Recent progress in linear algebra and lattice basis reduction (invited)

Gilles Villard

To cite this version:

Gilles Villard. Recent progress in linear algebra and lattice basis reduction (invited). ISSAC’11 - International symposium on Symbolic and algebraic computation, 2011, San Jose, United States. pp.3-4, 2011, ACM proceedings. <10.1145/1993886.1993889>. <hal-00644796>

HAL Id: hal-00644796
https://hal.archives-ouvertes.fr/hal-00644796
Submitted on 25 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Recent Progress in Linear Algebra and Lattice Basis Reduction

Gilles Villard
CNRS, ENS de Lyon, INRIA, UCBL, Université de Lyon
Laboratoire LIP
gilles.villard@ens-lyon.fr

Categories and Subject Descriptors
I.1.2 [Symbolic and Algebraic Manipulation]: Algorithms; F.2.1 [Analysis of Algorithms and Problem Complexity]: Numerical Algorithms and Problems

General Terms
Algorithms

ABSTRACT
A general goal concerning fundamental linear algebra problems is to reduce the complexity estimates to essentially the same as that of multiplying two matrices (plus possibly a cost related to the input and output sizes). Among the bottlenecks one usually finds the questions of designing a recursive approach and mastering the sizes of the intermediate quantities. A lattice $\Lambda$ of $\mathbb{Z}^n$ is the set of the integer combinations of a basis $b_1, \ldots, b_d$ in $\mathbb{Z}^n$. We consider the relaxed notion of reduction introduced by Lenstra, Lenstra and Lovász in [14]. The lengths of the vectors of a $\text{LLL reduced}$ basis are not minimal but fairly small. Among other properties, the lengths and the orthogonality defect satisfy

$$\Delta(b_1, \ldots, b_d) = \prod_j \frac{\|b_j\|/\det(\Lambda)}{\det(\Lambda)^{1/d}} \leq 2^{O(d^2)}.$$  

The problem of finding a reduced basis of a lattice given by an arbitrary basis is called basis reduction. We focus on the above particular cases for non singular matrices. A more general setting would be reduction for discrete subgroups of $\mathbb{R}^n$ and free modules of finite rank. A rich literature and the wide spectrum of applications of basis reduction show the importance of the domain. For the polynomial case we may refer to Kailath [10] and system theory references therein. About LLL and stronger reductions we may refer to Lovász [15] and the contributions in [16].

Basis reduction. Two matrices $A$ and $B$ whose columns form a basis of a given lattice are equivalent, i.e., $B = AU$ with $U$ unimodular. A typical approach for computing a reduced $B$ from a non reduced $A$ is to apply successive transformations. Over $\mathbb{K}[x]$, the transformations correspond to dependencies in coefficient matrices, and decrease the column degrees. In the integer case, geometrical informations are obtained from orthogonalizations over $\mathbb{R}$ for decreasing the column norms. The basic transformations consist in reducing vectors or matrices against others, and vice versa. Reduction algorithms are seen as generalizations of Euclid’s gcd or continued fraction algorithm (see [7] and seminal references therein). The intermediate computed bases play the role of “remainders”, and the successive basic transformations performed on the bases play the role of “quotients”.

Lehmer’s & Knuth-Schönhage algorithms. Lehmer’s modification of Euclid’s algorithm [13], and Knuth’s [12] and Schönhage’s [19] algorithms, have been a crucial progress. Their idea is to employ the fact that for small quotients,
truncated remainders suffice to compute the quotient sequence. Via an algorithm that multiplies two integers of size $\beta$ in $\mu(\beta)$ operations, this has led to the bit complexity estimate $O(\mu(\beta) \log \beta)$ for the gcd problem. Through some analogies between recent algorithms over $K[x]$ and $\mathbb{Z}$, we will see how similar recursive approaches may be developed for reduction.

**Reduced form over $K[x]$.** Beckermann and Labahn [1] has given a key generalization of the Knuth-Schönhage algorithm for Padé approximants. As a consequence one may show [9] that reconstructing a univariate and proper matrix fraction $CB^{-1}$ from its expansion, with $B$ column reduced of degree $O(\delta)$, can be done in $O(n^2 \delta)$ field operations. This may be applied to computing a reduced form $B$ of a non singular matrix $A$, by reconstructing a fraction $CB^{-1}$ from an appropriate (proper) segment of the expansion of $A^{-1}$ [4]. We will see that it follows that the reduction of a non singular $A \in K[x]^{n \times n}$ of degree $\delta$ can be performed within about the same number $O(n^2 \delta)$ of operations as that of multiplying two matrices of degree $\delta$ in $K[x]^{n \times n}$. The corresponding algorithm of Giorgi et al. [4] is randomized. A deterministic algorithm is given in [5] where the problem of reducing $A$ is reduced to the problem of reducing the Hermite form $H$ of $A$. The latter problem may be solved in $O(n^2 \delta)$ via a partial linearization of $H$, and using the algorithm designed in [4] for fraction reconstruction via approximants.

These reduction approaches work in two phases. A first phase transforms the problem into a “simpler”—with a small degree solution—matrix approximant problem (either via a fraction or the Hermite normal form). The second phase inherits the approximation method of [1], and works by recurrence on the approximation order. (The process may be decomposed into successive matrix factorizations over $K$.)

**LLL reduction over $\mathbb{Z}$.** For an insight into LLL reduction algorithms we may refer to the contributions of Nguyen (Ch. 2), Schnorr (Ch. 4), and Stehlé (Ch. 5) in [16]. We focus on the cost with respect to the integer bit size $\beta$, with the aim of obtaining a bit complexity estimate $O(\mu(\text{Poly}(n)\beta))$.

We will present the gradual strategy of [8, 17] for designing a Lehmer-like algorithm in the following special case (see also [18]). Assume that $B_0$ is LLL reduced, and let $\sigma = \text{diag}(\ell, 1, \ldots, 1)$ where $\ell \geq 1$. We call lift-reduction of $B_0$ the problem of reducing $\sigma B_0, k \in \mathbb{N}$. The lift-reduction of [17] is a recurrence on the order $k$ of the lifting, and implements the lift-reduction as $k$ successive elementary steps. $(B_{i+1}$ is a reduced basis of $\sigma B_i$.) With this setting, the Knuth-Schönhage algorithm may be generalized for the task of lift-reducing [17]. The truncation process of the successive bases ("remainders") relies on an LLL reduction definition that resists perturbation, and takes advantage of the numerical quality of the reduced bases that are fairly well conditioned (see [3]). The multi-dimensionality of lattice reduction leads to the manipulation of significantly differing magnitudes in the transformations themselves. The problem may be solved by truncating also the transformations. (Unlike the integer gcd case where the quotients are not truncated.)

We will see how to use lift-reduction for the LLL reduction of $A$ in time quasi-linear in $\beta$ [17] (non singular case). Lift-reduction is specialized to reducing a lift/shift of an already reduced basis. For example, appropriate reduced bases for calling the lift-reduction can be created iteratively from the Hermite form of $A$. The above approach takes a matrix point of view. An alternative approach to LLL reduction in time $O(\mu(\text{Poly}(n)\beta))$ has been recently obtained [6], by using the 2-dimensional Knuth-Schönhage algorithm from [20, 23].

**Acknowledgment.** We are grateful to Damien Stehlé for his help during the preparation of this talk.

**References**


