AN ALGORITHMIC PROOF
OF THE QUILLEN–SUSLIN THEOREM

By

Bernd Sturmfels

IMA Preprint Series # 409
April 1988
AN ALGORITHMIC PROOF OF THE QUILLEN-SUSLIN THEOREM

BERND STURMFELD*$

Abstract. We give a new constructive proof of the Quillen-Suslin theorem by explicitly computing a basis for an arbitrary stably free $\mathbb{C}[x_1,\ldots,x_n]$-module of finite rank. The resulting algorithm which completes a rectangular matrix to a square invertible matrix whenever possible can be implemented using Buchberger’s Gröbner bases method.

1. Introduction.

One of the most prominent theorems in commutative algebra, stating “Projective modules over polynomial rings are free” and widely known as Serre’s conjecture, has been proved independently by D. Quillen and A. Suslin in 1976. We refer to the expositions of Lam [8], Kunz [7] and Eagon [5] for an introduction to Serre’s conjecture, its history, several inconstructive proofs and further references. An interesting connection to electrical engineering and partially constructive solutions are found in Youla & Pickel [11] and Zak & Lee [12]. For simplicity we restrict ourselves to the field $\mathbb{C}$ of complex numbers.

It was known since 1958 that projective modules over $R := \mathbb{C}[x_1,\ldots,x_n]$ are stably free, i.e., up to isomorphism every finitely generated projective $R$-module is the kernel of an $R$-module epimorphism $A : R^m \to R^l$. In that situation the maximal minors of the corresponding matrix $A$ generate the unit ideal in $R$, and $A$ is said to be unimodular. This reduces the freeness of projective modules to the following elementary statement.

**Theorem 1.** [Quillen-Suslin] Let $A$ be a unimodular $l \times m$-matrix ($l \leq m$) over $\mathbb{C}[x_1,\ldots,x_n]$. Then there exists a unimodular $m \times m$-matrix $U$ over $\mathbb{C}[x_1,\ldots,x_n]$ such that

$$A \cdot U = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \end{pmatrix}.$$

It is the objective of this note to give an elementary algorithm which generates the matrix $U$, and to show how the Quillen-Suslin theorem ties in with some recent developments in computational commutative algebra. Note that the last $m - l$ columns of $U$ form a basis of the free module $\text{kernel}(A) \subset R^m$. Moreover, $A$ equals the first $l$ rows of $U^{-1}$, and hence proving Theorem 1 is equivalent to completing $A$ to a square unimodular or invertible matrix.

*$Institute for Mathematics and its Applications, University of Minnesota, Vincent Hall 514, 206 Church Street S.E., Minneapolis, MN 55455, U.S.A.

Address during the academic year 1988/89: Research Institut for Symbolic Computation, Johannes-Kepler-Universität Linz, 4040 Linz, AUSTRIA
To begin with, let us recall the well-known reduction to the Unimodular Row Problem ($l = 1$). Suppose $l \geq 2$ and the construction promised by Theorem 1 is possible for unimodular matrices with at most $l - 1$ rows. Given a unimodular $l \times m$-matrix $A = (a_{ij})$, we can then find a unimodular $m \times m$-matrix $U'$ with $(a_{11}, a_{12}, \ldots, a_{1m}) \cdot U' = (1, 0, \ldots, 0)$. Since $U'$ is invertible, also the matrix $A' := (a'_{ij}) := A \cdot U'$ is unimodular. By the choice of $U'$, the cofactor of $a'_{11} = 1$ in $A'$ is a unimodular $(l-1) \times (m-1)$-matrix. Induction on $l$ implies the equivalence of Theorem 1 to

**Theorem 2.** [Unimodular row property] Let $f = (f_1, \ldots, f_m) \in \mathbb{C}[x_1, \ldots, x_n]^m$ be a unimodular row. Then there exists a unimodular $m \times m$-matrix $U$ over $\mathbb{C}[x_1, \ldots, x_n]$ such that $f \cdot U = (1, 0, \ldots, 0)$.

**2. The algorithm.**

Our constructive proof of Theorem 2 proceeds by induction on the number of variables, and it consists of two main parts. After some easy preliminaries, we enter a "local loop" which generates solutions for finitely many suitable local rings. In the crucial second phase we pass from the local to the global by "patching together" the previously computed local solutions. In Section 3 we briefly discuss some complexity aspects, in particular, improvements for the special case $m \geq n + 1$.

Each step in the algorithm which is marked with an upper index "GB" can be programmed easily using B. Buchberger's Gröbner bases method which is available as a subroutine in many computer algebra systems (e.g. MACSYMA, MAPLE, SCRATCHPAD). See [3],[4] for details and further references to Gröbner bases theory. In particular, the following commutative algebra problems have well-known computational solutions via Gröbner bases:

- Given $f_1, \ldots, f_k, h \in \mathbb{C}[x_1, \ldots, x_n]$, is $h$ is in the ideal $I$ generated by the $f_i$?
- If so, construct polynomials $g_1, \ldots, g_k$ such that $h = g_1 f_1 + \ldots + g_k f_k$ [10].
- If $1 \not\in I$, find a zero of $I$ in $\mathbb{C}^n$.
- Compute resultant ideals, elimination ideals, and Noether normalizations [2, p.44].

**Algorithmic proof of Theorem 2.** If $m = 2$, we compute $^{GB} h_1, h_2 \in \mathbb{C}[x_1, \ldots, x_n]$ such that $h_1 f_1 + h_2 f_2 = 1$, and we set $U := \begin{pmatrix} h_1 & -f_2 \\ h_2 & f_1 \end{pmatrix}$. If $n = 1$, then the desired matrix $U$ is a product of elementary matrices obtained from the Euclidean algorithm for $\mathbb{C}[x_1]$. Therefore $n \geq 2$ and $m \geq 3$ may be assumed.

Using Noether normalization $^{GB}$ we can change variables and permute the $f_i$'s in order to have $f_1(x_1, \ldots, x_{n-1}, t)$ monic in $t = x_n$. Abbreviate $R := \mathbb{C}[x]$ where $x = (x_1, \ldots, x_{n-1})$, and let $k := 0$.

At this point we enter the local loop, and we set $k := k + 1$. Find $^{GB}$ a common zero $a_k \in \mathbb{C}^{n-1}$ of the polynomials $r_1, r_2, \ldots, r_{k-1}$, and let $M_k := \{ g \in R \mid g(a_k) = 0 \}$ be
the corresponding maximal ideal. (In the first iteration $a_1 \in \mathbb{C}^{n-1}$ is arbitrary.)

Abbreviate $\tilde{f}_i(t) := f_i(a_k, t)$ for $i = 1, \ldots, m$. Since $f(a_k, t) = (\tilde{f}_1, \tilde{f}_2, \ldots, \tilde{f}_m)$ is a unimodular row over $\mathbb{C}[t]$, we have

\begin{equation}
< p > + < \tilde{f}_1 > = \mathbb{C}[t].
\end{equation}

where $p$ generates the principal ideal $< \tilde{f}_2, \ldots, \tilde{f}_m >$ in $\mathbb{C}[t]$. Using the Euclidean algorithm, we find a unimodular $(m-1) \times (m-1)$-matrix $E(t)$ over $\mathbb{C}[t]$ such that

\begin{equation}
(\tilde{f}_2(t), \tilde{f}_3(t), \ldots, \tilde{f}_m(t)) \cdot E(t) = (p(t), 0, \ldots, 0).
\end{equation}

It follows from the definition of $\tilde{f}_i$ that $f_i(x, t) - \tilde{f}_i(t) \in \mathcal{M}_k[t]$. Hence (2) implies

\begin{equation}
f(x, t) \cdot \begin{pmatrix}
1 & 0 \\
0 & E(t)
\end{pmatrix} = (f_1(x, t), p(t) + q_2(x, t), q_3(x, t), \ldots, q_m(x, t))
\end{equation}

where $q_2, \ldots, q_m$ are elements of $\mathcal{M}_k[t]$. In particular we have $q_2(a_k, t) = 0$.

Next compute $^GB$ the resultant $r_k(x)$ of the two polynomials $f_1$ and $p + q_2$ with respect to the variable $t$, and find $^GB$ $v, w \in R[t]$ such that

\begin{equation}
v(x, t)f_1(x, t) + w(x, t)[p(t) + q_2(x, t)] = r_k(x).
\end{equation}

Since $f_1$ is monic in $t$, the resultant has the property that $r_k(x_0) = 0$ if and only if there exists $t_0 \in \mathbb{C}$ with $f_1(x_0, t_0) = p(t_0) + q_2(x_0, t_0) = 0$ [6, Satz I.3.1]. Using (1) this implies $r_k(a_k) \neq 0$. Hence $r_k$ is invertible in the corresponding local ring $R_{\mathcal{M}_k}$, and the matrix

\begin{equation}
U_k(x, t) := \begin{pmatrix}
1 & 0 \\
0 & E(t)
\end{pmatrix} \begin{pmatrix}
v r_k^{-1} & -p - q_2 \\
w r_k^{-1} & f_1
\end{pmatrix} \begin{pmatrix}
1 & 0 & -q_3 & \ldots & -q_m \\
0 & 1 \\
0 & 1 \\
\vdots & \ddots & \ddots & \ddots & \ddots
\end{pmatrix}
\end{equation}

is unimodular over $R_{\mathcal{M}_k}[t]$. By (3),(4) and (5) we have

\begin{equation}
f(x, t) \cdot U_k(x, t) = (1, 0, 0 \ldots, 0).
\end{equation}

Next decide $^GB$ whether the ideal $< r_1, \ldots, r_k >$ is (already) equal to the whole ring $R$. If yes, then we exit the local loop; if not, we return to its beginning.

Note that the termination criterion "$< r_1, \ldots, r_k > = R$" will be satisfied after a finite number of iterations because $r_k \notin < r_1, \ldots, r_{k-1} >$ in each step, and $R$ is noetherian by Hilbert's basis theorem.
Also the powers \( \{r_1^m, \ldots r_k^m\} \) generate the unit ideal in \( R \), and we can find \( GB \) elements \( g_1, \ldots, g_k \in R \) such that
\[
g_1 r_1^m + g_2 r_2^m + \ldots + g_k r_k^m = 1 \quad \text{in} \quad R.
\]

Next we introduce two new variables \( s \) and \( z \), and we define matrices
\[
\Delta_i(s, z) := U_i(s) \cdot U_i^{-1}(s + z) \quad \text{for} \quad i = 1, \ldots, k.
\]
\( \Delta_i(s, z) \) is a unimodular matrix over \( R_M[s, z] \), and \( r_i^m \) is a common denominator for \( \Delta_i(s, z) \). To see this, recall that \( r_i \in R \) is a common denominator for \( U_i(s) \) and \( U_i(s + z) \). The inverse of the latter matrix equals its adjoint (up to a scalar), and thus \( r_i^{m-1} \) is common denominator for \( U_i^{-1}(s + z) \).

We expand \( \Delta_i(s, z) \) as a polynomial in \( z \) with matrix coefficients over \( R_M[s] \):
\[
\Delta_i(s, z) = \Delta_{i0}(s) + \Delta_{i1}(s)z + \Delta_{i2}(s)z^2 + \ldots + \Delta_{id_i}(s)z^{d_i}.
\]
It follows directly from (8) that \( \Delta_{i0}(s) = \Delta_i(s, 0) \) equals the identity matrix \( Id_m \). Replacing \( z \) by \( z r_i^m \) we get
\[
\Delta_i(s, z r_i^m) = Id_m + r_i^m \Delta_{i1}(s)z + r_i^{2m} \Delta_{i2}(s)z^2 + \ldots + r_i^{d_i m} \Delta_{i d_i} z^{d_i}.
\]
Since \( r_i^m \) is a common denominator for \( \Delta_i(s, z) \), it is a common denominator for all summands in the expansion (9). Hence all summands on the right hand side of (10) are denominator-free, and \( \Delta_i(s, z r_i^m) \) is a unimodular matrix over the polynomial ring \( R[s, z] \). Observe furthermore that
\[
f(s) \cdot \Delta_i(s, z r_i^m) = f(s + z r_i^m) \quad \text{in} \quad R[s, z]
\]
by (6). Finally, define
\[
U(t) := \Delta_1(t, -tg_1 r_1^m) \cdot \Delta_2(t - tg_1 r_1^m, -tg_2 r_2^m) \cdot \Delta_3(t - tg_1 r_1^m - tg_2 r_2^m - tg_3 r_3^m) \cdot \ldots \cdot \Delta_{k-1}(t - \sum_{i=1}^{k-2} tg_i r_i^m, -tg_{k-1} r_{k-1}^m) \cdot \Delta_k(t - \sum_{i=1}^{k-1} tg_i r_i^m, -tg_k r_k^m).
\]
The \( k \) factor matrices in (12) are obtained from \( \Delta_i(s, z r_i^m) \) by polynomial specializations \( R[s, z] \to R[t] \), and consequently \( U(t) \) is unimodular over \( R[t] \). By repeated application of (11) and by (7) we get
\[
f(t) \cdot U(t) = f(t - \sum_{i=1}^k tg_i r_i^m) = f(0).\]

The row \( f(0) \in R^m \) is unimodular in \( n - 1 \) variables. By induction on the number of variables this completes the algorithm and the proof of Theorems 1 and 2. \( \square \)
3. Some remarks.

Since complexity upper bounds are known for all individual steps in our algorithm (see [1] for recent developments), the crucial unknown for an overall complexity analysis is the number \( k \) of iterations in the local loop. In all examples we computed so far, the construction could be done in \( k = n - 1 \) steps after a suitable change of variables. So far we were unable to decide whether this works in general or at least in "almost all" cases.

This question is closely related to the study of regular sequences. A sequence \( f_1, \ldots, f_m \) in \( R := \mathbb{C}[x_1, x_2, \ldots, x_n] \) is a regular sequence if \( f_{i+1} \) is neither zero nor a zero-divisor in \( R/I_i \) where \( I_i := \langle f_1, \ldots, f_i \rangle \). The length of the longest regular sequence contained in \( I_m \) is denoted \( ht(I_m) \). If \( ht(I_m) > i \) and if \( f_1, \ldots, f_i \) is regular, then \( f_1, \ldots, f_i, f_{i+1} := \sum_{j=i+1}^{n} \lambda_{ij} f_j \) is a regular sequence for sufficiently generic \( \lambda_{ij} \in \mathbb{C} \) [9, Lemma 2.2]. Hence we have a straightforward probabilistic algorithm to generate a unimodular matrix \( E \) over \( R \) such that the first \( ht(I) + 1 \) polynomials in \( f' := f \cdot E =: (f'_1, \ldots, f'_m) \) form a regular sequence. A deterministic procedure for determining suitable coefficients \( \lambda_{ij} \) can be implemented using a (Gröbner bases) primary decomposition algorithm as suggested in [4, Section 9].

This leads to a substantial simplification of our algorithm in Section 2 whenever the unimodular row \( f \) is of length \( m > n + 1 \): Already the initial segment \( f'_1, \ldots, f'_{n+1} \) of the transformed row \( f' \) generates the unit ideal \( I = R \). Finding \( g_1, \ldots, g_{n+1} \in R \) with \( g_1 f'_1 + \cdots + g_{n+1} f'_{n+1} = 1 \) suffices to obtain the desired unimodular transformation for \( f' \):

\[
U = \begin{pmatrix}
1 & (1-f'_m)g_1 \\
1 & \ddots & \ddots \\
& \ddots & \ddots & \ddots \\
& & \ddots & \ddots & 1 \\
& & & \ddots & -f'_1 & -f'_{m-1}
\end{pmatrix}
\]

(14)

Note that this reduces the important special case of two variables (compare [12]) to rows of length 3.

More generally, let us close with a discussion of the case \( n = m + 1 \). The following technique has proved to be very useful for practical computations. As before, let \( f \in \mathbb{C}[x, t]^{n+1} \) where \( x = (x_1, \ldots, x_{n-1}) \) and \( f_1 \) is monic in \( t = x_n \). We precompute the resultants \( r_i(x) \) of \( f_1 \) and \( f_i \) with respect to the variable \( t \) \((i = 2, \ldots, n)\), and we apply the steps (4) and (5) in the local loop to obtain solutions with denominators \( r_i \) (i.e. \( n - 1 \) solutions over \( \mathbb{C}[x][r_i(t)] \)). In many situations a generic linear change of variables suffices to have \( r_2, \ldots, r_{n+1} \) generating the unit ideal in \( \mathbb{C}[x] \). Then the patching procedure in steps (7)–(12) can be applied, and, using (14), the problem is solved. Again, it would be nice to know whether this preprocessing gives a solution in all or "almost all"
cases. If not, how can we achieve a minimum value for the natural objective function \( h_t(<r_2, \ldots, r_{n+1}>)? \) A deeper geometric analysis of the corresponding Hilbert scheme might answer this question and lead to further algorithmic improvements.

I am indebted to Jack Eagon for introducing me to the unimodular row problem and to Neil White for many helpful discussions on this subject.

REFERENCES