mathrm{R}}(\lceil n'/2 \rceil) + H_{\mathrm{M}}(n') \qquad (n' \ge 3). \end{aligned}$$

Constant-order base cases and $\omega \ge 2 > \log_2 3$ give $H_{\mathrm{M}}, H_{\mathrm{R}} = O((n')^\omega)$. The profile inequalities and the comparison bound control lfspread throughout a tree by its root value up to $O(\varepsilon_{\mathrm{ref}})$. The first tree therefore has weight $O(n^\omega \max\{1, \bar{\sigma}_{\mathrm{entry}}\})$.

A tree following a failed round on B starts at that round's exit lfspread, so (3.21) bounds its weight by $O(\gamma^{-3}(n')^{\omega-2}\Delta_B)$ using that round's decrease. Since $\Delta_B \ge G_{n'} \ge cn'\gamma$, the error in (3.23) is at most $\Delta_B/2$, giving $\Delta_B \le 2\Delta_B^{\mathrm{ref}}$.

Within each call, consecutive reference potentials cancel, and return decoding contributes zero. By (3.22), replacing a parent's child-return operation by all direct operations of that child preserves the summed decrease. This holds even when successive child blocks overlap, since all decreases use the reference execution from A. At each depth, the whole-call decreases therefore sum to at most the outer call's decrease: omitted operations have nonnegative reference decrease. Each call's failed rounds use disjoint direct-operation ranges, so the same bound holds for the sum of their Δ_B^{ref} at that depth.

Along each call-tree path, every edge halves the order up to rounding except at most one REDUCE-to-MERGE edge, so there are $O(\log(2n))$ depths. Summing with $(n')^{\omega-2} \le n^{\omega-2}$ proves the weight bound. The local costs derived from Lemmas 6 and 5 give the bit bound. □

Initialization guarantees. For the root call of order $n \geq 2$, initialization supplies the positive-diagonal R-factor $\mathbf{R}^{\mathrm{init}}$ of a basis $\mathbf{B}_{\mathrm{init}}$, an admissible scale $\mathbf{E}_{\mathrm{init}}$, and a computed norm bound K_{init} satisfying

$$\begin{aligned} \mathbf{B}^{\mathrm{sort}} &\approx_{\varepsilon_{\mathrm{init}}} \mathbf{B}_{\mathrm{init}}, \\ \max\{\|\mathbf{R}^{\mathrm{init}}\mathbf{E}_{\mathrm{init}}^{-1}\|_1, \|\mathbf{E}_{\mathrm{init}}(\mathbf{R}^{\mathrm{init}})^{-1}\|_1\} &\leq K_{\mathrm{init}}. \end{aligned} \tag{3.24}$$

The root uses

$$\varepsilon_{\mathrm{init}} = c^2\delta/n^2, \qquad K_{\mathrm{entry}} = K_{\mathrm{init}} \qquad (n \geq 2). \tag{3.25}$$

The root's lfspread bound $\bar{\sigma}_{\mathrm{entry}} \geq \mathrm{lfspread}(\mathbf{R}^{\mathrm{init}})$ satisfies

$$\bar{\sigma}_{\mathrm{entry}} + \log(2K_{\mathrm{entry}}) = O(1 + \log n + \log \kappa_{\mathrm{col}}(\mathbf{B})),$$

and condition estimation together with the initial QR costs

$$\widetilde{O}(n^{\omega}(1 + \log \kappa_{\mathrm{col}}(\mathbf{B}) + \log(1/\delta))) \,. \tag{3.26}$$

Appendix B.3 proves these guarantees (Lemma 10).

Theorem 6 (Main theorem). *Let $\mathbf{B} \in \mathbb{Z}^{n\times n}$, $n \geq 1$, be nonsingular and represented as in Section 2.2. Let $\delta = 2^{-k_\delta}$ and $\varepsilon_{\mathrm{out}} = 2^{-k_{\mathrm{out}}}$ for integers $k_\delta, k_{\mathrm{out}} \geqq 2$. The procedure* Root *in Algorithm 6, lines 1–9, deterministically returns $\widetilde{\mathbf{C}}$ and $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$ in dyadic representation. For $\mathbf{C} = \mathbf{BU} = \mathbf{QR}$ with positive QR diagonal,*

$$\mathrm{lfspread}(\mathbf{R}) \leq (a_{\mathrm{LLL}} + \delta)(n-1), \qquad \kappa_{\mathrm{col}}(\mathbf{R}) \leq 2^{O(n\log n)}.$$

The approximate basis $\widetilde{\mathbf{C}}$ satisfies (3.15). In particular, the exact first column satisfies $\|\mathbf{c}_1\|_2 \leq 2^{(a_{\mathrm{LLL}}+\delta)(n-1)}\lambda_1(\mathcal{L}(\mathbf{B}))$. The bit complexity is

$$\widetilde{O}\Big(n^{\omega}\Big[(1 + \log \kappa_{\mathrm{col}}(\mathbf{B}))\delta^{-3}(1 + \log(1/\delta))^{O(1)} + \log(1/\varepsilon_{\mathrm{out}})\Big]\Big) \,. \tag{3.27}$$

Proof of Theorems 6 and 2. Lemma 10 and Theorem 11 establish the numerical guarantees used below. For $n = 1$, take $\mathbf{U} = [1]$ and round the single column as in Root. The quality and output-error bounds follow directly.

Quality. Let $n \geq 2$, let $\mathbf{U}_{\mathrm{loc}}$ be the integer update returned by the root call, and let $\mathbf{R}_{\mathrm{root}}$ be the positive-diagonal QR factor of $\mathbf{R}^{\mathrm{init}}\mathbf{U}_{\mathrm{loc}}$. The profile and norm guarantees (3.1) and (3.19) give

$$\mathrm{lfspread}(\mathbf{R}_{\mathrm{root}}) \leq (a_{\mathrm{LLL}} + \delta/2)(n-1), \qquad \kappa_{\mathrm{col}}(\mathbf{R}_{\mathrm{root}}) \leq 2^{O(n\log n)}.$$

Set $\mathbf{U} = \mathbf{PU}_{\mathrm{loc}}$, and let $\mathbf{R}_{\mathrm{final}}$ be the positive-diagonal QR factor of $\mathbf{C} = \mathbf{BU}$. Similarity is preserved by the root's integer update. Lemma 1 bounds the change in each log-diagonal ratio by $\log_2((1+\varepsilon_{\mathrm{init}})/(1-\varepsilon_{\mathrm{init}})) \leq \delta(n-1)/2$, giving $\mathrm{lfspread}(\mathbf{R}_{\mathrm{final}}) \leq (a_{\mathrm{LLL}} + \delta)(n-1)$.

Among i independent vectors of $\boldsymbol{\Lambda} = \mathcal{L}(\mathbf{C})$ of length at most $\lambda_i(\boldsymbol{\Lambda})$, one lies outside the span of the first $i-1$ columns of $\mathbf{C}$. Its last nonzero integer basis

coordinate has index $j \geq i$, so $(\mathbf{R}_{\text{final}})_{jj} \leq \lambda_i(\boldsymbol{\Lambda})$. The successive-minimum claim follows from

$$\|\mathbf{c}_i^*\|_2 = (\mathbf{R}_{\text{final}})_{ii} \leq \text{fspread}(\mathbf{R}_{\text{final}})\lambda_i(\boldsymbol{\Lambda}).$$

After any common column scaling, similarity changes the condition number by at most a factor $(1+\varepsilon_{\text{init}})/(1-\varepsilon_{\text{init}})$. Thus $\kappa_{\text{col}}(\mathbf{R}_{\text{final}}) \leq 2^{O(n \log n)}$. This proves Theorem 2.

Output accuracy. For the approximate output, put $\mathbf{A} = \mathbf{B}^{\text{sort}}\mathbf{E}_{\text{init}}^{-1}$ and $\mathbf{Y} = \mathbf{A}\mathbf{M}_{\text{loc}} = \mathbf{C}\mathbf{E}_{\text{out}}^{-1}$. Initialization bounds $\|\mathbf{A}\|_2$ and $\|\mathbf{A}^{-1}\|_2$ by $\text{poly}(n)K_{\text{init}}$. Profile monotonicity and the stopping test bound $\text{lfspread}(\mathbf{R}_{\text{root}})$ by both $\bar{\sigma}_{\text{entry}}$ and $O(n)$. The sharper output-norm bound (3.19), together with initialization similarity, therefore gives

$$\log \max\{1, \|\mathbf{Y}\|_2, \|\mathbf{Y}^{-1}\|_2\} = O((1 + \min\{\bar{\sigma}_{\text{entry}}, n\}) \log(2n) + \log^2(2n)).$$

Since $\mathbf{M}_{\text{loc}} = \mathbf{A}^{-1}\mathbf{Y}$, the stable multiplication bound of Theorem 7, including operand rounding, ensures the output accuracy (3.15) at

$$p_{\text{out}} = \widetilde{O}(1 + \log K_{\text{init}} + \min\{\bar{\sigma}_{\text{entry}}, n\} + \log(1/\varepsilon_{\text{out}})).$$

Indeed, for $\boldsymbol{\Delta}_Y = (\widetilde{\mathbf{C}} - \mathbf{C})\mathbf{E}_{\text{out}}^{-1}$, this precision gives

$$\|\boldsymbol{\Delta}_Y \mathbf{Y}^{-1}\|_F \leq \|\boldsymbol{\Delta}_Y\|_F \|\mathbf{Y}^{-1}\|_2 \leq \frac{\varepsilon_{\text{out}}}{64n^2}.$$

Since $\widetilde{\mathbf{C}} = (\mathbf{I} + \boldsymbol{\Delta}_Y \mathbf{Y}^{-1})\mathbf{C}$, the required similarity holds with $\mathbf{O} = \mathbf{I}$.

Cost. Apply Theorem 5 at the root with the initial lfspread bound, total potential decrease at most $n(n-1)\bar{\sigma}_{\text{entry}}/2$, and $\gamma = \delta/(2\lceil \log_2 n \rceil)$. Each entry of $\mathbf{U}$ has an $O(p_{\max})$-bit significand (Section C.1), and the working-precision guarantee gives $p_{\max} = \widetilde{O}(1 + \log \kappa_{\text{col}}(\mathbf{B}) + \log(1/\delta))$. Thus writing $\mathbf{U}$ costs $\widetilde{O}(n^2 p_{\max})$, within the reduction bound. Output construction, including input reads and multiplication, costs $\widetilde{O}(n^\omega p_{\text{out}})$. Adding initialization and output construction gives (3.27), proving Theorem 1 for fixed δ and ε_{out}. □

4 Experiments

We compare our implementation with `flatter` on 153 q-ary, NTRU-like, scaled q-ary, and Coppersmith inputs of dimension up to 128. Our implementation uses fewer instructions in every execution. The median ratios of `flatter`'s instruction count to ours are 1.55 for ordinary q-ary inputs and 1.86 for NTRU-like inputs. Both implementations satisfy the common bounds on forward spread and column-scaled condition numbers for every output. Appendix E gives the implementation choices, setup, and results.

5 Open questions

How many polylogarithmic factors hidden by $\widetilde{O}$ can be removed? Sharing FFT transforms within matrix multiplication already reduces arithmetic overhead [8]. Sharper precision and threshold schedules may further reduce the overhead of our recursion.

What lower bounds on parallel depth hold for our output guarantees when total work stays within the sequential bound up to polylogarithmic factors? Following work on parallel LLL [10,26], we also ask how closely a parallel version of our algorithm can approach these bounds. Independent subcalls and matrix operations provide parallelism, leaving the dependencies between merge rounds to be analyzed.

References

1. Chang, X.W., Li, R.C.: Multiplicative perturbation analysis for qr factorizations. Numerical Algebra, Control and Optimization **1**(2), 301–316 (2011)
2. Chang, X., Stehlé, D., Villard, G.: Perturbation analysis of the QR factor R in the context of LLL lattice basis reduction. Math. Comput. **81**(279), 1487–1511 (2012)
3. Coppersmith, D.: Small solutions to polynomial equations, and low exponent rsa vulnerabilities. Journal of cryptology **10**(4), 233–260 (1997)
4. Coppersmith, D., Shamir, A.: Lattice attacks on NTRU. In: Fumy, W. (ed.) Advances in Cryptology – EUROCRYPT '97. Lecture Notes in Computer Science, vol. 1233, pp. 52–61. Springer (1997)
5. Demmel, J., Dumitriu, I., Holtz, O.: Fast linear algebra is stable. Numerische Mathematik **108**(1), 59–91 (Oct 2007)
6. Demmel, J., Dumitriu, I., Holtz, O., Kleinberg, R.: Fast matrix multiplication is stable. Numerische Mathematik **106**(2), 199–224 (Apr 2007)
7. Frieze, A.M., Kannan, R., Lagarias, J.C.: Linear congruential generators do not produce random sequences. In: 25th Annual Symposium on Foundations of Computer Science. pp. 480–484. IEEE Computer Society (1984)
8. Harvey, D., Van Der Hoeven, J.: On the complexity of integer matrix multiplication. Journal of Symbolic Computation **89**, 1–8 (2018)
9. Håstad, J., Just, B., Lagarias, J.C., Schnorr, C.P.: Polynomial time algorithms for finding integer relations among real numbers. SIAM Journal on Computing **18**(5), 859–881 (1989)
10. Heckler, C., Thiele, L.: Complexity analysis of a parallel lattice basis reduction algorithm. SIAM Journal on Computing **27**(5), 1295–1302 (1998)
11. Higham, N.J.: Accuracy and Stability of Numerical Algorithms. Society for Industrial and Applied Mathematics, second edn. (2002)
12. Howgrave-Graham, N.: Finding small roots of univariate modular equations revisited. In: Darnell, M. (ed.) Cryptography and Coding. Lecture Notes in Computer Science, vol. 1355, pp. 131–142. Springer (1997)
13. Howgrave-Graham, N.: Approximate integer common divisors. In: Silverman, J.H. (ed.) Cryptography and Lattices. pp. 51–66. Springer (2001)
14. Joux, A., Stern, J.: Lattice reduction: A toolbox for the cryptanalyst. Journal of Cryptology **11**(3), 161–185 (1998)
15. Kirchner, P., Espitau, T., Fouque, P.A.: Towards faster polynomial-time lattice reduction. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology – CRYPTO 2021. Lecture Notes in Computer Science, vol. 12826, pp. 760–790. Springer International Publishing, Cham (2021)
16. Koy, H., Schnorr, C.P.: Segment LLL-reduction of lattice bases. In: Silverman, J.H. (ed.) Cryptography and Lattices. pp. 67–80. Springer Berlin Heidelberg, Berlin, Heidelberg (2001)
17. Lagarias, J.C.: The computational complexity of simultaneous diophantine approximation problems. In: 23rd Annual Symposium on Foundations of Computer Science. pp. 32–39. IEEE Computer Society (1982)
18. Lenstra, A., Lenstra, H., Lovász, L.: Factoring polynomials with rational coefficients. Math. ann **261**(4), 515–534 (1982)
19. Neumaier, A., Stehlé, D.: Faster lll-type reduction of lattice bases. In: Proceedings of the 2016 ACM International Symposium on Symbolic and Algebraic Computation. p. 373–380. ISSAC '16, Association for Computing Machinery, New York, NY, USA (2016)

20. Nguyen, P.Q., Stehlé, D.: An lll algorithm with quadratic complexity. SIAM Journal on Computing **39**(3), 874–903 (2009)
21. Novocin, A., Stehlé, D., Villard, G.: An lll-reduction algorithm with quasi-linear time complexity: extended abstract. In: Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing. p. 403–412. STOC '11, Association for Computing Machinery, New York, NY, USA (2011)
22. Ryan, K., Heninger, N.: Fast practical lattice reduction through iterated compression. In: Handschuh, H., Lysyanskaya, A. (eds.) Advances in Cryptology – CRYPTO 2023. pp. 3–36. Springer Nature Switzerland, Cham (2023)
23. Saruchi, Morel, I., Stehlé, D., Villard, G.: LLL reducing with the most significant bits. In: Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation. p. 367–374. ISSAC '14, Association for Computing Machinery, New York, NY, USA (2014)
24. Seysen, M.: Simultaneous reduction of a lattice basis and its reciprocal basis. Combinatorica **13**(3), 363–376 (Sep 1993)
25. development team, T.F.: fplll, a lattice reduction library, Version: 5.5.0 (2023)
26. Villard, G.: Parallel lattice basis reduction. In: Proceedings of the 1992 International Symposium on Symbolic and Algebraic Computation. pp. 269–277. ISSAC '92, Association for Computing Machinery, New York, NY, USA (1992)
27. Yap, C.K.: Fast unimodular reduction: Planar integer lattices. In: Proceedings., 33rd Annual Symposium on Foundations of Computer Science. pp. 437–446. IEEE Computer Society (1992)

Appendices

Appendices A–D give the numerical analysis, Appendix E reports the experiments, and Appendix F contains auxiliary proofs and the nearest-plane application.

A Linear algebra and perturbation bounds

A.1 Stable linear algebra

Condition-number conversion. For the perturbation and initialization estimates below, put

$$\operatorname{cond}_1(\mathbf{B}) := \big\| |\mathbf{B}|\,|\mathbf{B}^{-1}| \big\|_1.$$

This is the induced 1-norm version of the Skeel condition number used in QR and lattice-reduction analyses [2, 23]. For a nonsingular matrix of order n,

$$\frac{\kappa_{\mathrm{col}}(\mathbf{B})}{n} \le \operatorname{cond}_1(\mathbf{B}) \le n\,\kappa_{\mathrm{col}}(\mathbf{B}). \tag{A.1}$$

For the upper bound, column-scale invariance and norm inequalities give $\operatorname{cond}_1(\mathbf{B}) \le n\,\kappa_2(\mathbf{BD})$ for every positive diagonal $\mathbf{D}$. For the lower bound, choose $D_{jj} = \|\mathbf{B}_j\|_1^{-1}$, so $\|\mathbf{BD}\|_1\|(\mathbf{BD})^{-1}\|_1 = \operatorname{cond}_1(\mathbf{B})$, and apply norm inequalities again.

At p-bit precision put $u = 2^{-p}$. Lemma 6 converts the operation counts below into bit costs for the scaled operands used in reduction.

Theorem 7 (Stable arithmetic bounds). *There are dyadic thresholds $u_\times(n), u_{\mathrm{QR}}(n) \le 1$ and nondecreasing factors $\mu_\omega(n), \mu_{\mathrm{QR}}(n) \ge 1$ such that*

$$\log(1/u_\times(n)) = O(\log n), \quad \log(1/u_{\mathrm{QR}}(n)) = O(\log n),$$
$$\log \max\{\mu_\omega(n), \mu_{\mathrm{QR}}(n)\} = O(\log n).$$

The following three bounds hold.

1. *If $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$ and $u \le u_\times(n)$, fast multiplication returns*

$$\|\widetilde{\mathbf{AB}} - \mathbf{AB}\|_{\max} \le u\mu_\omega(n)\|\mathbf{A}\|_{\max}\|\mathbf{B}\|_{\max}$$

   in $O(n^\omega)$ arithmetic operations.
2. *If $n \ge 1$, $u \le u_{\mathrm{QR}}(n)$, and $\mathbf{A} \in \mathbb{R}^{n\times n}$ has no zero column, column-scaled QR costs $O(n^\omega)$ operations and returns an upper-triangular matrix $\widetilde{\mathbf{R}}$ whose nonzero diagonal entries are positive. There is a matrix $\widetilde{\mathbf{Q}}$ with $\widetilde{\mathbf{Q}}^T\widetilde{\mathbf{Q}} = \mathbf{I}$ such that, with $\widetilde{\mathbf{A}} = \widetilde{\mathbf{Q}}\widetilde{\mathbf{R}}$,*

$$\|(\widetilde{\mathbf{A}} - \mathbf{A})_i\|_1 \le u\mu_{\mathrm{QR}}(n)\|\mathbf{A}_i\|_1 \qquad (i \in [n]).$$

   If $\mathbf{A}$ is nonsingular and

$$u\mu_{\mathrm{QR}}(n)\|\mathbf{A}\|_1\|\mathbf{A}^{-1}\|_1 < 1,$$

   then $\widetilde{\mathbf{R}}$ has positive diagonal.

3. *Recursive inversion uses $O(n^\omega)$ operations and either reports failure or returns $\mathbf{Y}$. For a condition bound $K_2 \geq 1$, there is a nondecreasing factor $\mu_{\mathrm{inv}}(n, K_2) \geq 1$ with*

$$\log \mu_{\mathrm{inv}}(n, K_2) = O\big(\log^2(2n) + \log(2n)\log(2 + K_2)\big).$$

Whenever $\mathbf{X} \in \mathbb{R}^{n\times n}$ is nonsingular, $\kappa_2(\mathbf{X}) \leq K_2$, $0 < \varepsilon_{\mathrm{inv}} \leq 1/2$, and $u\mu_{\mathrm{inv}}(n, K_2) \leq \varepsilon_{\mathrm{inv}}$, recursive inversion returns $\mathbf{Y}$ satisfying

$$\max\{\|\mathbf{I} - \mathbf{YX}\|_2, \|\mathbf{I} - \mathbf{XY}\|_2\} \leq \varepsilon_{\mathrm{inv}}.$$

Proof. Multiplication. Apply [6, Theorem 3.3].

QR factorization. Fast QR of an $n \times n$ matrix costs $O(n^\omega)$ operations. For its computed triangular factor $\widetilde{\mathbf{R}}$, there is an orthogonal matrix $\widetilde{\mathbf{Q}}$ such that $\widetilde{\mathbf{Q}}\widetilde{\mathbf{R}} = \mathbf{A}+\boldsymbol{\Delta}$, where $\|\boldsymbol{\Delta}\|_2 \leq u\mu_0(n)\|\mathbf{A}\|_2$, with $\log \mu_0 = O(\log n)$ [5, Section 4.1, Lemma 4.1].

To bound the perturbation of each column separately as in [5, Section 4.3], put

$$D_{ii} = 2^{\lfloor \log_2 \|\mathbf{A}_i\|_1 \rfloor}, \qquad \mathbf{C} = \mathbf{AD}^{-1}.$$

Then $1 \leq \|\mathbf{C}_i\|_1 < 2$. Apply the cited QR algorithm to $\mathbf{C}$ and multiply its triangular factor by $\mathbf{D}$. Writing $\boldsymbol{\Delta}$ for the perturbation of $\mathbf{C}$, the perturbation of the original column i is $\boldsymbol{\Delta}_i D_{ii}$, with

$$\|\boldsymbol{\Delta}_i D_{ii}\|_1 \leq 2u\mu_0(n)n\,\|\mathbf{A}_i\|_1.$$

Include $2n$ in μ_{QR}.

For rank-deficient inputs, use the identity Householder transformation on a zero active vector and record a zero diagonal. The analysis factor $\widetilde{\mathbf{Q}}$ need not be stored. Round the admissible thresholds $u_\times, u_{\mathrm{QR}}$ down and the error factors $\mu_\omega, \mu_{\mathrm{QR}}$ up to dyadic values, taking nondecreasing upper bounds for the factors. Power-of-two scaling is exact and adds $O(n^2)$ work to the QR cost. Absorbing the exact sign normalization into $\widetilde{\mathbf{Q}}$ makes every nonzero diagonal positive.

For nonsingular $\mathbf{A}$, the columnwise bound gives $\|\widetilde{\mathbf{A}} - \mathbf{A}\|_1 \leq u\mu_{\mathrm{QR}}(n)\|\mathbf{A}\|_1$ and hence

$$\widetilde{\mathbf{A}} = \big(\mathbf{I}_n + (\widetilde{\mathbf{A}} - \mathbf{A})\mathbf{A}^{-1}\big)\mathbf{A}.$$

The displayed smallness condition implies that $\widetilde{\mathbf{A}}$ is nonsingular, so $\widetilde{\mathbf{R}}$ has positive diagonal.

Inversion. The recurrences in [5, Sections 3.1–3.2, equations (8)–(9)] imply

$$\frac{\|\mathbf{Y} - \mathbf{X}^{-1}\|_2}{\|\mathbf{X}^{-1}\|_2} \leq u\,\nu(n, K_2),$$
$$\nu(n, K_2) \leq n^{O(1)}(2 + K_2)^{O(\log(2n))} \max_{1\leq n'\leq n} \mu_\omega(n').$$

The reduction $\mathbf{X}^{-1} = \mathbf{X}^T(\mathbf{XX}^T)^{-1}$ from general to symmetric positive-definite inversion squares K_2 and only changes the absolute constants in these bounds.

The inversion induction also bounds higher-order terms when the right side is at most $1/2$. Since

$$\max\{\|\mathbf{I}-\mathbf{YX}\|_2, \|\mathbf{I}-\mathbf{XY}\|_2\} \le \|\mathbf{X}\|_2\,\|\mathbf{Y}-\mathbf{X}^{-1}\|_2,$$

take μ_{inv} to be a nondecreasing upper bound on $K_2\nu(n,K_2)$. □

A.2 QR perturbations

For matrix-norm estimates, we express similarity using an error matrix. For $\mathbf{C} \approx_\varepsilon \widetilde{\mathbf{C}}$, choose an orthogonal $\mathbf{O}$ realizing this similarity and set $\mathbf{F} = (\mathbf{O}\widetilde{\mathbf{C}} - \mathbf{C})\mathbf{C}^{-1}$. Then

$$\mathbf{O}\widetilde{\mathbf{C}} = (\mathbf{I}+\mathbf{F})\mathbf{C}, \qquad \|\mathbf{F}\|_2 \le \varepsilon.$$

The norm bound follows from the defining inequality by substituting $\mathbf{x} = \mathbf{C}^{-1}\mathbf{y}$. The next lemma converts columnwise error to this form.

Lemma 7 (Left perturbation bound). *Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be nonsingular, and let $\widetilde{\mathbf{A}}$ have the same dimensions. There is a matrix $\mathbf{F}$ such that*

$$\widetilde{\mathbf{A}} = (\mathbf{I}+\mathbf{F})\mathbf{A}, \qquad \|\mathbf{F}\|_S \le \big\|\,|\mathbf{A}|\,|\mathbf{A}^{-1}|\,\big\|_S \max_{1\le i\le n} \frac{\|(\widetilde{\mathbf{A}}-\mathbf{A})_i\|_1}{\|\mathbf{A}_i\|_1}.$$

Proof. Put $\boldsymbol{\Delta} = \widetilde{\mathbf{A}} - \mathbf{A}$ and $\mathbf{F} = \boldsymbol{\Delta}\mathbf{A}^{-1} \in \mathbb{R}^{n\times n}$. Since $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_n$, $(\mathbf{I}_n+\mathbf{F})\mathbf{A} = \widetilde{\mathbf{A}}$. If ρ is the largest relative column error in the statement, then

$$\|\mathbf{F}\|_S \le \sum_{k=1}^{n} \|\boldsymbol{\Delta}_k\|_1 \sum_j |(\mathbf{A}^{-1})_{kj}| \le \rho \sum_{k=1}^{n} \|\mathbf{A}_k\|_1 \sum_j |(\mathbf{A}^{-1})_{kj}|,$$

and the last sum is $\big\|\,|\mathbf{A}|\,|\mathbf{A}^{-1}|\,\big\|_S$. □

The corresponding QR relation is [1, Theorem 3.1, equation (3.2), pp. 304–305].

Theorem 8 (QR under a left perturbation). *Let $n \ge 1$, let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be nonsingular, and let $\mathbf{F} \in \mathbb{R}^{n\times n}$ satisfy $\|\mathbf{F}\|_2 < 1$. Then $(\mathbf{I}+\mathbf{F})\mathbf{A}$ is nonsingular. Let $\mathbf{A} = \mathbf{QR}$ and $(\mathbf{I}+\mathbf{F})\mathbf{A} = \widetilde{\mathbf{Q}}\widetilde{\mathbf{R}}$ be the positive-diagonal QR factorizations. The matrix $\widetilde{\mathbf{R}}\mathbf{R}^{-1}$ is positive-diagonal upper triangular and*

$$\|\widetilde{\mathbf{R}}\mathbf{R}^{-1} - \mathbf{I}\|_F \le \frac{\sqrt{2}\|\mathbf{F}\|_F}{1-\|\mathbf{F}\|_2}.$$

If $\varepsilon > 0$ and

$$\|\mathbf{F}\|_S \le \frac{\varepsilon}{(\sqrt{2}+1)\varepsilon + 2\sqrt{2}},$$

then the preceding norm condition holds, and every ratio of positive QR diagonals changes by a factor between $(1+\varepsilon)^{-1}$ and $1+\varepsilon$.

Proof. The norm condition makes $\mathbf{I} + \mathbf{F}$ nonsingular. The QR bound follows directly from the cited theorem.

For the last assertion, put $\rho_F = \|\mathbf{F}\|_S$. Lemma 1 bounds each diagonal change by $1 \pm \rho_F$. The stated hypothesis gives $\rho_F \le \varepsilon/(\varepsilon + 2)$, so each diagonal ratio changes by at most $(1 + \rho_F)/(1 - \rho_F) \le 1 + \varepsilon$ in either direction. □

Lemma 8 (Condition number under perturbation). *Let $\mathbf{R}$ be nonsingular and square, and put $\widetilde{\mathbf{R}} = (\mathbf{I}+\mathbf{F})\mathbf{R}$. Let ρ be either $\|\mathbf{F}\|_1$ or $\|\mathbf{F}\|_S$. If $\rho < 1$, then*

$$\mathrm{cond}_1(\widetilde{\mathbf{R}}) \le \frac{1+\rho}{1-\rho}\mathrm{cond}_1(\mathbf{R}).$$

Proof. Since $\|\mathbf{F}\|_1 \le \rho < 1$, $\widetilde{\mathbf{R}}^{-1} = \mathbf{R}^{-1}(\mathbf{I}+\mathbf{F})^{-1}$. The Neumann series and entrywise triangle inequality give $|\widetilde{\mathbf{R}}| \le (\mathbf{I}+|\mathbf{F}|)|\mathbf{R}|$ and $|\widetilde{\mathbf{R}}^{-1}| \le |\mathbf{R}^{-1}| \sum_{j\ge 0} |\mathbf{F}|^j$. Multiply as nonnegative matrices and apply the induced 1-norm:

$$\mathrm{cond}_1(\widetilde{\mathbf{R}}) \le (1+\rho)\mathrm{cond}_1(\mathbf{R}) \sum_{j\ge 0} \rho^j = \frac{1+\rho}{1-\rho}\mathrm{cond}_1(\mathbf{R}).$$

□

B Initialization

Initialization estimates the condition number from the truncated, column-scaled input and uses the estimate to choose the initial QR precision. The resulting factor supplies the initial scales for later operations.

B.1 Column scaling and input truncation

Lemma 9 (Input truncation error). *Let $\mathbf{B} \in \mathbb{Z}^{n\times n}$, $n \ge 1$, have no zero column and be represented as in Section 2.2. Let $e_j^{\max}$ be the largest exponent in column j, and put $\mathbf{D}^{\mathrm{col}} = \mathrm{diag}(2^{-e_j^{\max}})$ and $\mathbf{A}_{\mathrm{col}} = \mathbf{B}\mathbf{D}^{\mathrm{col}}$. Then*

$$1 \le \|(\mathbf{A}_{\mathrm{col}})_j\|_1 < 2n \qquad (j \in [n]). \tag{B.1}$$

At precision $p \ge 1$, truncating each entry to $p + O(1)$ significant bits gives a working matrix $\widetilde{\mathbf{A}}_p$ satisfying

$$\|(\widetilde{\mathbf{A}}_p - \mathbf{A}_{\mathrm{col}})_j\|_1 \le c2^{-p}\|(\mathbf{A}_{\mathrm{col}})_j\|_1 \qquad (j \in [n]). \tag{B.2}$$

Over precision doubling to p_{final}, constructing these matrices costs $\widetilde{O}(n^2 p_{\mathrm{final}})$.

Proof. Finding each column's largest exponent costs $\widetilde{O}(n^2)$ without reading the significands. Subtracting these maxima keeps each exponent in $O(1)$ machine words. Reading and retaining each entry's first $p+O(1)$ significand bits costs $\widetilde{O}(p)$ per entry and gives relative error at most $c2^{-p}$. Summing within a column proves (B.2). Geometric doubling bounds the total cost by that of the last attempt. □

B.2 Condition estimation

We estimate the condition number of a triangular QR factor, whose computed inverse can be checked by multiplication. For the positive-diagonal QR factorization $\mathbf{A} = \mathbf{QR}$, write

$$\mathrm{cond}_{R,1}(\mathbf{A}) := \mathrm{cond}_1(\mathbf{R}).$$

Orthogonality and norm equivalence give

$$n^{-1}\mathrm{cond}_{R,1}(\mathbf{A}) \le \mathrm{cond}_1(\mathbf{A}) \le n\mathrm{cond}_{R,1}(\mathbf{A}).$$

Algorithm 3 computes the QR factor of a column-scaled input and doubles the precision until the inverse and QR error tests below pass. At precision p, put $u = 2^{-p}$, let $\widetilde{\mathbf{R}}_p$ be the computed factor, and let $\widetilde{\mathbf{Y}}_p$ approximate its inverse. We denote directed upward rounding by roundUp and use $p+1$ significand bits, so each directed rounding has relative error at most u. We compute an inverse-residual bound $\bar{\rho}_{\mathrm{inv},p}$ and a condition estimate $\bar{\kappa}_p$ by

$$\bar{\rho}_{\mathrm{inv},p} = n\ \mathrm{roundUp}\Big((1-u)^{-1}\|\,\mathrm{fl}(\widetilde{\mathbf{R}}_p\widetilde{\mathbf{Y}}_p - \mathbf{I})\|_{\max} + u\mu_\omega(n)\|\widetilde{\mathbf{R}}_p\|_{\max}\|\widetilde{\mathbf{Y}}_p\|_{\max}\Big),$$

$$\bar{\kappa}_p = \mathrm{roundUp}\left(\max_j \sum_\ell |(\widetilde{Y}_p)_{\ell j}| \sum_i |(\widetilde{R}_p)_{i\ell}|\right). \tag{B.3}$$

The acceptance tests are

$$\begin{aligned} &u \le \min\{u_\times(n), u_{\mathrm{QR}}(n)\}, \qquad \bar{\rho}_{\mathrm{inv},p} < c, \\ &\mathrm{roundUp}\big(2^{-p}n^4(1+\mu_{\mathrm{QR}}(n))\bar{\kappa}_p\big) < \frac{c}{\sqrt{n}}. \end{aligned} \tag{B.4}$$

Algorithm 3 Condition estimation by precision doubling

```
procedure EstimateCondition(B)
    Form D^col from the column maxima as in Lemma 9
    p ← 2 max{⌈log2(n^2)⌉, 1}
    while true do
        u ← 2^{-p}
        if u ≤ min{u_×(n), u_QR(n)} then
            Truncate B D^col to Ã_p as in Lemma 9
            R̃_p ← computeR_p(Ã_p)
            Attempt recursive inversion of R̃_p at precision p
            if inversion succeeds with result Ỹ_p then
                Compute ρ̄_{inv,p}, κ̄_p by (B.3)
                if (B.4) holds then
                    return κ̄_p
        p ← 2p
```

Theorem 9 (Factor-two condition estimation). *Let $\mathbf{B} \in \mathbb{Z}^{n\times n}$, $n \geq 1$, be a nonsingular input represented as in Section 2.2. Algorithm 3 terminates with an estimate $\bar{\kappa}$ of $\mathrm{cond}_{R,1}(\mathbf{B})$ satisfying $\bar{\kappa}/\mathrm{cond}_{R,1}(\mathbf{B}) \in [1/2, 2]$. Including the QR factorizations, its total cost is $\widetilde{O}(n^\omega(1 + \log \mathrm{cond}_{R,1}(\mathbf{B})))$ bit operations under these conventions.*

Proof. Scaling and residual bounds. The column normalization $\mathbf{A}_{\mathrm{col}} = \mathbf{B}\mathbf{D}^{\mathrm{col}}$ satisfies

$$\max\left\{\kappa_2(\mathbf{A}_{\mathrm{col}}), \|\mathbf{A}_{\mathrm{col}}\|_1\|\mathbf{A}_{\mathrm{col}}^{-1}\|_1\right\} \leq 2n^3\,\mathrm{cond}_{R,1}(\mathbf{B}). \tag{B.5}$$

Since every column of $\mathbf{A}_{\mathrm{col}}$ has 1-norm in $[1, 2n)$ by (B.1),

$$\|\mathbf{A}_{\mathrm{col}}\|_1 < 2n, \qquad \|\mathbf{A}_{\mathrm{col}}^{-1}\|_1 \leq \mathrm{cond}_1(\mathbf{A}_{\mathrm{col}}).$$

Using $\kappa_2(\mathbf{A}_{\mathrm{col}}) \leq n\|\mathbf{A}_{\mathrm{col}}\|_1\|\mathbf{A}_{\mathrm{col}}^{-1}\|_1$, $\mathrm{cond}_1(\mathbf{A}_{\mathrm{col}}) \leq n\,\mathrm{cond}_{R,1}(\mathbf{A}_{\mathrm{col}})$ and column-scale invariance proves (B.5). At precision p, put $u = 2^{-p}$ and $\widetilde{\mathbf{R}}_p := \mathrm{computeR}_p(\widetilde{\mathbf{A}}_p)$. The computed factor is exact for $\mathbf{A}_p^{\mathrm{qr}} = \mathbf{Q}_p\widetilde{\mathbf{R}}_p$, where $\mathbf{Q}_p$ is orthogonal. At precisions satisfying the arithmetic thresholds and $u(1 + \mu_{\mathrm{QR}}(n)) \leq 1/4$, the columnwise truncation and QR bounds give

$$\max_j \frac{\|(\mathbf{A}_{\mathrm{col}} - \mathbf{A}_p^{\mathrm{qr}})_j\|_1}{\|(\mathbf{A}_p^{\mathrm{qr}})_j\|_1} \leq u(6 + 2\mu_{\mathrm{QR}}(n)). \tag{B.6}$$

The extra smallness condition holds at acceptance because $\bar{\kappa}_p \geq \|\widetilde{\mathbf{R}}_p\widetilde{\mathbf{Y}}_p\|_1 \geq 1 - c$. It also holds at the sufficient precision below.

After a successful inversion, put $\boldsymbol{\Delta}_{\mathrm{inv},p} = \widetilde{\mathbf{R}}_p\widetilde{\mathbf{Y}}_p - \mathbf{I}$ and $\kappa_p^{\mathrm{proxy}} = \||\widetilde{\mathbf{R}}_p|\,|\widetilde{\mathbf{Y}}_p|\|_1$. The directed evaluations in (B.3) and $\|\cdot\|_1 \leq n\|\cdot\|_{\max}$ guarantee

$$\|\boldsymbol{\Delta}_{\mathrm{inv},p}\|_1 \leq \bar{\rho}_{\mathrm{inv},p}, \qquad \kappa_p^{\mathrm{proxy}} \leq \bar{\kappa}_p \leq \frac{\kappa_p^{\mathrm{proxy}}}{1 - 2nu} \quad (2nu < 1). \tag{B.7}$$

Accepted estimate. At acceptance, $\|\boldsymbol{\Delta}_{\mathrm{inv},p}\|_1 < c$. Direct and Neumann bounds for $\widetilde{\mathbf{R}}_p\widetilde{\mathbf{Y}}_p = \mathbf{I} + \boldsymbol{\Delta}_{\mathrm{inv},p}$ show that $\kappa_p^{\mathrm{proxy}}/\mathrm{cond}_1(\widetilde{\mathbf{R}}_p)$ lies in $[1 - \|\boldsymbol{\Delta}_{\mathrm{inv},p}\|_1, 1 + \|\boldsymbol{\Delta}_{\mathrm{inv},p}\|_1]$. Equation (B.7) and the acceptance tests (B.4) also give $\bar{\kappa}_p/\kappa_p^{\mathrm{proxy}} = 1 + O(c)$.

The accepted residual makes $\widetilde{\mathbf{R}}_p\widetilde{\mathbf{Y}}_p$ nonsingular, so $\mathbf{A}_p^{\mathrm{qr}}$ is nonsingular. Thus $\mathbf{F}_{\mathrm{col}} = (\mathbf{A}_{\mathrm{col}} - \mathbf{A}_p^{\mathrm{qr}})(\mathbf{A}_p^{\mathrm{qr}})^{-1}$ satisfies $\mathbf{A}_{\mathrm{col}} = (\mathbf{I} + \mathbf{F}_{\mathrm{col}})\mathbf{A}_p^{\mathrm{qr}}$. Lemma 7, (B.6), and $\||\mathbf{A}_p^{\mathrm{qr}}||(\mathbf{A}_p^{\mathrm{qr}})^{-1}|\|_S \leq n^4\mathrm{cond}_1(\widetilde{\mathbf{R}}_p)$ give

$$\|\mathbf{F}_{\mathrm{col}}\|_S = O\big(un^4(1 + \mu_{\mathrm{QR}}(n))\bar{\kappa}_p\big) = O(c/\sqrt{n}). \tag{B.8}$$

Theorem 8 and Lemma 8, applied in both directions, compare the two QR condition numbers. Let $\mathbf{R}_{\mathrm{col}}$ be the positive-diagonal R-factor of $\mathbf{A}_{\mathrm{col}}$. Then $\mathrm{cond}_1(\widetilde{\mathbf{R}}_p)/\mathrm{cond}_1(\mathbf{R}_{\mathrm{col}}) = 1 + O(c)$. Column-scale invariance gives $\mathrm{cond}_1(\mathbf{R}_{\mathrm{col}}) = \mathrm{cond}_{R,1}(\mathbf{B})$. Combining this QR comparison with the proxy and directed-rounding bounds puts $\bar{\kappa}_p/\mathrm{cond}_{R,1}(\mathbf{B})$ in $[1/2, 2]$.

Precision and cost. Theorem 7 and the residual and test bounds above give a sufficient precision satisfying

$$P_{\mathrm{est}}(n,K) = O\big(\log n + \log^2(2n) + \log(2n)\log(2+K)\big)\,,$$
$$p \geq P_{\mathrm{est}}\big(n, 2n^3\,\mathrm{cond}_{R,1}(\mathbf{B})\big)\,.$$

At this precision, inversion and the tests succeed. Each attempt reads significand prefixes and costs $\widetilde{O}(n^\omega p)$. Doubling reaches this precision within a factor of two, and the final attempt dominates the total cost up to logarithmic factors. □

B.3 Initial QR factor

Fix $0 < \varepsilon_{\mathrm{init}} \leq c$. We obtain the initial factor at this accuracy by strengthening the estimator's acceptance test. Order $\mathbf{B}^{\mathrm{sort}} = \mathbf{BP}$ by the norm estimates s_j. Run Algorithm 3 on $\mathbf{B}^{\mathrm{sort}}$, retaining its column scale $\mathbf{D}^{\mathrm{col}}$. At precision p, set

$$\bar{\rho}_p^{\mathrm{init}} = \mathrm{roundUp}\big(c^{-1}un^4(1+\mu_{\mathrm{QR}}(n))\bar{\kappa}_p\big)\,.$$

Add $\bar{\rho}_p^{\mathrm{init}} \leq c\varepsilon_{\mathrm{init}}$ to the acceptance test in Algorithm 3, line 12, and stop at the first $p = p_{\mathrm{init}}$ satisfying both tests. Return

$$\mathbf{R}^{\mathrm{init}} = \widetilde{\mathbf{R}}_{p_{\mathrm{init}}}(\mathbf{D}^{\mathrm{col}})^{-1}, \qquad \mathbf{E}_{\mathrm{init}} = \mathrm{scale}(\mathbf{R}^{\mathrm{init}}), \qquad K_{\mathrm{init}} = c^{-1}\bar{\kappa}_p.$$

Lemma 10 (Initialization guarantees). *Let $\mathbf{B} \in \mathbb{Z}^{n\times n}$, $n \geq 1$, be a nonsingular input represented as in Section 2.2, and let $0 < \varepsilon_{\mathrm{init}} \leq c$. The initialization procedure terminates with a triangular factor $\mathbf{R}^{\mathrm{init}}$ that is the positive-diagonal R-factor of a basis $\mathbf{B}_{\mathrm{init}}$ satisfying*

$$\mathbf{B}^{\mathrm{sort}} \approx_{\varepsilon_{\mathrm{init}}} \mathbf{B}_{\mathrm{init}}.$$

The scaled factor satisfies

$$\max\{\|\mathbf{R}^{\mathrm{init}}(\mathbf{E}_{\mathrm{init}})^{-1}\|_1, \|\mathbf{E}_{\mathrm{init}}(\mathbf{R}^{\mathrm{init}})^{-1}\|_1\} \leq K_{\mathrm{init}}.$$

The initialization also gives

$$K_{\mathrm{init}} = O(n\,\mathrm{cond}_1(\mathbf{B}))\,, \qquad \mathrm{fspread}(\mathbf{R}^{\mathrm{init}}) = O(\kappa_{\mathrm{col}}(\mathbf{B}))\,.$$

The working precision and initialization cost are

$$p_{\mathrm{init}} = \widetilde{O}(1 + \log n + \log \mathrm{cond}_1(\mathbf{B}) + \log(1/\varepsilon_{\mathrm{init}}))$$

and

$$\widetilde{O}(n^\omega(1 + \log \mathrm{cond}_1(\mathbf{B}) + \log(1/\varepsilon_{\mathrm{init}})))\,.$$

Proof. Accuracy. For the accepted attempt $p = p_{\mathrm{init}}$, reuse $\mathbf{A}_p^{\mathrm{qr}}$ and $\mathbf{F}_{\mathrm{col}}$ from the estimator proof. Equation (B.8) and the additional test give

$$\|\mathbf{F}_{\mathrm{col}}\|_S \le \bar{\rho}_p^{\mathrm{init}} \le c\varepsilon_{\mathrm{init}}.$$

Set $\mathbf{B}_{\mathrm{init}} = \mathbf{A}_p^{\mathrm{qr}}(\mathbf{D}^{\mathrm{col}})^{-1}$. Restoring column scales gives $\mathbf{B}^{\mathrm{sort}} = (\mathbf{I} + \mathbf{F}_{\mathrm{col}})\mathbf{B}_{\mathrm{init}}$, with the stated R-factor $\mathbf{R}^{\mathrm{init}}$. The reverse bound in Lemma 1 proves $\mathbf{B}^{\mathrm{sort}} \approx_{\varepsilon_{\mathrm{init}}} \mathbf{B}_{\mathrm{init}}$, since $c\varepsilon_{\mathrm{init}}/(1 - c\varepsilon_{\mathrm{init}}) \le \varepsilon_{\mathrm{init}}$.

Norm bounds and forward spread. Scale invariance and the accepted residual give $\mathrm{cond}_1(\mathbf{R}^{\mathrm{init}}) = \mathrm{cond}_1(\widetilde{\mathbf{R}}_p) \le (1 + O(c))\bar{\kappa}_p$. For $\mathbf{E}_{\mathrm{init}} = \mathrm{scale}(\mathbf{R}^{\mathrm{init}})$, the two scaled-factor 1-norms are therefore at most $4\bar{\kappa}_p \le K_{\mathrm{init}}$. Apply Theorem 9 to $\mathbf{B}^{\mathrm{sort}}$. The theorem's factor-two estimate and

$$\mathrm{cond}_{R,1}(\mathbf{B}^{\mathrm{sort}}) \le n\,\mathrm{cond}_1(\mathbf{B}^{\mathrm{sort}}) = n\,\mathrm{cond}_1(\mathbf{B}),$$

yield $K_{\mathrm{init}} = O(n\,\mathrm{cond}_1(\mathbf{B}))$.

Theorem 8, the choice of c, and Theorem 3 give $\mathrm{fspread}(\mathbf{R}^{\mathrm{init}}) \le 4\kappa_{\mathrm{col}}(\mathbf{B})$.

Precision and cost. The tests (B.4) pass at $p = \widetilde{O}(1 + \log n + \log \mathrm{cond}_1(\mathbf{B}))$. At this precision, directed rounding uses the accepted condition estimate to bound $\bar{\rho}_p^{\mathrm{init}}$ by $2^{-p}n^{O(1)}\mathrm{cond}_1(\mathbf{B})$, so the last test adds $O(\log(1/\varepsilon_{\mathrm{init}}))$ bits. All doubled attempts together have the same $\widetilde{O}$ bound as the final attempt. The initial norm sorting costs $\widetilde{O}(n^2)$, and Lemma 9 adds $\widetilde{O}(n^2 p_{\mathrm{init}})$ work. Since $\omega \ge 2$, this is $\widetilde{O}(n^\omega p_{\mathrm{init}})$. □

For $n \ge 2$, the choice (3.25) in Lemma 10, together with (A.1), gives (3.24)–(3.26). Taking $\mathbf{R}^{\mathrm{init}}$ as both root input factors gives zero local and reference error. This supplies the initial state for Theorem 11.

C Numerical updates

We prove that the scaled representation supports the required operations at the claimed precision. Bounds on return matrices control operand preparation and grid rounding. Local error bounds cover Apply, GridSeysen, and ReduceTwo.

A stored pair $(\widetilde{\mathbf{R}}_{\mathrm{sc}}, \mathbf{E})$ represents $\widetilde{\mathbf{R}} = \widetilde{\mathbf{R}}_{\mathrm{sc}}\mathbf{E}$. Each local routine takes its supplied stored input as exact. The GridSeysen and Apply error bounds give the increments in the composition law (2.1). The surrounding Apply incorporates each ReduceTwo return. In each result, K_{norm} bounds the indicated matrices and inverses in the induced 1-norm. Direct errors satisfy (3.10).

C.1 Return norms and grid decoding

Return matrices inherit norm bounds from the scaled factors they connect.

Lemma 11 (Norm bounds for return matrices). *Let $\mathbf{R}_{\mathrm{in}}, \mathbf{R}_{\mathrm{out}}$ be nonsingular upper triangular matrices of order n and $\mathbf{E}_{\mathrm{in}}, \mathbf{E}_{\mathrm{out}}$ power-of-two scales. Suppose*

$$\mathbf{R}_{\mathrm{in}}\mathbf{U} = \mathbf{Q}\mathbf{R}_{\mathrm{out}}, \qquad \mathbf{U} \in \mathrm{GL}_n(\mathbb{Z}), \qquad \mathbf{Q}^T\mathbf{Q} = \mathbf{I}.$$

Then the return matrix $\mathbf{M} = \mathbf{E}_{\mathrm{in}}\mathbf{U}\mathbf{E}_{\mathrm{out}}^{-1}$ satisfies

$$\|\mathbf{M}\|_1 \le \sqrt{n}\,\|\mathbf{E}_{\mathrm{in}}\mathbf{R}_{\mathrm{in}}^{-1}\|_1\|\mathbf{R}_{\mathrm{out}}\mathbf{E}_{\mathrm{out}}^{-1}\|_1,$$
$$\|\mathbf{M}^{-1}\|_1 \le \sqrt{n}\,\|\mathbf{E}_{\mathrm{out}}\mathbf{R}_{\mathrm{out}}^{-1}\|_1\|\mathbf{R}_{\mathrm{in}}\mathbf{E}_{\mathrm{in}}^{-1}\|_1.$$

Proof. The QR relation gives

$$\mathbf{M} = (\mathbf{R}_{\mathrm{in}}\mathbf{E}_{\mathrm{in}}^{-1})^{-1}\mathbf{Q}(\mathbf{R}_{\mathrm{out}}\mathbf{E}_{\mathrm{out}}^{-1}).$$

Use $\|\mathbf{Q}\|_1 \le \sqrt{n}$ and invert this identity. □

After floating-point multiplication, grid decoding restores each return entry to its grid. For a matrix $\mathbf{X}$ and compatible power-of-two scales $\mathbf{E}$ and $\mathbf{F}$, define the entrywise grid projection by

$$\big(\mathrm{GridDecode}(\mathbf{X}, \mathbf{E}, \mathbf{F})\big)_{ij} = \frac{E_{ii}}{F_{jj}}\left\lfloor \frac{F_{jj}}{E_{ii}} X_{ij} \right\rceil.$$

The represented update $\mathbf{E}^{-1}\,\mathrm{GridDecode}(\mathbf{X}, \mathbf{E}, \mathbf{F})\mathbf{F}$ is therefore integral.

Lemma 12 (Correctness of grid decoding). *Let $\mathbf{E}_0, \mathbf{E}_1$ be power-of-two scales of order n, and let $\mathbf{M}_{\mathrm{pre}}, \mathbf{M}_{\mathrm{dec}}$ be matrices. Suppose $\mathbf{U}_{\mathrm{pre}} := \mathbf{E}_0^{-1}\mathbf{M}_{\mathrm{pre}}\mathbf{E}_1 \in \mathrm{GL}_n(\mathbb{Z})$ and $\mathbf{U}_{\mathrm{dec}} := \mathbf{E}_0^{-1}\mathbf{M}_{\mathrm{dec}}\mathbf{E}_1 \in \mathbb{Z}^{n\times n}$. For a positive-diagonal triangular $\mathbf{R}$, let $\mathbf{R}_{\mathrm{pre}}$ be the positive-diagonal R-factor of $\mathbf{R}\mathbf{U}_{\mathrm{pre}}$. Set*

$$\boldsymbol{\Delta} := \mathbf{M}_{\mathrm{pre}}^{-1}(\mathbf{M}_{\mathrm{dec}} - \mathbf{M}_{\mathrm{pre}}), \qquad \mathbf{U}_{\mathrm{corr}} := \mathbf{U}_{\mathrm{pre}}^{-1}\mathbf{U}_{\mathrm{dec}}.$$

If $\mathbf{E}_1$ is admissible for $\mathbf{R}_{\mathrm{pre}}$, $\mathrm{lfspread}(\mathbf{R}_{\mathrm{pre}}) \le \bar{\sigma}$ for some $\bar{\sigma} \ge 0$, and

$$\|\boldsymbol{\Delta}\|_1 \le c2^{-\bar{\sigma}},$$

then $\mathbf{U}_{\mathrm{corr}}$ is upper unitriangular over $\mathbb{Z}$.

Whenever $\mathbf{U}_{\mathrm{corr}}$ is upper unitriangular, $\mathbf{U}_{\mathrm{dec}} \in \mathrm{GL}_n(\mathbb{Z})$. If $\mathbf{R}_{\mathrm{dec}}$ is the positive-diagonal R-factor of $\mathbf{R}\mathbf{U}_{\mathrm{dec}}$, then

$$\mathbf{R}_{\mathrm{dec}} = \mathbf{R}_{\mathrm{pre}}\mathbf{U}_{\mathrm{corr}}.$$

Thus $\mathbf{R}_{\mathrm{dec}}$ and $\mathbf{R}_{\mathrm{pre}}$ have the same exact diagonal, logarithmic forward spread, and potential. If $\|\boldsymbol{\Delta}\|_1 \le c$ as well, then

$$\|\mathbf{M}_{\mathrm{dec}}\|_1 \le 2\|\mathbf{M}_{\mathrm{pre}}\|_1, \qquad \|\mathbf{M}_{\mathrm{dec}}^{-1}\|_1 \le 2\|\mathbf{M}_{\mathrm{pre}}^{-1}\|_1.$$

Proof. Since $\mathbf{U}_{\mathrm{pre}}^{-1}$ and $\mathbf{U}_{\mathrm{dec}}$ are integral, so is $\mathbf{U}_{\mathrm{corr}}$. Scale cancellation gives

$$\boldsymbol{\Delta} = \mathbf{E}_1(\mathbf{U}_{\mathrm{corr}} - \mathbf{I})\mathbf{E}_1^{-1}.$$

For $i \geq j$, admissibility and $\mathrm{lfspread}(\mathbf{R}_{\mathrm{pre}}) \leq \bar{\sigma}$ give

$$|(\mathbf{U}_{\mathrm{corr}} - \mathbf{I})_{ij}| = \frac{(E_1)_{jj}}{(E_1)_{ii}}|\Delta_{ij}| \leq 8\,2^{\bar{\sigma}}\|\boldsymbol{\Delta}\|_1 \leq 8c < 1.$$

These integer entries vanish, so $\mathbf{U}_{\mathrm{corr}}$ is upper unitriangular.

If $\mathbf{U}_{\mathrm{corr}}$ is upper unitriangular, then $\mathbf{U}_{\mathrm{dec}} = \mathbf{U}_{\mathrm{pre}}\mathbf{U}_{\mathrm{corr}}$ is unimodular. Writing $\mathbf{R}\mathbf{U}_{\mathrm{pre}} = \mathbf{Q}\mathbf{R}_{\mathrm{pre}}$ gives $\mathbf{R}\mathbf{U}_{\mathrm{dec}} = \mathbf{Q}(\mathbf{R}_{\mathrm{pre}}\mathbf{U}_{\mathrm{corr}})$. The right factor is upper triangular with the same positive diagonal as $\mathbf{R}_{\mathrm{pre}}$, so it is exactly $\mathbf{R}_{\mathrm{dec}}$. If $\|\boldsymbol{\Delta}\|_1 \leq c$, then $\mathbf{M}_{\mathrm{dec}} = \mathbf{M}_{\mathrm{pre}}(\mathbf{I} + \boldsymbol{\Delta})$. Direct and Neumann estimates give the norm bounds. □

For the call-return composition (3.12), put $\mathbf{V} = \mathbf{E}_0^{-1}(\mathbf{M}_1 \cdots \mathbf{M}_s)\mathbf{E}_s$ and $\mathbf{U} = \mathbf{E}_0^{-1}\mathbf{T}_1\mathbf{E}_s$. If the decoding hypotheses hold at every step, Lemma 12 expresses each correction as multiplication on the right at the common final scale. Induction from $\mathbf{T}_s = \mathbf{M}_s$ gives $\mathbf{U} = \mathbf{V}\mathbf{W}$ with $\mathbf{W}$ integral upper unitriangular. For every fixed input, the exact returned factor is therefore the last direct-operation factor times $\mathbf{W}$. This proves the relation (3.13).

To apply this result to a computed product, we need an error tolerance that ensures the decoding hypotheses.

Lemma 13 (Grid decoding error). *Let $n \geq 1$, let $\mathbf{E}_0, \mathbf{E}_1$ be power-of-two scales of order n, and let $\mathbf{M}_{\mathrm{pre}}$ and $\widetilde{\mathbf{M}}_{\mathrm{agg}}$ be matrices of order n. Put $\mathbf{U}_{\mathrm{pre}} := \mathbf{E}_0^{-1}\mathbf{M}_{\mathrm{pre}}\mathbf{E}_1$. Assume $\mathbf{U}_{\mathrm{pre}} \in \mathrm{GL}_n(\mathbb{Z})$ and*

$$\max\{\|\mathbf{M}_{\mathrm{pre}}\|_1, \|\mathbf{M}_{\mathrm{pre}}^{-1}\|_1\} \leq K_{\mathrm{norm}}.$$

Put

$$\rho_{\mathrm{dec}} = \|\widetilde{\mathbf{M}}_{\mathrm{agg}} - \mathbf{M}_{\mathrm{pre}}\|_1.$$

Define

$$\mathbf{M}_{\mathrm{dec}} := \mathrm{GridDecode}(\widetilde{\mathbf{M}}_{\mathrm{agg}}, \mathbf{E}_0, \mathbf{E}_1). \tag{C.1}$$

Then

$$\|\mathbf{M}_{\mathrm{dec}} - \mathbf{M}_{\mathrm{pre}}\|_1 \leq 2\rho_{\mathrm{dec}},$$
$$\max\{\|(\mathbf{M}_{\mathrm{dec}} - \mathbf{M}_{\mathrm{pre}})\mathbf{M}_{\mathrm{pre}}^{-1}\|_1, \|\mathbf{M}_{\mathrm{pre}}^{-1}(\mathbf{M}_{\mathrm{dec}} - \mathbf{M}_{\mathrm{pre}})\|_1\} \leq 2\rho_{\mathrm{dec}}K_{\mathrm{norm}}.$$

For a positive-diagonal triangular $\mathbf{R}$, let $\mathbf{R}_{\mathrm{pre}}$ be the positive-diagonal R-factor of $\mathbf{R}\mathbf{E}_0^{-1}\mathbf{M}_{\mathrm{pre}}\mathbf{E}_1$. If $\mathrm{lfspread}(\mathbf{R}_{\mathrm{pre}}) \leq \bar{\sigma}$, $\mathbf{E}_1$ is admissible for $\mathbf{R}_{\mathrm{pre}}$, and $2\rho_{\mathrm{dec}}K_{\mathrm{norm}} \leq c2^{-\bar{\sigma}}$, then $\mathbf{U}_{\mathrm{dec}} := \mathbf{E}_0^{-1}\mathbf{M}_{\mathrm{dec}}\mathbf{E}_1 \in \mathrm{GL}_n(\mathbb{Z})$, and $\mathbf{U}_{\mathrm{pre}}^{-1}\mathbf{U}_{\mathrm{dec}}$ is upper unitriangular. Lemma 12 gives the same exact QR diagonal and potential for $\mathbf{R}\mathbf{U}_{\mathrm{pre}}$ and $\mathbf{R}\mathbf{U}_{\mathrm{dec}}$. Both $\|\mathbf{M}_{\mathrm{dec}}\|_1$ and $\|\mathbf{M}_{\mathrm{dec}}^{-1}\|_1$ are at most $2K_{\mathrm{norm}}$.

Proof. Every entry of $\mathbf{M}_{\mathrm{pre}}$ lies on its grid, so nearest-grid rounding gives

$$\|\mathbf{M}_{\mathrm{dec}} - \mathbf{M}_{\mathrm{pre}}\|_1 \le 2\rho_{\mathrm{dec}}.$$

Submultiplicativity gives both relative-error bounds. Grid membership makes the decoded update integral, so Lemma 12 gives the remaining conclusions under the stated smallness hypothesis. □

Encoding length. Before decoding, we round each newly computed return entry to p significant bits. Equation (C.1) shifts and rounds that significand, leaving at most $p+1$ bits. Copied blocks store references to shared entries. The order-one return is a power of two, and ReduceTwo uses $O(p)$-bit significands. Induction over the return construction gives every nonzero return entry a representation

$$x = z2^e, \qquad z, e \in \mathbb{Z}, \qquad \log_2(1+|z|) = O(p_{\max}). \tag{C.2}$$

Here $p_{\max}$ is the largest precision used before that return. The norm bounds and precision rule (D.9) give the return-norm hypothesis of Lemma 6.

For the root output $\mathbf{U} = \mathbf{P}\mathbf{E}_{\mathrm{init}}^{-1}\mathbf{M}_{\mathrm{loc}}\mathbf{E}_{\mathrm{out}}$, the permutation and power-of-two scales only reorder entries and change their exponents. Thus writing $\mathbf{U}$ in dyadic representation costs $\widetilde{O}(n^2 p_{\max})$, including exponent processing.

C.2 Operand preparation and Apply

For recursive updates, return composition, and basis output, we prepare each operand $\mathbf{X}$ by discarding entries below $2^{-p-O(1)}$ in magnitude and rounding the rest to p significant bits. For these operations, fl_p includes this preparation. The prepared matrix $\widetilde{\mathbf{X}}$ satisfies $|(\widetilde{\mathbf{X}}-\mathbf{X})_{ij}| \le u(|X_{ij}|+1)$. If $\max\{\|\mathbf{X}\|_1, \|\mathbf{X}^{-1}\|_1\} \le K_{\mathrm{norm}}$, then

$$\|(\widetilde{\mathbf{X}} - \mathbf{X})\mathbf{X}^{-1}\|_1 \le 2nuK_{\mathrm{norm}}^2. \tag{C.3}$$

Indeed, $\|\widetilde{\mathbf{X}}-\mathbf{X}\|_1 \le u(K_{\mathrm{norm}}+n)$. Multiplication by $\mathbf{X}^{-1}$ gives (C.3). For return composition, the absolute operand errors enter ρ_{dec}. Preparing an operand costs $\widetilde{O}(n^2 p)$, which is bounded by the dense-operation cost. The next lemma includes both relative operand errors in the error of Apply.

Lemma 14 (Similarity error after Apply). *Let $\mathbf{R}_{\mathrm{sc}}, \mathbf{M}$ and their product be nonsingular matrices of order n, each with 1-norm and inverse 1-norm at most $K \ge 1$. For input and output scales $\mathbf{E}, \mathbf{E}'$, put $\mathbf{R} = \mathbf{R}_{\mathrm{sc}}\mathbf{E}$ and $\mathbf{U} = \mathbf{E}^{-1}\mathbf{M}\mathbf{E}'$. Prepare both operands at precision p and compute*

$$\widetilde{\mathbf{Y}} = \mathrm{fl}_p(\widetilde{\mathbf{R}}_{\mathrm{sc}}\widetilde{\mathbf{M}}), \qquad \widetilde{\mathbf{R}}^{+} = \mathrm{computeR}_p(\widetilde{\mathbf{Y}}).$$

Define

$$\mu_{\mathrm{app}}(n,K) = 32n^2K^4(1+\mu_\omega(n)+\mu_{\mathrm{QR}}(n)), \qquad \eta_{\mathrm{app}} = 2^{-p}\mu_{\mathrm{app}}(n,K).$$

If $u = 2^{-p} \le \min\{u_\times(n), u_{\mathrm{QR}}(n)\}$ and $\eta_{\mathrm{app}} \le c$, then $\mathbf{R}\mathbf{U} \approx_{\eta_{\mathrm{app}}} \widetilde{\mathbf{R}}^{+}\mathbf{E}'$. The same factor μ_{app} bounds the absolute multiplication error, including operand preparation, by $u\mu_{\mathrm{app}}$.

Proof. Equation (C.3) bounds the sum of relative operand errors by $4nuK^2$. Stable multiplication then gives

$$\widetilde{\mathbf{Y}} = (\mathbf{I} + \mathbf{F}_\times)\mathbf{R}_{\mathrm{sc}}\mathbf{M}, \qquad \|\mathbf{F}_\times\|_1 \le 8nuK^4 + 4n\mu_\omega(n)uK^3 \le \tfrac{1}{4}.$$

Thus $\|\widetilde{\mathbf{Y}}\|_1 \le 5K/4$ and $\|\widetilde{\mathbf{Y}}^{-1}\|_1 \le 4K/3$. The QR bound in Theorem 7 and Lemma 7 give

$$\widetilde{\mathbf{Q}}\widetilde{\mathbf{R}}^+ = (\mathbf{I} + \mathbf{F}_{\mathrm{QR}})\widetilde{\mathbf{Y}}, \qquad \|\mathbf{F}_{\mathrm{QR}}\|_1 \le 2nK^2\mu_{\mathrm{QR}}(n)u.$$

The smallness and full-rank conditions follow from $u\mu_{\mathrm{app}} \le c$. The product $(\mathbf{I} + \mathbf{F}_{\mathrm{QR}})(\mathbf{I} + \mathbf{F}_\times)$ differs from the identity in spectral norm by at most $u\mu_{\mathrm{app}}$. Right multiplication by $\mathbf{E}'$ proves the similarity. For the absolute product error, operand preparation gives $\|\widetilde{\mathbf{X}} - \mathbf{X}\|_1 \le 2nuK$ for each operand. Expansion of their product and stable multiplication give $\|\widetilde{\mathbf{Y}} - \mathbf{R}_{\mathrm{sc}}\mathbf{M}\|_1 \le nuK^2(6 + 4\mu_\omega(n)) \le u\mu_{\mathrm{app}}$. □

C.3 GridSeysen

Algorithm 4 GridSeysen recursion

procedure GridSeysen($\mathbf{T}, \mathbf{E}, p$) ▷ $\mathbf{T}$ is the stored scaled factor
  if $n = 1$ **then**
    return $(\mathbf{I}, \mathbf{T})$
  split $\mathbf{T} = \begin{pmatrix} \mathbf{T}_L & \mathbf{T}_X \\ 0 & \mathbf{T}_R \end{pmatrix}$ and $\mathbf{E} = \mathrm{diag}(\mathbf{E}_L, \mathbf{E}_R)$ after the first $\lfloor n/2 \rfloor$ indices
  $(\mathbf{M}_L, \mathbf{T}'_L) \leftarrow$ GridSeysen$(\mathbf{T}_L, \mathbf{E}_L, p)$
  $(\mathbf{M}_R, \mathbf{T}'_R) \leftarrow$ GridSeysen$(\mathbf{T}_R, \mathbf{E}_R, p)$
  $\mathbf{T}'_X \leftarrow \mathrm{fl}_p(\mathbf{T}_X\mathbf{M}_R)$
  $\widetilde{\mathbf{X}}_{\mathrm{sc}} \leftarrow \mathrm{Solve}_p(\mathbf{T}'_L, \mathbf{T}'_X)$ ▷ solve $\mathbf{T}'_L\mathbf{X} = \mathbf{T}'_X$
  $(M_{\mathrm{cross}})_{ij} \leftarrow \dfrac{(E_L)_{ii}}{(E_R)_{jj}} \left\lfloor -\dfrac{(E_R)_{jj}}{(E_L)_{ii}} (\widetilde{X}_{\mathrm{sc}})_{ij} \right\rceil$ for every entry of the upper-right block
  $\mathbf{T}^+_X \leftarrow \mathrm{fl}_p(\mathbf{T}'_X + \mathbf{T}'_L\mathbf{M}_{\mathrm{cross}})$
  $\mathbf{M}_X \leftarrow \mathrm{GridDecode}\big(\mathrm{fl}_p(\mathbf{M}_L\mathbf{M}_{\mathrm{cross}}), \mathbf{E}_L, \mathbf{E}_R\big)$
  $\mathbf{M} \leftarrow \begin{pmatrix} \mathbf{M}_L & \mathbf{M}_X \\ 0 & \mathbf{M}_R \end{pmatrix}$
  return $\left(\mathbf{M}, \begin{pmatrix} \mathbf{T}'_L & \mathbf{T}^+_X \\ 0 & \mathbf{T}'_R \end{pmatrix}\right)$

The next theorem bounds the precision needed for a Seysen-reduced output with unchanged exact QR diagonals.

Theorem 10 (Finite-precision GridSeysen). *Let $\mathbf{R}_{\mathrm{in}}$ be a stored positive-diagonal upper-triangular matrix of order n. Let $\mathbf{E}_0$ be a power-of-two scale satisfying*

$$\frac{9}{16}(E_0)_{ii} < (R_{\mathrm{in}})_{ii} < \frac{32}{9}(E_0)_{ii} \qquad (1 \le i \le n).$$

Fix $\bar{\sigma} \geq 0$, $K_{\mathrm{norm}} \geq 1$, and a local output-error tolerance $0 < \varepsilon_{\mathrm{out}} \leq c$. Assume

$$\mathrm{lfspread}(\mathbf{R}_{\mathrm{in}}) \leq \bar{\sigma}, \qquad \max\{\|\mathbf{R}_{\mathrm{in}}\mathbf{E}_0^{-1}\|_1, \|\mathbf{E}_0\mathbf{R}_{\mathrm{in}}^{-1}\|_1\} \leq K_{\mathrm{norm}}.$$

There is a sufficient precision $P_{\mathrm{GS}} = P_{\mathrm{GS}}(n, \bar{\sigma}, K_{\mathrm{norm}}, \varepsilon_{\mathrm{out}})$ such that

$$P_{\mathrm{GS}} = \widetilde{O}\big(1 + \log K_{\mathrm{norm}} + \log(2n)\bar{\sigma} + \log(1/\varepsilon_{\mathrm{out}})\big).$$

At every $p \geq P_{\mathrm{GS}}$, a GridSeysen *operation returns an output scale $\mathbf{E}_1$, a matrix $\mathbf{M}$, and a stored positive-diagonal triangular factor $\widetilde{\mathbf{R}}_{\mathrm{out}}$, with $\mathbf{E}_1 = \mathbf{E}_0$. The represented update*

$$\mathbf{U} := \mathbf{E}_0^{-1}\mathbf{M}\mathbf{E}_1 \in \mathrm{GL}_n(\mathbb{Z})$$

is upper unitriangular. Let $\mathbf{R}_{\mathrm{out}}$ be the positive-diagonal R-factor of $\mathbf{R}_{\mathrm{in}}\mathbf{U}$. Then $\mathbf{E}_1$ is admissible for $\mathbf{R}_{\mathrm{out}}$ and $\widetilde{\mathbf{R}}_{\mathrm{out}}$. Both $(\mathbf{R}_{\mathrm{out}}, \mathbf{E}_1)$ and $(\widetilde{\mathbf{R}}_{\mathrm{out}}, \mathbf{E}_1)$ are Seysen-reduced with parameter $C_{\mathrm{S}}\, 2^{\bar{\sigma}}$.

The output satisfies

$$\mathbf{R}_{\mathrm{in}}\mathbf{U} \approx_{\varepsilon_{\mathrm{out}}} \widetilde{\mathbf{R}}_{\mathrm{out}}.$$

The exact scaled output also satisfies the local coefficient bound (3.8) at every recursive split. Under the exponent bounds of Lemma 6, the local bit cost is $\widetilde{O}(n^{\omega}p)$.

Proof. Write $\mathbf{T}_J = \mathbf{R}_{\mathrm{in},J}\mathbf{E}_J^{-1}$ for the scaled input at a node J, and $(\mathbf{M}_J, \widetilde{\mathbf{T}}_J)$ for its returned pair. The exact scaled output is $\mathbf{T}_J^{\star} = \mathbf{T}_J\mathbf{M}_J$.

Structure and diagonals. Each return matrix copies the two child return matrices into its diagonal blocks and grid-decodes only its upper-right block. Induction from the identity returns at leaves therefore gives

$$\mathbf{E}_J^{-1}\mathbf{M}_J\mathbf{E}_J \in \mathrm{GL}_{|J|}(\mathbb{Z}) \quad \text{upper unitriangular},$$
$$\operatorname{diag}\widetilde{\mathbf{T}}_J = \operatorname{diag}\mathbf{T}_J^{\star} = \operatorname{diag}\mathbf{T}_J.$$

Thus the input scale remains admissible for both outputs, and their exact QR diagonals are unchanged.

Operand bounds. Put $\eta = C_{\mathrm{S}}2^{\bar{\sigma}}$ and

$$K_0 = C_{\mathrm{S}}n\max\{2, K_{\mathrm{norm}}, 2^{\bar{\sigma}}, S_{\eta}(n)\}, \qquad K = (8K_0)^{16}.$$

Principal triangular blocks and their inverses inherit the input norm bound. Suppose the two child outputs are Seysen-reduced with parameter η. Lemma 4 bounds their scaled factors and inverses by K_0. The identity $\mathbf{M}_H = \mathbf{T}_H^{-1}\mathbf{T}_H^{\star}$ bounds each child return and its inverse by K_0^2. The crossing product has norm $O(K_0^3)$, its solved coefficient matrix has norm $O(K_0^4)$, and rounding adds at most $4n2^{\bar{\sigma}}$ to the shear norm. The block formulas for the return and its inverse then bound all remaining block operands by K. These norm bounds also cover prepared operands, using (C.3) with relative error at most c for every block that is inverted. The bounds are uniform over the recursion and give

$$\log K = O\big(1 + \log K_{\mathrm{norm}} + \log(2n)\bar{\sigma} + \log^2(2n)\big)\,. \tag{C.4}$$

Representation error. Track only

$$\mathbf{F}_J = \widetilde{\mathbf{T}}_J - \mathbf{T}_J\mathbf{M}_J, \qquad e_J = \|\mathbf{F}_J\|_1.$$

At a nonleaf node, omit J from the algorithm's block notation and write $\mathbf{S} = \mathbf{M}_{\mathrm{cross}}$. Define the local residuals by

$$\begin{aligned}
\mathbf{T}'_X &= \mathbf{T}_X\mathbf{M}_R + \mathbf{A},\\
\mathbf{T}^+_X &= \mathbf{T}'_X + \mathbf{T}'_L\mathbf{S} + \mathbf{B},\\
\mathbf{M}_X &= \mathbf{M}_L\mathbf{S} + \mathbf{D},\\
\mathbf{r} &= \mathbf{T}'_L\widetilde{\mathbf{X}}_{\mathrm{sc}} - \mathbf{T}'_X.
\end{aligned}$$

The exact product $\mathbf{M}_L\mathbf{S}$ lies on the output grid, so nearest-grid rounding increases its multiplication error by a factor of at most two. Theorem 7, including operand preparation and the solve, gives a computable nondecreasing factor $\mu_{\mathrm{loc}}(n, K) \geq 1$ such that

$$\begin{aligned}
&\max\{\|\mathbf{A}\|_1, \|\mathbf{B}\|_1, \|\mathbf{r}\|_1, \tfrac{1}{2}\|\mathbf{D}\|_1\} \leq u\mu_{\mathrm{loc}},\\
&\log \mu_{\mathrm{loc}} = O\big(\log^2(2n) + \log(2n)\log K\big).
\end{aligned}$$

Here the solve may be implemented by recursive inversion and multiplication. Subtraction of the exact block product gives

$$\mathbf{F}_J = \begin{pmatrix} \mathbf{F}_L & \mathbf{A} + \mathbf{F}_L\mathbf{S} + \mathbf{B} - \mathbf{T}_L\mathbf{D} \\ 0 & \mathbf{F}_R \end{pmatrix}.$$

Consequently, with $\mu_{\mathrm{node}} = 4K(1 + \mu_{\mathrm{loc}})$,

$$e_J \leq \mu_{\mathrm{node}}(u + e_L + e_R), \qquad e_J = 0 \quad (|J| = 1). \tag{C.5}$$

Let $\mu_{\mathrm{GS}}(1) = 1$ and, for $q > 1$, define

$$\mu_{\mathrm{GS}}(q) = \mu_{\mathrm{node}}\big(1 + \mu_{\mathrm{GS}}(\lfloor q/2 \rfloor) + \mu_{\mathrm{GS}}(\lceil q/2 \rceil)\big).$$

This nondecreasing bound satisfies $\log \mu_{\mathrm{GS}}(n) = O(\log(2n)(1 + \log \mu_{\mathrm{node}}))$. Choose P_{GS} large enough for the arithmetic thresholds and

$$\varepsilon_{\mathrm{GS}} := 2^{-P_{\mathrm{GS}}}\mu_{\mathrm{GS}}(n) \leq \frac{c\varepsilon_{\mathrm{out}}}{nK^3}. \tag{C.6}$$

Together with (C.4), this gives the stated precision bound, computed before the recursion starts.

Coefficient bounds and induction. Fix $p \geq P_{\mathrm{GS}}$. Induct on $|J|$, maintaining Seysen reduction of both outputs and $e_J \leq u\mu_{\mathrm{GS}}(|J|)$. The leaf case is exact. At a nonleaf node, the child conclusions justify the operand bounds above, and (C.5) bounds e_J. For

$$g_J = \frac{1}{2}\max_{i,j}\frac{(E_L)_{ii}}{(E_R)_{jj}} \leq 4\,2^{\bar{\sigma}},$$

grid rounding gives $\|\widetilde{\mathbf{X}}_{\mathrm{sc}} + \mathbf{S}\|_{\max} \le g_J$. The stored coefficient block satisfies the identity

$$\widetilde{\mathbf{X}} := (\mathbf{T}'_L)^{-1}\mathbf{T}^+_X = \widetilde{\mathbf{X}}_{\mathrm{sc}} + \mathbf{S} + (\mathbf{T}'_L)^{-1}(\mathbf{B} - \mathbf{r}).$$

Hence $\|\widetilde{\mathbf{X}}\|_{\max} \le g_J + 2Ku\mu_{\mathrm{loc}}$. For the exact coefficient block $\mathbf{X} = (\mathbf{T}^\star_J)^{-1}_L(\mathbf{T}^\star_J)_X$,

$$\mathbf{X} - \widetilde{\mathbf{X}} = (\mathbf{T}^\star_J)^{-1}_L\big((\mathbf{F}_J)_L\widetilde{\mathbf{X}} - (\mathbf{F}_J)_X\big).$$

The child inverse bound and $\|\widetilde{\mathbf{X}}\|_1 \le K$ give $\|\mathbf{X}-\widetilde{\mathbf{X}}\|_{\max} \le 2K^2 e_J$. The precision choice makes $2Ku\mu_{\mathrm{loc}} + 2K^2 e_J < 1$. Thus $\|\mathbf{X}\|_{\max} \le g_J + 1$, proving (3.8), and both new coefficient blocks are bounded by $C_{\mathrm{S}}2^{\bar\sigma}$. The child blocks are copied, so this proves Seysen reduction at every split and closes the induction. Lemma 4 also supplies the output norm bounds used by the next node.

Accuracy and cost. At the root, put $\mathbf{H} = \widetilde{\mathbf{T}}(\mathbf{T}^\star)^{-1}$. Then

$$\|\mathbf{H} - \mathbf{I}\|_1 = \|\mathbf{F}(\mathbf{T}^\star)^{-1}\|_1 \le K\varepsilon_{\mathrm{GS}}, \qquad \|\mathbf{H} - \mathbf{I}\|_2 \le \sqrt{n}K\varepsilon_{\mathrm{GS}} \le \varepsilon_{\mathrm{out}}.$$

Multiplication by the common right scale gives the required similarity. Equation (C.2) bounds the return encoding. The balanced recurrence $T(n) \le T(\lfloor n/2\rfloor) + T(\lceil n/2\rceil) + \widetilde{O}(n^\omega p)$ gives cost $\widetilde{O}(n^\omega p)$. □

C.4 ReduceTwo

Algorithm 5 implements Lemma 5 on the supplied scaled input. Its internal QR selects the output scale. The surrounding Apply computes and stores the updated factor with its own direct-error tolerance. Write $\mathbf{E}_0 = \mathrm{diag}(2^{e_1}, 2^{e_2})$ and $\mathbf{R}_{\mathrm{sc}} = \mathbf{R}\mathbf{E}_0^{-1}$. The initial DiagonalTest returns `stop` only if $\mathrm{lfspread}(\mathbf{R}) \le c^2\vartheta/32$. Otherwise $R_{11} > R_{22}$. The rounding step [23, Algorithm 1] uses one power-of-two grid per column, retaining p bits of its largest entry. Planar reduction uses [27] and retains column order on equal norms.

Algorithm 5 ReduceTwo with the scaled representation

Input: $(\mathbf{R}_{\mathrm{sc}}, \mathbf{E}_0)$, $\bar\sigma$, ϑ satisfying the scaled-input hypotheses of Lemma 5, and $p \ge P_2(\bar\sigma, \vartheta)$

$x \leftarrow c^2\vartheta$
if DiagonalTest$(\mathbf{R}_{\mathrm{sc}}, \mathbf{E}_0, x) =$ `stop` **then**
  return $(\mathbf{I}, \mathbf{E}_0)$ ▷ already within the lfspread target
$e_\star \leftarrow e_2$, $\mathbf{D}_0 \leftarrow 2^{-e_\star}\mathbf{E}_0$ ▷ keep the common scale implicit
$\mathbf{R}^\circ \leftarrow \mathbf{R}_{\mathrm{sc}}\mathbf{D}_0$
$\widehat{\mathbf{R}} \leftarrow$ RoundColumns$_p(\mathbf{R}^\circ)$ ▷ round on each column's grid
$\mathbf{V} \leftarrow$ PlanarReduce$(\widehat{\mathbf{R}})$
$\rho \leftarrow \|(\widehat{\mathbf{R}}\mathbf{V})_1\|_2^2 / \|\widehat{\mathbf{R}}_1\|_2^2$ ▷ exact on the rounded input
if $\rho \ge (1 - 3x/2)^2$ **then**
  return $(\mathbf{I}, \mathbf{E}_0)$
$\widetilde{\mathbf{R}}^\circ_V \leftarrow$ computeR$_p(\mathbf{R}^\circ\mathbf{V})$
$\mathbf{D}_1 \leftarrow$ scale$(\widetilde{\mathbf{R}}^\circ_V)$
return $(\mathbf{D}_0\mathbf{V}\mathbf{D}_1^{-1}, 2^{e_\star}\mathbf{D}_1)$

Lemma 15 (Planar reduction by rounding). *Let* $\mathbf{A}$ *be a positive-diagonal upper-triangular matrix of order two, with* $\max\{\|\mathbf{A}\|_2, \|\mathbf{A}^{-1}\|_2\} \le 2^b$ *for* $b \ge 1$*, and let* $0 < x < 1/4$*. Reduce by rounding [23, Algorithm 1], with Yap's algorithm [27] as the exact planar reduction, uses a sufficient precision* $p = \widetilde{O}(b + \log(1/x))$*. The resulting* $\mathbf{V} \in \mathrm{GL}_2(\mathbb{Z})$ *makes the exact R-factor* $\mathbf{T}$ *of* $\mathbf{AV}$ *satisfy*

$$\mathrm{lfspread}(\mathbf{T}) \le a_{\mathrm{LLL}} + x, \qquad |T_{12}|/T_{22} = O(1).$$

Proof. Let $\widehat{\mathbf{A}}$ be the rounded input. Columnwise rounding gives $\|\widehat{\mathbf{A}} - \mathbf{A}\|_2 \le 2^{-p+b+O(1)}$. For $p \ge 2b + \log(1/x) + O(1)$, the inverse perturbation bound gives

$$\mathbf{A} = (\mathbf{I} + \mathbf{F})\widehat{\mathbf{A}}, \qquad \|\mathbf{F}\|_2 \le 2^{-p+2b+O(1)} \le x/64.$$

Let $\widehat{\mathbf{T}}$ be the exact R-factor of $\widehat{\mathbf{A}}\mathbf{V}$. Gauss reduction gives $\widehat{T}_{11}/\widehat{T}_{22} \le 2/\sqrt{3}$ and $|\widehat{T}_{12}| \le \widehat{T}_{11}/2$. The same left perturbation relates $\mathbf{AV}$ and $\widehat{\mathbf{A}}\mathbf{V}$, so Theorem 8 gives $\mathbf{T} = (\mathbf{I} + \mathbf{H})\widehat{\mathbf{T}}$ with $\|\mathbf{H}\|_F \le x/16$. Consequently,

$$\mathrm{lfspread}(\mathbf{T}) \le \tfrac{1}{2}\log_2(4/3) + \log_2 \frac{1 + x/16}{1 - x/16} \le a_{\mathrm{LLL}} + x,$$

$$\frac{|T_{12}|}{T_{22}} \le \frac{(1 + x/16)/\sqrt{3} + x/16}{1 - x/16} = O(1).$$

□

Proof of Lemma 5. Rounding and rejection. First assume the scaled-input hypotheses. Write $d_i = R_{ii}$ and $x = c^2\vartheta$. Directed evaluation of $\log_2(d_1/d_2)$ to absolute error $x/64$ implements DiagonalTest. Its identity return meets all the claims. Otherwise $d_1 > d_2$, and the lfspread and Seysen bounds give $d_1/d_2 \le 2^{\bar{\sigma}}$ and $|R_{12}|/d_2 = O(C_S 2^{\bar{\sigma}})$. Consequently $\mathbf{R}^\circ = 2^{-e_2}\mathbf{R}$ and its inverse have norms $2^{O(1+\bar{\sigma})}$, and its column exponents differ by $O(1 + \bar{\sigma})$.

Lemma 15 applied to $\mathbf{R}^\circ$ gives $P_2 = \widetilde{O}(1 + \bar{\sigma} + \log(1/\vartheta))$. Let $\mathbf{R}_V$ be the exact R-factor of $\mathbf{RV}$. Common scaling leaves its lfspread and coefficient ratio unchanged, so it satisfies the lemma's lfspread and coefficient-ratio bounds. Increasing the constant in P_2, Lemma 7 gives $\mathbf{R}^\circ \approx_{x/128} \widehat{\mathbf{R}}$. Similarity bounds every vector norm, so the exact test $\rho < (1 - 3x/2)^2$ gives

$$\text{acceptance} \implies R_{V,11} < (1 - x)d_1, \qquad \text{rejection} \implies R_{V,11} \ge (1 - 2x)d_1.$$

On acceptance, $\mathbf{R}' = \mathbf{R}_V$ gives the required lfspread and decrease. On rejection, determinant preservation gives

$$\mathrm{lfspread}(\mathbf{R}) \le \mathrm{lfspread}(\mathbf{R}_V) - 2\log_2(1 - 2x) \le a_{\mathrm{LLL}} + O(c^2\vartheta) < a_{\mathrm{LLL}} + (1 - 2c)\vartheta,$$

so returning the identity, with $\mathbf{R}' = \mathbf{R}$, is also valid.

Profile monotonicity. For an accepted update $\mathbf{U} = \mathbf{V}$, take any positive-diagonal upper-triangular $\mathbf{S}$ with $\mathbf{S} \approx_{x/64} \mathbf{R}$, and let $\mathbf{S}'$ be the exact R-factor of $\mathbf{SU}$. Lemma 1, before and after the same update, gives

$$\frac{S'_{11}}{S_{11}} \le (1 - x)\frac{1 + x/64}{1 - x/64} < 1.$$

Unimodularity and strict decrease force $U_{21} \neq 0$, so $S_{22} \leq S'_{11} < S_{11}$. Determinant preservation then gives $S_{22} < S'_{22} \leq S_{11}$, proving (3.3). Identity returns satisfy them with equality.

Output scale and cost. Clearing the rounded input's common dyadic denominator gives $O(p)$-bit integer entries, so Yap's reduction and the exact norm comparison cost $\widetilde{O}(p)$. Planar reduction does not increase the longest column norm. The input norm bounds and unimodularity therefore give $\|\mathbf{V}^{\pm 1}\|_2 = 2^{O(1+\bar{\sigma})}$. At this precision, $\widetilde{\mathbf{R}}^{\circ}_V$ is the exact R-factor of $(\mathbf{I}+\mathbf{G})\mathbf{R}^{\circ}\mathbf{V}$ for some $\|\mathbf{G}\|_S \leq x/128$. Theorem 8 makes the returned scale $\mathbf{E}_1 = 2^{e_2}\,\mathrm{scale}(\widetilde{\mathbf{R}}^{\circ}_V)$ admissible for $\mathbf{R}_V$. The coefficient bound in Lemma 15 gives the required Seysen bound. An identity return retains both the input scale and its Seysen bound. The return satisfies $\mathbf{M} = \mathbf{E}_0\mathbf{V}\mathbf{E}_1^{-1}$ on acceptance. The common scale cancels, and all remaining arithmetic uses $O(p)$-bit significands and costs $\widetilde{O}(p)$.

For an arbitrary positive-diagonal triangular input of order two, choose the canonical input scale and take $\bar{\sigma}$ large enough to satisfy the scaled-input hypotheses. The same construction then applies. □

D Recursive analysis

Algorithm 6 expands the sketch in Algorithm 2 using the call and update rules of Section 3.5. Here GridSeysenCall supplies the entry and exit GridSeysen operations and composes the returned update, as in the sketch. Algorithm 7 gives this procedure and Apply.

We establish guarantees 1–7 of Section 3.6 using $\varepsilon_{\mathrm{call}} = c^5\gamma^3/n^5_{\mathrm{root}}$. We first derive the error and precision bounds assuming valid completed prefixes, child returns, and return-composition suffixes. We then verify these assumptions by joint induction in Theorems 11 and 12.

At each direct-operation prefix, use the exact factors $\mathbf{R}_A, \mathbf{R}_{A,\mathrm{ref}}$ from (3.16), with this call's recorded product $\mathbf{U}_t$ and fixed supplied input $\mathbf{R}^{\mathrm{in}}_A$. The stored factor's local error obeys (3.11), with zero error at entry. Grid decoding at return obeys (3.13).

To count error transfers, contract each GridSeysenCall to one node. Algorithm 6 gives at most $h_{\mathrm{root}}+1$ edges per path: all edges reduce $\lceil \log_2 n \rceil$ except a possible Reduce-to-Merge edge, which occurs at most once because merges have only merge descendants.

D.1 Reference errors and inherited margins

Let $H = h_{\mathrm{root}} + 1$ bound the remaining call depth. Allocate the reference-error tolerance by setting

$$\varepsilon_h = 5^{-(h+1)}\varepsilon_{\mathrm{ref}} \quad (0 \leq h \leq H), \qquad a = \sqrt{n_{\mathrm{root}}}\varepsilon_{\mathrm{call}}.$$

A call with at most h further child edges allows reference inputs with

$$\|\mathbf{R}^{\mathrm{in}}(\mathbf{R}^{\mathrm{in}}_{\mathrm{ref}})^{-1} - \mathbf{I}\|_F \leq \varepsilon_h. \tag{D.1}$$

Algorithm 6 Recursive reduction

procedure ROOT($\mathbf{B}, \delta, \varepsilon_{\text{out}}$)
  if $n_{\text{root}} = 1$ **then**
    return $\mathbf{B}$ rounded columnwise to relative error $\varepsilon_{\text{out}}/64$, and [1]
  $\varepsilon_{\text{init}} \leftarrow c^2\delta/n_{\text{root}}^2$
  Sort columns by s_j to form $\mathbf{B}^{\text{sort}} = \mathbf{BP}$ and initialize $(\mathbf{R}^{\text{init}}\mathbf{E}_{\text{init}}^{-1}, \mathbf{E}_{\text{init}})$
  $(\mathbf{M}_{\text{loc}}, \mathbf{E}_{\text{out}}) \leftarrow \text{GRIDSEYSENCALL}_{[n_{\text{root}}]}(\text{REDUCE}([n_{\text{root}}]))$
  $\mathbf{U} \leftarrow \mathbf{P}\mathbf{E}_{\text{init}}^{-1}\mathbf{M}_{\text{loc}}\mathbf{E}_{\text{out}}$ in dyadic representation
  Form $\widetilde{\mathbf{C}}$ by (3.14)
  return $\widetilde{\mathbf{C}}, \mathbf{U}$

procedure REDUCE(I)
  if $|I| = 1$ **then return**
  if $|I| = 2$ **then**
    $\text{APPLY}_I(\text{REDUCETWO}(I, \gamma))$ **return**
  for $J = I_{\text{L}}, I_{\text{R}}$ **do**
    $(\mathbf{M}, \mathbf{E}') \leftarrow \text{GRIDSEYSENCALL}_J(\text{REDUCE}(J))$
    $\text{APPLY}_I(\text{emb}_J(\mathbf{M}, \mathbf{E}'))$
  $(\mathbf{M}, \mathbf{E}') \leftarrow \text{GRIDSEYSENCALL}_I(\text{MERGE}(I_{\text{L}}, I_{\text{R}}))$
  $\text{APPLY}_I(\mathbf{M}, \mathbf{E}')$

procedure MERGE(I, J)
  $A \leftarrow I \cup J$, $n \leftarrow |A|$
  if $n = 2$ **then**
    $\text{APPLY}_A(\text{REDUCETWO}(A, \gamma))$ **return**
  $\bar{\sigma}_{\text{entry}} \leftarrow \text{SPREADBOUND}(A)$, $L_{\text{merge}} \leftarrow L(n, \bar{\sigma}_{\text{entry}})$
  for $t = 1, \ldots, L_{\text{merge}}$ **do**
    for $(I', J') = (I_{\text{R}}, J_{\text{L}}), (I_{\text{L}}, I_{\text{R}}), (J_{\text{L}}, J_{\text{R}})$ **do**
      if $I' \neq J'$ **then**
        $B \leftarrow I' \cup J'$
        $(\mathbf{M}, \mathbf{E}') \leftarrow \text{GRIDSEYSENCALL}_B(\text{MERGE}(I', J'))$
        $\text{APPLY}_A(\text{emb}_B(\mathbf{M}, \mathbf{E}'))$
    $\text{APPLY}_A(\text{GRIDSEYSEN}(A))$
    $\bar{\sigma}_t \leftarrow \text{SPREADBOUND}(A)$
    if $\bar{\sigma}_t \le a_n^{\text{stop}}(n-1)$ **then return**
  fail with $\perp$

Lemma 16 (Reference errors and margins). *Consider completed prefixes of the execution from* ROOT *in Algorithm 6, satisfying (3.11) and the local routines' output guarantees. Assume (D.1) at each initial reference input, and pass reference blocks at child boundaries. Then the following hold.*

1. *At every completed direct prefix of a call with remaining depth h,*

$$\|\widetilde{\mathbf{R}}_t(\mathbf{R}_{\text{ref},t})^{-1} - \mathbf{I}\|_F \le 4(\varepsilon_h + a) \le 5\varepsilon_h \le \varepsilon_{\text{ref}}. \tag{D.2}$$

   If $h \ge 1$, the child input satisfies (D.1) with $h-1$. Corresponding log-diagonals on the two exact chains differ by at most $-\log_2(1-2\varepsilon_h)$, including after return decoding.

Algorithm 7 Applying updates and composing returns

procedure APPLY$_A$(update)
  $j \leftarrow j+1$, choose p to meet η_j^{dir}
  obtain $(\mathbf{M}, \mathbf{E}')$ from update, evaluating a local routine at precision p
  if update is a local GRIDSEYSEN call **then**
    $\widetilde{\mathbf{R}}_{\mathrm{sc}} \leftarrow$ the factor supplied by Algorithm 4
  else
    $\widetilde{\mathbf{R}}_{\mathrm{sc}} \leftarrow \mathrm{computeR}_p\big(\mathrm{fl}_p(\widetilde{\mathbf{R}}_{\mathrm{sc}}\mathbf{M})\big)$ on the full block A
  $\mathbf{E}_A \leftarrow \mathbf{E}'$, append $(\mathbf{M}, \mathbf{E}')$ to $\mathcal{H}$

procedure GRIDSEYSENCALL$_A$(body)
  $\mathbf{E}_0 \leftarrow \mathbf{E}_A$, $\mathcal{H} \leftarrow [\,]$, $j \leftarrow 0$
  APPLY$_A$(GRIDSEYSEN(A))
  execute body in this call's state, propagating any failure
  APPLY$_A$(GRIDSEYSEN(A))
  return COMPOSERETURNS($\mathbf{E}_0, \mathcal{H}$)

2. *The algorithm's current scale satisfies*

$$\frac{3}{4}E_{jj} \le S_{jj} < \frac{8}{3}E_{jj} \quad \textit{for } \mathbf{S} \in \{\widetilde{\mathbf{R}}, \mathbf{R}\}. \tag{D.3}$$

   It is also admissible for every allowed reference factor.
3. *Suppose the last successful reduction of a nonleaf block A' gives* lfspread *at most $(a_{n_{A'}} - 2c\gamma)(n_{A'} - 1)$ on its exact chain. Until the next update that changes its exact diagonals, the stored factor, the exact factor from the call's input, and every allowed reference factor have* lfspread *on A' at most $(a_{n_{A'}} - c\gamma)(n_{A'} - 1)$.*

Proof. Reference transfer. Write $\mathbf{R}^{\mathrm{in}} = (\mathbf{I} + \mathbf{F}_{\mathrm{in}})\mathbf{R}^{\mathrm{in}}_{\mathrm{ref}}$. Composing this perturbation with the call error gives Frobenius error at most $(1+\varepsilon_{\mathrm{call}})\varepsilon_h + a$ before exact QR. Theorem 8 gives (D.2). Restriction of a triangular relative factor to a contiguous block does not increase the Frobenius error. Since $5\varepsilon_h = \varepsilon_{h-1}$, the child assertion follows from $4a \le \varepsilon_H$, verified by

$$\frac{4a}{\varepsilon_H} = \frac{4c\,5^{H+1}}{n_{\mathrm{root}}^{5/2}} < 500c\, n_{\mathrm{root}}^{\log_2 5 - 5/2} < 1.$$

Here $H = \lceil \log_2 n_{\mathrm{root}} \rceil + 1$ and $c \le 1/1024$. Common-update invariance and Lemma 1 give the exact-diagonal comparison, which decoding preserves. Also $\log(1/\varepsilon_H) = O(\log n_{\mathrm{root}} + \log(1/\gamma))$, so the reference budget is covered by the working precision.

Scale and reduction margins. Compare two measurements of an unchanged exact diagonal using their calls' nearest common ancestor. Use its supplied input as reference and pass reference blocks along both paths. Equations (D.2) and (3.11) bound their log-diagonal discrepancy by $O(\varepsilon_{\mathrm{ref}} + a)$, independently of the number of intervening returns.

For a scale entry E_{jj}, use the diagonal $\widetilde{d}_j$ that selected it at initialization or an accepted ReduceTwo update. It satisfies $E_{jj} \le \widetilde{d}_j < 2E_{jj}$. Child returns carry this choice on their blocks. Between selections, GridSeysen, rejection, and decoding preserve exact diagonals, and embedded updates preserve them outside their blocks. The selecting planar QR adds $O(c^2\gamma/128)$ to the endpoint error. Thus selected and current diagonals differ by at most a factor $4/3$, proving (D.3). The reference comparison also keeps every allowed reference diagonal in the admissible interval.

For the last successful reduction of A', the same endpoint comparison changes its lfspread by $O(\varepsilon_{\mathrm{ref}} + a) < c\gamma(n_{A'} - 1)$. Subtracting this once from the $2c\gamma$ margin proves the final assertion. □

D.2 Operand norm bounds

Fix a call on A, with $n = |A|$, direct returns $\mathbf{M}_1, \dots, \mathbf{M}_s$, and scales $\mathbf{E}_0, \dots, \mathbf{E}_s$. Write $\mathbf{R}_i^{\mathrm{in}}$ and $\widetilde{\mathbf{R}}_i$ for the exact and stored factors before operation i. Operation i uses input scale $\mathbf{E}_i^{\mathrm{in}} = \mathbf{E}_{i-1}$. Let $\mathbf{M}_i^{\mathrm{pre}}$ be its return before local decoding, and let $\mathbf{R}_i^{\mathrm{pre}}$ be the exact R-factor of $\mathbf{R}_i^{\mathrm{in}}\mathbf{E}_{i-1}^{-1}\mathbf{M}_i^{\mathrm{pre}}\mathbf{E}_i$. For a received child return, $\mathbf{M}_i^{\mathrm{pre}} = \mathbf{M}_i$ is already decoded. Let K_{entry} bound the call's input scaled factor and inverse. All bounds below use completed prefixes and child guarantees.

lfspread *bounds.* Directed logs of the stored scaled diagonal give an upper bound $\widehat{\sigma}_i$ on $\mathrm{lfspread}(\widetilde{\mathbf{R}}_i)$ with error at most $\varepsilon_{\mathrm{stop}}/4$. By Lemma 16, the stored and exact input values of lfspread also differ by at most $\varepsilon_{\mathrm{stop}}/4$. The local output guarantees give $\mathrm{lfspread}(\mathbf{R}_i^{\mathrm{pre}}) \le \mathrm{lfspread}(\widetilde{\mathbf{R}}_i) + \varepsilon_{\mathrm{stop}}/4$: GridSeysen preserves diagonals, ReduceTwo uses its reference monotonicity, and child returns use the inductive profile inequalities on the updated block. Consequently the computable choice

$$\bar{\sigma}_i = \widehat{\sigma}_i + \varepsilon_{\mathrm{stop}}/4 \tag{D.4}$$

satisfies

$$\mathrm{lfspread}(\widetilde{\mathbf{R}}_i) \le \bar{\sigma}_i \le \mathrm{lfspread}(\widetilde{\mathbf{R}}_i) + \varepsilon_{\mathrm{stop}},$$
$$\max\{\mathrm{lfspread}(\mathbf{R}_i^{\mathrm{in}}), \mathrm{lfspread}(\mathbf{R}_i^{\mathrm{pre}})\} \le \bar{\sigma}_i.$$

Apply the same construction to the final stored factor. For the exact factor $\mathbf{R}^{\mathrm{fin}}$ after all direct updates, it gives

$$\mathrm{lfspread}(\mathbf{R}^{\mathrm{fin}}) \le \bar{\sigma}^{\mathrm{fin}} \le \mathrm{lfspread}(\mathbf{R}^{\mathrm{fin}}) + \varepsilon_{\mathrm{stop}}.$$

Lemma 6 implements these tests in $\widetilde{O}(n^2 p)$ bit operations. The diagonal logs and stopping threshold need $O(\log n + \log(1/\gamma))$ fractional bits.

Operand norms. Let $\bar{\sigma}_{\mathrm{seg}}$ be the stored lfspread bound at the start of a segment between consecutive GridSeysen operations, enlarged by $\varepsilon_{\mathrm{stop}}$. Profile monotonicity and reference transfer make it an upper bound throughout the segment. Put

$$\eta_{\mathrm{seg}} = 16C_{\mathrm{S}}2^{\bar{\sigma}_{\mathrm{seg}}}, \qquad K_{\mathrm{seg}} = C_{\mathrm{S}}nS_{\eta_{\mathrm{seg}}}(n), \qquad \mathcal{K}_{\mathrm{seg}} = (8C_{\mathrm{S}}nK_{\mathrm{seg}})^d,$$

where d is a sufficiently large fixed integer. Round each bound upward to a power of two. We use $K_i^{\mathrm{dir}} = \mathcal{K}_{\mathrm{seg}}$ for every direct operation in the segment and $K_i^{\mathrm{acc}} = \mathcal{K}_{\mathrm{seg}}$ for the suffix product beginning at its prefix $i-1$. For the first GRID-SEYSEN operation and the final multiplication $\mathbf{M}_1\mathbf{T}_2$, include $\max\{1, K_{\mathrm{entry}}\}^d$ in the bound, using the call-entry lfspread. These bounds cover the scaled factors, returns, products, and required inverses. In either case, writing K_i for the applicable bound gives

$$\log K_i = \widetilde{O}(1 + \log(2n)\max\{1, \bar{\sigma}_{\mathrm{seg}}\})\,, \tag{D.5}$$

with an additional $O(\log K_{\mathrm{entry}})$ for the two exceptional operations.

To verify this uniform choice, the first stored factor in a segment is Seysen-reduced. Each child's entry lfspread is at most $\bar{\sigma}_{\mathrm{seg}} + 1$, so its output bound in Theorem 12 also gives scaled norms at most K_{seg}. If K^{state} bounds the current scaled factors and inverses, Lemma 11 and block multiplication give bounds K^{ret} for the next return and its inverse, and $K_{\mathrm{next}}^{\mathrm{state}}$ for the updated factors and inverses:

$$K^{\mathrm{ret}} \le C_{\mathrm{S}}\sqrt{n}\,K^{\mathrm{state}}K_{\mathrm{seg}}, \qquad K_{\mathrm{next}}^{\mathrm{state}} \le C_{\mathrm{S}}nK_{\mathrm{seg}}(K^{\mathrm{state}})^2.$$

There are at most three child returns per segment. These recurrences and the fixed local formulas therefore use only constant powers of $C_{\mathrm{S}}nK_{\mathrm{seg}}$. The order-two case follows from Lemma 5. Increasing the fixed d covers all direct inputs and returns.

To bound a return product through its input and output factors, we must transfer the final stored norm bound through local error and decoding.

Lemma 17 (Transfer of scaled norm bounds). *Let $\mathbf{R}^{\mathrm{fin}}$ and $\widetilde{\mathbf{R}}$ be positive-diagonal upper-triangular factors of order $n \le n_{\mathrm{root}}$ with $\mathbf{R}^{\mathrm{fin}} \approx_{\varepsilon_{\mathrm{call}}} \widetilde{\mathbf{R}}$. For a positive diagonal scale $\mathbf{E}_s$, suppose*

$$\max\{\|\widetilde{\mathbf{R}}\mathbf{E}_s^{-1}\|_1, \|\mathbf{E}_s\widetilde{\mathbf{R}}^{-1}\|_1\} \le K.$$

Let $\mathbf{C}_1$ be upper unitriangular with $\|\mathbf{C}_1 - \mathbf{I}\|_1 < c$, and put $\mathbf{R}^{\mathrm{out}} = \mathbf{R}^{\mathrm{fin}}\mathbf{E}_s^{-1}\mathbf{C}_1\mathbf{E}_s$. Then

$$\max\{\|\mathbf{R}^{\mathrm{out}}\mathbf{E}_s^{-1}\|_1, \|\mathbf{E}_s(\mathbf{R}^{\mathrm{out}})^{-1}\|_1\} \le 4K.$$

Proof. For $\mathbf{H} = \widetilde{\mathbf{R}}(\mathbf{R}^{\mathrm{fin}})^{-1}$, Theorem 8 and the call similarity give

$$\|\mathbf{H} - \mathbf{I}\|_1 \le \frac{\sqrt{2nn_{\mathrm{root}}}\,\varepsilon_{\mathrm{call}}}{1 - \varepsilon_{\mathrm{call}}} < c, \qquad \max\{\|\mathbf{H}\|_1, \|\mathbf{H}^{-1}\|_1\} \le 2.$$

Neumann's estimate gives the same norm bound for $\mathbf{C}_1$ and its inverse. The scaled output

$$\mathbf{R}^{\mathrm{out}}\mathbf{E}_s^{-1} = \mathbf{H}^{-1}(\widetilde{\mathbf{R}}\mathbf{E}_s^{-1})\mathbf{C}_1$$

and its inverse therefore have 1-norm at most $4K$. □

Write $\mathbf{M}_i^{\mathrm{suf}} = \mathbf{M}_i \cdots \mathbf{M}_s$ and $\mathbf{M}_i^{\mathrm{acc}} = \mathbf{M}_i \mathbf{T}_{i+1}$. For the backward recurrence (3.12), put

$$\boldsymbol{\Delta}_i = (\mathbf{M}_i^{\mathrm{acc}})^{-1}(\mathbf{T}_i - \mathbf{M}_i^{\mathrm{acc}}), \qquad \mathbf{C}_s = \mathbf{I}, \qquad \mathbf{C}_i = \mathbf{C}_{i+1}(\mathbf{I} + \boldsymbol{\Delta}_i).$$

Induct over decreasing i, assuming only the completed steps $j > i$ have integral upper-unitriangular corrections $\mathbf{E}_s^{-1}(\mathbf{I} + \boldsymbol{\Delta}_j)\mathbf{E}_s$ and $\|\boldsymbol{\Delta}_j\|_1 \le c\varepsilon_{\mathrm{call}}/s$. The ordered recurrence gives $\mathbf{T}_{i+1} = \mathbf{M}_{i+1}^{\mathrm{suf}} \mathbf{C}_{i+1}$, and

$$\|\mathbf{C}_{i+1}\|_1 \le (1 + c\varepsilon_{\mathrm{call}}/s)^s = O(1), \qquad \|\mathbf{C}_{i+1}^{-1}\|_1 \le (1 - c\varepsilon_{\mathrm{call}}/s)^{-s} = O(1). \tag{D.6}$$

If $\mathbf{R}_i^{\mathrm{acc}}$ is the exact factor after this suffix, then

$$\mathbf{R}_i^{\mathrm{acc}} = \mathbf{R}^{\mathrm{fin}} \mathbf{E}_s^{-1} \mathbf{C}_{i+1} \mathbf{E}_s.$$

Thus its diagonal equals that of $\mathbf{R}^{\mathrm{fin}}$, and the final scale and lfspread bound apply at every decoding step. The last GridSeysen step and Lemma 17 with $\mathbf{C}_1 = \mathbf{I}$ give scaled norms $O(K_{\mathrm{seg}})$ for $\mathbf{R}^{\mathrm{fin}}$, since its lfspread is at most the segment-entry value. The correction bounds (D.6) preserve this estimate. Lemma 11, applied between the exact prefix factor and this suffix output, now bounds $\mathbf{T}_{i+1}$, $\mathbf{M}_i^{\mathrm{acc}}$, and their inverses by $\mathcal{K}_{\mathrm{seg}}$. Only $i = 1$ uses the input before GridSeysen and hence K_{entry}. Once step i meets its tolerance, the same argument bounds $\mathbf{T}_i$ and its inverse. This proves (D.5) from completed data before each operation, with no growth proportional to the number of return products.

D.3 Selecting the working precision

Consider a call on A, with $n = |A|$, supplied factor $\mathbf{R}_A^{\mathrm{in}}$, scale $\mathbf{E}_A^{\mathrm{in}}$, and reference input satisfying (D.1). Assume the inherited scale margin and fix

$$\bar{\sigma}_{\mathrm{entry}} \ge \mathrm{lfspread}(\mathbf{R}_A^{\mathrm{in}}), \qquad K_{\mathrm{entry}} \ge \max\left\{\|\mathbf{R}_A^{\mathrm{in}}(\mathbf{E}_A^{\mathrm{in}})^{-1}\|_1, \|\mathbf{E}_A^{\mathrm{in}}(\mathbf{R}_A^{\mathrm{in}})^{-1}\|_1\right\}. \tag{D.7}$$

For a Merge body, the adjacent balanced sides I, J are profile-reduced in the supplied input. Write L_{merge} for its entry loop bound, and set $L_{\mathrm{merge}} = 0$ for Reduce and order-two calls.

For $K \ge 1$ and $b \ge 0$, define a common arithmetic error factor by

$$\ell = \lceil \log_2(2n) \rceil, \qquad \mathcal{A}(n, K, b) = 2^{C\ell^3(\ell + \lceil \log_2 K \rceil + \lceil b \rceil)},$$

where the fixed integer C is chosen from the arithmetic bounds in Theorem 7 and the construction of P_{GS}. This factor dominates μ_{app}, the reciprocals of the arithmetic thresholds, and the error amplification in GridSeysen. In particular,

$$\log \mathcal{A}(n, K, b) = \widetilde{O}(1 + \log K + b).$$

For a direct operation take $(K_i, b_i) = (K_i^{\mathrm{dir}}, \bar{\sigma}_i)$ and tolerance $\tau_i = \eta_i^{\mathrm{dir}}$. For a return product take $(K_i, b_i) = (K_i^{\mathrm{acc}}, \bar{\sigma}^{\mathrm{fin}})$ and

$$\tau_i = \frac{c^2}{K_i^{\mathrm{acc}}} \min\left\{2^{-\lceil \bar{\sigma}^{\mathrm{fin}} \rceil}, \frac{\varepsilon_{\mathrm{call}}}{s}\right\}, \qquad \rho_{\mathrm{dec},i} := \|\widetilde{\mathbf{M}}_{\mathrm{agg},i} - \mathbf{M}_i^{\mathrm{acc}}\|_1 \le \tau_i. \tag{D.8}$$

Here $\widetilde{\mathbf{M}}_{\mathrm{agg},i}$ is the computed product before grid projection. Its error is bounded by $2^{-p}\mathcal{A}(n, K_i, b_i)$. Choose the least integer precision satisfying

$$p_i \geq \left\lceil \log_2 \frac{\mathcal{A}(n, K_i, b_i)}{\tau_i} \right\rceil, \tag{D.9}$$

and $p_i \geq P_2(\bar{\sigma}_i, \gamma)$ when evaluating ReduceTwo. The fixed C includes guard bits for directed rounding. All inputs to this rule are known before the operation: the direct index fixes η_i^{dir}, and the completed history fixes s before return composition. Section D.2 supplies the norm and lfspread bounds from completed prefixes and suffixes. The rule satisfies

$$p_i = \widetilde{O}\left(\begin{array}{l} 1 + \log(1/\varepsilon_{\mathrm{call}}) \\ \quad + \begin{cases} \log(2i) + \log K_i^{\mathrm{dir}} + \log(2n)\bar{\sigma}_i, & \text{for a direct operation,} \\ \log(2s) + \bar{\sigma}^{\mathrm{fin}} + \log K_i^{\mathrm{acc}}, & \text{for a return product.} \end{cases} \end{array}\right). \tag{D.10}$$

Verification of the precision rule. For Apply and multiplication, use Lemma 14. For GridSeysen, let K_{GS} be the norm bound for intermediate matrices and their inverses. Equation (C.4) gives $\log K_{\mathrm{GS}} = O(\log K_i + \ell b_i + \ell^2)$, and the GridSeysen error recurrence gives $\log \mu_{\mathrm{GS}} = O(\ell^2 \log K_{\mathrm{GS}} + \ell^3)$. With a fixed C, (D.9) meets the GridSeysen arithmetic thresholds and (C.6) for $\varepsilon_{\mathrm{out}} = \tau_i$. A return product meets (D.8), so Lemma 13 makes $\mathbf{E}_s^{-1}(\mathbf{I} + \boldsymbol{\Delta}_i)\mathbf{E}_s$ integral upper unitriangular, with $\|\boldsymbol{\Delta}_i\|_1 \leq c\varepsilon_{\mathrm{call}}/s$.

Taking logarithms of the tolerances proves (D.10). The schedule gives $i, s = O(L_{\mathrm{merge}} + 1)$, accounting for the counter term. The extra planar precision is covered because $\log(1/\gamma) = O(\log(1/\varepsilon_{\mathrm{call}}))$. Finally,

$$\sum_{j \geq 1} \log_2(1 + \eta_j^{\mathrm{dir}}) \leq \frac{\varepsilon_{\mathrm{call}}}{8 \ln 2} \sum_{j \geq 1} j^{-2} < \log_2(1 + \varepsilon_{\mathrm{call}}).$$

Lemma 1 gives (3.11), counting each child return once. The resulting diagonal ratios lie in $(1/2, 2)$, as required by Lemma 6. □

D.4 Correctness and cost

Theorem 11 (Correctness of recursive calls). *Every call on A of order $n = |A|$ generated by* Root*, with initialization from Lemma 10, precision rule (D.9), and entry bounds (D.7), terminates. Allow any reference input satisfying (D.1).*

The call and its descendants satisfy the seven guarantees of Section 3.6, with scale margin (D.3) throughout. All local routines and return products meet their numerical hypotheses, with precisions bounded by (D.10) and return tolerance (D.8).

Let $\mathbf{M}_A$ and $\mathbf{E}_A^{\mathrm{out}}$ be the return matrix and output scale. Then $\mathbf{U} := (\mathbf{E}_A^{\mathrm{in}})^{-1}\mathbf{M}_A\mathbf{E}_A^{\mathrm{out}} \in \mathrm{GL}_n(\mathbb{Z})$. Every direct transition on either exact chain satisfies (3.3), as does grid decoding at return. These claims hold uniformly for every allowed reference input and its restrictions passed to descendants.

Theorem 12 (Output profiles, scales, and norm bounds). *For a call as in Theorem 11, let* $\mathbf{R}_A^{\mathrm{out}}, \mathbf{R}_{A,\mathrm{ref}}^{\mathrm{out}}$ *be its exact factors in* (3.16) *for the returned update* $\mathbf{U}$*, and let* $\widetilde{\mathbf{R}}_A$ *be its stored output factor. Each of these three factors* $\mathbf{S}$ *satisfies*

$$\mathrm{lfspread}(\mathbf{S}_{A',A'}) \le \big(a_{n_{A'}} - c\gamma\big)\,(n_{A'} - 1) \tag{D.11}$$

for $A' \in \mathcal{T}(A)$ *in* Reduce*, and* $A' \in \mathcal{T}(I) \cup \mathcal{T}(J) \cup \{A\}$ *in* Merge*.*

Put $\sigma_A = \mathrm{lfspread}(\mathbf{R}_A^{\mathrm{in}})$ *and*

$$\bar{\sigma}_{\mathrm{out}} := \min\{(a_n - c\gamma)(n-1),\, \sigma_A + 1\}, \qquad \eta_{\mathrm{out}} := 4C_{\mathrm{S}} 2^{\bar{\sigma}_{\mathrm{out}}}.$$

The returned scale $\mathbf{E}_A^{\mathrm{out}}$ *is admissible for all three factors and satisfies* (D.3) *for* $\mathbf{R}_A^{\mathrm{out}}$ *and* $\widetilde{\mathbf{R}}_A$*. The stored pair is Seysen-reduced with parameter* η_{out}*, and*

$$\max\left\{\|\mathbf{R}_A^{\mathrm{out}}(\mathbf{E}_A^{\mathrm{out}})^{-1}\|_1, \|\mathbf{E}_A^{\mathrm{out}}(\mathbf{R}_A^{\mathrm{out}})^{-1}\|_1\right\} = O(nS_{\eta_{\mathrm{out}}}(n))\,.$$

Proof of Theorems 11 and 12. Induct jointly on call order, with Merge before Reduce at equal order. We also maintain two facts about the earlier operations. Each scale entry comes from its canonical selection at initialization or an accepted ReduceTwo update. For every nonleaf block D in either side tree at a Merge entry, an earlier successful reduction gave lfspread at most $(a_{|D|} - 2c\gamma)(|D| - 1)$ on its own exact chain, and subsequent exact updates have preserved its diagonals. At return, the latter fact holds for every nonleaf block in the output trees of Theorem 12. Initialization supplies the scale choices and zero error. The root Reduce requires no reduced input blocks.

Through the direct operations, maintain (3.11), the scale margin (D.3), and profile monotonicity on both exact chains. Then induct over return products in decreasing index. After each local error estimate, Lemma 16 transfers the scale and reduction margins and the reference bounds (3.17)–(3.18) to the next prefix and child input.

Base case and child inputs. For order one, both GridSeysen operations and the composed return represent the identity. The similarity error is zero, and all claims follow.

The two children of Reduce require no reduced input blocks. The first child's updates preserve exact diagonals on the other half. Both inherit the scale choices, and their returns supply the successful reductions on both side trees needed by the same-order Merge.

In a merge round, the middle child's side trees lie in $\mathcal{T}(I_{\mathrm{R}})$ and $\mathcal{T}(J_{\mathrm{L}})$, so they inherit the entry history. The middle child's return supplies new successful reductions on those trees and leaves the diagonals on I_{L} and J_{R} unchanged. Thus the left child has the required history on both sides. Its return supplies new successful reductions on $\mathcal{T}(I)$ and preserves the right side's diagonals. The right child's return then supplies the reductions on $\mathcal{T}(J)$. These histories also give the next round's entry condition. Singleton sides need no reduction history. GridSeysen and return decoding preserve all these exact diagonals. Lemma 16 compares each current block with its last successful reduction, giving profile-reduced supplied inputs without accumulating a new loss at every call boundary.

Each child on B starts with zero local error and inherits the scale and reference bounds. Apply induction twice, using $(\mathbf{R}_A)_{B,B}$ and $(\mathbf{R}_{A,\mathrm{ref}})_{B,B}$ as reference inputs to the same supplied block $(\widetilde{\mathbf{R}}_A)_{B,B}$. The returned scale replaces the parent's scale on B, and the embedded update preserves exact diagonals elsewhere.

Direct operations. The segment bounds determine $\bar{\sigma}_t, K_t^{\mathrm{dir}}$ from completed prefixes and child returns before (D.9) chooses p_t. The inherited scale margin meets the input bounds of Theorem 10. Section D.3 then gives the local errors for GridSeysen and Apply. For a child return $\mathbf{V}_t$, common-update invariance preserves the preceding similarity bound when both the exact and stored factors are multiplied by $\mathbf{V}_t$. The parent's new multiplication and QR satisfy the local tolerance η_t^{dir} in (3.10). Composing these local errors yields (3.11).

The GridSeysen update is upper unitriangular, so (3.3) holds with equality. For a child return, induction and block QR transfer the inequalities from its two reference chains to the parent chains. For ReduceTwo, Lemma 16 makes both exact factors $\varepsilon_{\mathrm{ref}}$-similar to the stored input, since the spectral norm is at most the Frobenius norm. With $\varepsilon_{\mathrm{ref}} \le c^2\gamma/64$, Lemma 5 gives (3.3) on both chains. Its $2c\gamma$ output margin and Lemma 16 give (D.11). This supplies the order-two reduction history, including for identity returns. Accepted updates select canonical scales, and other updates retain or inherit these choices. The preceding prefix meets the scale, diagonal-comparison, and monotonicity hypotheses of Lemma 6, so the comparisons cost $\widetilde{O}(p_t)$.

Return decoding. At step i, the completed suffix satisfies the correction bounds of Section D.2 and determines K_i^{acc} before p_i is chosen. The precision-rule verification in Section D.3 makes $\mathbf{E}_s^{-1}(\mathbf{I}+\boldsymbol{\Delta}_i)\mathbf{E}_s$ integral upper unitriangular, with $\|\boldsymbol{\Delta}_i\|_1 \le c\varepsilon_{\mathrm{call}}/s$. The ordered product of these $s-1$ corrections therefore satisfies $\|\mathbf{C}_1 - \mathbf{I}\|_1 = O(c\varepsilon_{\mathrm{call}}) < c/n$ and $\mathbf{E}_s^{-1}\mathbf{C}_1\mathbf{E}_s \in \mathrm{GL}_n(\mathbb{Z})$ is upper unitriangular. Decoding leaves the stored factor and its error relative to $\mathbf{U}_s$ unchanged.

Profile margins and progress. For $n \ge 3$, child induction and Lemma 16 supply the side and middle-block path bounds required by Theorem 4: their orders $n' \le \lceil n/2 \rceil$ give $a_{n'} \le a_n - \gamma$, and their margins persist until the next diagonal-changing update. With SpreadBound, the margin estimates of Section 3.6 give (3.20), (3.21), and termination within the loop bound. Success gives exact lfspread at most $(a_n - 2c\gamma)(n-1)$, establishing the reduction history for A.

On each nonleaf side, the final recursive merge supplies the bounds on its entire tree. The final GridSeysen and return decoding preserve these diagonals, and the stopping test bounds lfspread on A. A Reduce call of order at least three ends with this merge. Thus the histories cover every nonleaf block in the returned trees. Lemma 16 transfers the bounds to all three factors, proving (D.11). Singletons have lfspread $= 0$.

Output scales and norms. Profile monotonicity gives $\mathrm{lfspread}(\mathbf{R}_A^{\mathrm{out}}) \le \sigma_A$. The diagonal comparisons change this bound by less than one, so (D.11) bounds lfspread for all three output factors by $\bar{\sigma}_{\mathrm{out}}$. Lemma 16 gives the output scale margins and admissibility, which return decoding preserves. The last GridSey-

SEN uses a bound at most $\bar{\sigma}_{\mathrm{out}} + \varepsilon_{\mathrm{stop}}$ and makes the stored pair Seysen-reduced with parameter η_{out}. Lemma 4 bounds its scaled norms by $K = O(nS_{\eta_{\mathrm{out}}}(n))$. The call error and return correction meet Lemma 17, which transfers this bound to $\mathbf{R}_A^{\mathrm{out}}$. Since $\bar{\sigma}_{\mathrm{out}} = O(1 + \min\{\sigma_A, n\})$, the definition (3.7) gives

$$\log K = O((1 + \min\{\sigma_A, n\}) \log(2n) + \log^2(2n)),$$

proving (3.19) also when the entry lfspread is $o(n)$.

Working precision and local cost. The charging rule in Section 3.6 assigns $O(1)$ operations to each call or round, including its children's first GridSeysen step and final return multiplication. These use entry norm bounds supplied by that parent. The computed lfspread bounds exceed the call or round's exact entry value σ by at most $O(1)$, so (D.5) and (D.10) give $p = \widetilde{O}(\max\{1, \sigma\})$, $O(p)$-bit significands, and operand norms at most $2^{O(p)}$. For the outer call, these two operations may add $\widetilde{O}(\log(2K_{\mathrm{entry}}))$ precision bits. Lemmas 6 and 5 give the local cost, including tests and updates to scales and counters. □

Corollary 1 (Bit cost of recursive calls). *Under the hypotheses of Theorem 11, let $\mathbf{R}_A^{\mathrm{out}}$ be the exact returned factor from Theorem 12. The total bit cost of the call, including descendants, is*

$$\widetilde{O}\left(\begin{array}{l} n^{\omega}\big(\log(2K_{\mathrm{entry}}) + \max\{1, \bar{\sigma}_{\mathrm{entry}}\} + \log(L_{\mathrm{merge}} + 1) + \log(1/\varepsilon_{\mathrm{call}})\big) \\ \quad + \gamma^{-3} n^{\omega-2}\big(\mathrm{Pot}(\mathbf{R}_A^{\mathrm{in}}) - \mathrm{Pot}(\mathbf{R}_A^{\mathrm{out}})\big) \end{array}\right). \tag{D.12}$$

Proof. Theorem 5 now gives (D.12), with the accuracy and loop-counter terms supplied by (D.10). □

For the root call, the initialization bounds and the profile monotonicity in Theorem 11, together with (D.5) and (D.10), bound the largest reduction precision by

$$p_{\max} = \widetilde{O}(1 + \log n + \log \mathrm{cond}_1(\mathbf{B}) + \log(1/\delta)).$$

The exponent overhead from Lemma 6 is included in the logarithmic factors of (3.27).

E Implementation and benchmarks

The implementation and benchmark data are available in the repository.

E.1 Implementation choices

The implementation retains our recursive framework and scaled representation. Following practical optimizations used in `flatter` [22, Section 5], we tune reduction targets and stopping rules, use `fplll` on eligible small blocks, and reuse computed factors to restrict size reduction and QR updates to affected blocks.

Working precision is computed dynamically from local operand bounds, and return products are accumulated exactly on their integer grids. The LLL updates used by `fplll` on these blocks are profile-monotone in exact arithmetic: size reductions preserve QR diagonals, and Lovász swaps satisfy the profile inequalities (3.3).

E.2 Experimental setup

We use `fplll`'s `latticegen` for q-ary and NTRU-like inputs [25]. The q-ary bases have block form $\left(\begin{smallmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{H} & q\mathbf{I} \end{smallmatrix}\right)$, where both diagonal blocks have order $n/2$ and $\mathbf{H}$ is uniform modulo q. In the NTRU-like family, $\mathbf{H}$ is a circulant multiplication matrix. For $n = 16, 32$, the modulus bit lengths are $256, 1024, 4096, 16384, 32768$. We use the first four values for $n = 64$ and the first two for $n = 128$.

To test changes in column scale at fixed κ_{col}, we use a scaled q-ary family. It starts from a 256-bit q-ary basis in each dimension $n = 16, 32, 64$. We multiply each column i independently by 2^{e_i}, where

$$e_i \leftarrow \mathrm{Unif}\{1, \ldots, S+1\}, \qquad S \in \{0, 256, 1024, 4096, 16384, 32768\}.$$

Every column is scaled, and S bounds the exponent span. For each seed and dimension, the same base is used at every S, preserving κ_{col}.

We also use three fixed inputs from `flatter`'s RSA high-bits generator, with 400, 450, and 480 unknown bits [3, 22]. Their dimensions are 5, 9, and 20.

Writing $\mathbf{C} = \mathbf{QR}$ for the output basis, we use the common quality requirements

$$\mathrm{lfspread}(\mathbf{R}) \le An, \qquad \kappa_{\mathrm{col}}(\mathbf{C}) \le n^2 2^{2n}, \qquad A = \log_2(2/\sqrt{3}) + 0.1.$$

Our target is $A(n-1)$, and `flatter` is configured for the lfspread bound An. For each output, we independently check lfspread, certify an upper bound on κ_{col}, and verify lattice preservation exactly. We use an AMD Ryzen 9 5900HX, with one thread per reducer and four independent cases running concurrently. For each generated parameter set, we use seeds 0, 1, and 2 and run each input once per implementation. Each fixed Coppersmith case is run once per implementation. We count retired user-space instructions over the reduction call, including construction of $\mathbf{U}$ and $\mathbf{BU}$ in our implementation. Each instruction ratio is the `flatter` count divided by ours, so a ratio above one means that ours uses fewer instructions.

E.3 Results

All 306 outputs meet the quality bounds and preserve the input lattice. Our implementation uses fewer instructions in all 153 paired executions. Figure 1 shows the ordinary q-ary and NTRU-like runs. Table 2 gives the scaled q-ary and Coppersmith results.

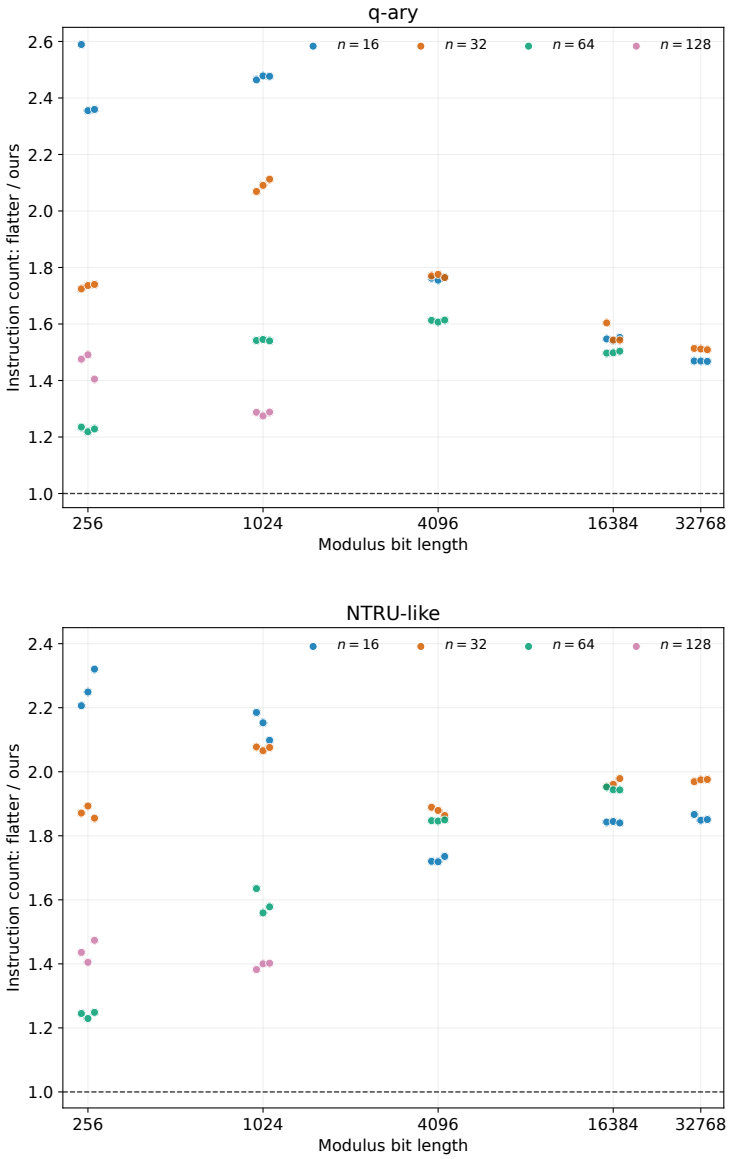

Fig. 1. Instruction-count ratios (`flatter`/ours) for ordinary q-ary inputs (top) and NTRU-like inputs (bottom). Each point represents one seed–case execution, with colors indicating dimension n. The horizontal axis is the modulus bit length on a log scale. The dashed line denotes equal counts.

Table 2. Scaled q-ary inputs (top) and Coppersmith inputs (bottom). Ratios are `flatter`'s instruction count divided by ours. The upper table also gives our counts in millions. Counts and ratios in that table are geometric means over three seeds. S bounds the rescaling exponent span. Bits is the number of unknown RSA bits.

	$n = 16$		$n = 32$		$n = 64$	
S	Instr.	Ratio	Instr.	Ratio	Instr.	Ratio
0	175.3	2.432	1279.5	1.734	11900.9	1.227
256	87.5	3.492	742.0	2.173	6173.3	1.938
1024	39.6	3.805	278.9	3.178	1946.2	2.977
4096	37.3	10.619	249.9	10.033	1688.4	12.764
16384	50.6	50.628	370.1	46.182	2469.9	64.857
32768	77.9	89.537	631.1	73.658	4178.0	104.056

Coppersmith inputs		
Bits	n	Instr. ratio
400	5	1.698
450	9	1.803
480	20	1.535

F Auxiliary proofs and nearest-plane precision

F.1 Norm bounds for Seysen reduction

Proof of Lemma 4. At a split, put $\mathbf{X} = \mathbf{T}_{\text{left}}^{-1}\mathbf{T}_{\text{cross}}$. Then

$$\mathbf{T}_{\text{cross}} = \mathbf{T}_{\text{left}}\mathbf{X}, \qquad (\mathbf{T}^{-1})_{\text{cross}} = -\mathbf{X}\mathbf{T}_{\text{right}}^{-1}.$$

For either class of matrices in the statement, let $K(n)$ bound the maximum-entry norms of $\mathbf{T}$ and $\mathbf{T}^{-1}$ through order n. The block identities give

$$K(n) \le \max\{1, \eta\lceil n/2\rceil\}K(\lceil n/2\rceil).$$

The leaf bound is $K(1) = 1$ for unit-diagonal $\mathbf{T}$ and $K(1) \le 4$ for $\mathbf{T} = \mathbf{R}\mathbf{E}^{-1}$ by admissibility. Multiplying over the $\lceil \log_2 n \rceil$ levels gives $K(n) \le K(1)S_\eta(n)$. Column summation gives the induced 1-norm bounds for the scaled factor. Moreover,

$$\kappa_{\text{col}}(\mathbf{R}) \le \|\mathbf{T}\|_2\|\mathbf{T}^{-1}\|_2 \le n^2\|\mathbf{T}\|_{\max}\|\mathbf{T}^{-1}\|_{\max},$$

which gives the condition bound. Finally, $(\mathbf{T}_{I,I})^{-1} = (\mathbf{T}^{-1})_{I,I}$ by upper triangularity. Thus restriction preserves the maximum-entry bounds, and each column of the restricted matrices has at most $|I|$ entries. □

F.2 Exponent arithmetic and bit costs

Let N bound the total input and working storage in bits, and take machine words of length $w = \Theta(\log(2 + N))$. Here $\widetilde{O}$ also suppresses polylogarithmic factors in

N. An input exponent can have $\Theta(w)$ bits, hence magnitude $2^{\Theta(w)}$. For a nonzero integer x in ordinary multiprecision format, $e = \mathrm{bitlength}(|x|) - 1 < N$, so the exponent bound is automatic. An entry record stores its sign, bit length, and a pointer to its binary digits.

An integer input serialized as a flat binary string of length L_{in} can be converted to entry records in $\widetilde{O}(L_{\mathrm{in}})$ bit operations before the in-memory computation. Expanding the dyadic representation of $\mathbf{U}$ into ordinary binary integers costs its expanded length.

Accuracy parameters $\delta = 2^{-k_\delta}$ and $\varepsilon_{\mathrm{out}} = 2^{-k_{\mathrm{out}}}$ are specified by binary integers $k_\delta, k_{\mathrm{out}} \geq 2$. Comparison tolerances ϑ are positive rationals with $O(1 + \log(1/\vartheta))$-bit representations.

Proof of Lemma 6. Let $E_{\mathrm{in}} = 1 + \max_{B_{ij} \neq 0} |e_{ij}| \leq 2^{O(w)}$, where e_{ij} are the input exponents. The initial QR tests (B.4) and column bounds (B.1), followed by restoring the column scales, give $|\log_2 R^{\mathrm{init}}_{jj}| = O(E_{\mathrm{in}} + p_{\mathrm{init}} + \log n)$. Profile monotonicity along the reference factors (3.16) and admissibility keep later scale exponents in the same range. The grid spacing and significand length bound a nonzero return entry's exponent from below, and its norm bounds it from above. Scale and return exponents have binary length $O(w + \log n + \log(2 + p_{\max})) = O(w)$, since the working space includes the matrices and their significands. These $O(1)$-word exponents support addition, subtraction, and comparison in constant word time.

Preparing an operand discards entries below $2^{-p-O(1)}$ and rounds the rest to p significant bits, giving the error in (C.3). Grid decoding compares exponents before shifting at most $O(p)$ significand bits. Diagonal log-ratios use exact exponent differences and directed logarithms of normalized significands to p fractional bits. Each scalar operation thus costs $\widetilde{O}(p)$.

The prepared operands have $O(p)$-bit fixed-point representations. Theorem 7 and the balanced GridSeysen recursion therefore give $\widetilde{O}((n')^\omega p)$ for the matrix operations. Grid decoding, operand preparation, lfspread tests, and updates to scales and counters add $O((n')^2)$ scalar operations. □

F.3 Precision for nearest plane

For the basis $\mathbf{C} = \mathbf{QR}$ of Theorem 2, Babai's algorithm rounds $a_i = (y_i - \sum_{j>i} R_{ij} z_j)/R_{ii}$ to obtain z_i, in descending order. Here $\mathbf{y} = \mathbf{Q}^T\mathbf{t}$ for the target $\mathbf{t}$, and $\mathbf{Cz}$ is the returned lattice vector. Fix $0 < \tau \leq 1$ and put $\eta = \tau/(2(2+\tau))$. Let $d = \min_i R_{ii}$ and let L bound the target and solution size by

$$2^L \geq 2 + \frac{\|\mathbf{t}\|_2 + \|\mathbf{Cz}\|_2}{d}, \qquad p = L + O(n \log(2n) + \log(1/\tau)).$$

With p-bit operands and arithmetic in the triangular stage, each a_i can be evaluated to absolute error at most η. For $\mathbf{D} = \mathrm{diag}(R_{ii})$, both $\mathbf{RD}^{-1}$ and its inverse have spectral norm at most $\kappa_{\mathrm{col}}(\mathbf{R}) \leq 2^{O(n \log n)}$ by the output condition bound. Thus $\|\mathbf{Dz}\|_2 \leq 2^{O(n \log n)} \|\mathbf{Cz}\|_2$, and Higham's substitution error bound [11, Section 8.1] gives $|\widehat{a}_i - a_i| \leq 2^{-p+L+O(n \log n)}$. Choose $\varepsilon_{\mathrm{out}} \leq 2^{-p}$ for the basis. For

the projected target, require

$$\|\widetilde{\mathbf{y}} - \mathbf{y}\|_2 \le 2^{-p}\|\mathbf{t}\|_2.$$

This contributes at most 2^{-p+L} to the error in each a_i.

Rounding $\widehat{a}_i$ gives residuals of magnitude at most $(1/2+\eta)R_{ii}$. At the last coordinate j differing from a closest lattice vector, that vector's residual has magnitude at least $(1/2-\eta)R_{jj}$. These residual bounds give approximation factor

$$\frac{1/2+\eta}{1/2-\eta}\sqrt{n}\ \mathrm{fspread}(\mathbf{R}) = (1+\tau)\sqrt{n}\ \mathrm{fspread}(\mathbf{R}).$$

Either rounding choice is allowed near a half-integer boundary. In exact arithmetic, the two residual bounds are $R_{ii}/2$ and $R_{jj}/2$, giving approximation factor

$$\sqrt{n}\ \mathrm{fspread}(\mathbf{R}) \le \sqrt{n}\, 2^{(a_{\mathrm{LLL}}+\delta)(n-1)}.$$