LessGenerators - Finding a minimal generating set for a module (part of the homalg project)

Vinay Wagh

IIT Guwahati

25$^{\text{th}}$ August 2014

Question

Let k be a commutative ring with unity and $A = k[\mathbf{X}] = k[X_1, \ldots, X_n]$ be a polynomial ring over k. Let M be a finitely presented module over A.

Then the problem is to find a minimal generating set for M.

Question

Let k be a commutative ring with unity and $A = k[\mathbf{X}] = k[X_1, \ldots, X_n]$ be a polynomial ring over k. Let M be a finitely presented module over A.

Then the problem is to find a minimal generating set for M.

Question

Let k be a commutative ring with unity and $A = k[\mathbf{X}] = k[X_1, \dots, X_n]$ be a polynomial ring over k. Let M be a finitely presented module over A.
Then the problem is to find a minimal generating set for M.

In 1955 J.-P. Serre asked the question whether any projective module over a polynomial ring $k[X_1, \ldots, X_n]$ in several variables over a field is free, which is known as *Serre's conjecture* [Ser 55] .

The conjecture was proved (independently) by D. Quillen and A. Suslin [Qui 76, Sus 76].

Theorem (Serre's Conjecture – Quillen-Suslin Theorem [Qui 76, Sus 76])

If k is a field then every projective module over a polynomial ring $k[X_1, \ldots, X_n]$ is free.

In geometric language, Serre's Problem simply translates to: Is every vector bundle over $\mathbb{A}^n_k$ trivial?

In 1955 J.-P. Serre asked the question whether any projective module over a polynomial ring $k[X_1, \ldots, X_n]$ in several variables over a field is free, which is known as *Serre's conjecture* [Ser 55] .

The conjecture was proved (independently) by D. Quillen and A. Suslin [Qui 76, Sus 76].

Theorem (Serre's Conjecture – Quillen-Suslin Theorem [Qui 76, Sus 76])

If k is a field then every projective module over a polynomial ring $k[X_1, \ldots, X_n]$ is free.

In geometric language, Serre's Problem simply translates to: Is every vector bundle over $\mathbb{A}_k^n$ trivial?

In 1955 J.-P. Serre asked the question whether any projective module over a polynomial ring $k[X_1, \ldots, X_n]$ in several variables over a field is free, which is known as *Serre's conjecture* [Ser 55] .
The conjecture was proved (independently) by D. Quillen and A. Suslin [Qui 76, Sus 76].

Theorem (Serre's Conjecture – Quillen-Suslin Theorem [Qui 76, Sus 76])

If k is a field then every projective module over a polynomial ring $k[X_1, \ldots, X_n]$ is free.

In geometric language, Serre's Problem simply translates to: Is every vector bundle over $\mathbb{A}^n_k$ trivial?

In 1955 J.-P. Serre asked the question whether any projective module over a polynomial ring $k[X_1, \ldots, X_n]$ in several variables over a field is free, which is known as *Serre's conjecture* [Ser 55] .
The conjecture was proved (independently) by D. Quillen and A. Suslin [Qui 76, Sus 76].

Theorem (Serre's Conjecture – Quillen-Suslin Theorem [Qui 76, Sus 76])

If k is a field then every projective module over a polynomial ring $k[X_1, \ldots, X_n]$ is free.

In geometric language, Serre's Problem simply translates to: Is every vector bundle over $\mathbb{A}^n_k$ trivial?

Here is an alternate version in a slightly general situation:

Theorem (Serre's Conjecture – Alternate formulation)

Let k be a principal ideal domain and $A = k[X_1, \ldots, X_n]$ a polynomial ring with coefficients in k. Let R be a right invertible matrix of size $p \times q$. Then, there exists a unimodular matrix $U \in GL_p(A)$ satisfying:

$$RU = (I_q \quad 0).$$

Thus, the problem gets reduced to completion of unimodular row to an invertible matrix.

Here is an alternate version in a slightly general situation:

Theorem (Serre's Conjecture – Alternate formulation)

Let k be a principal ideal domain and $A = k[X_1, \ldots, X_n]$ a polynomial ring with coefficients in k. Let R be a right invertible matrix of size $p \times q$. Then, there exists a unimodular matrix $U \in GL_p(A)$ satisfying:

$$RU = (I_q \quad 0).$$

Thus, the problem gets reduced to completion of unimodular row to an invertible matrix.

Here is an alternate version in a slightly general situation:

Theorem (Serre's Conjecture – Alternate formulation)

Let k be a principal ideal domain and $A = k[X_1, \ldots, X_n]$ a polynomial ring with coefficients in k. Let R be a right invertible matrix of size $p \times q$. Then, there exists a unimodular matrix $U \in GL_p(A)$ satisfying:

$$RU = (I_q \quad 0).$$

Thus, the problem gets reduced to completion of unimodular row to an invertible matrix.

The following result by Quillen plays a crucial role in proving the Quillen-Suslin theorem.

Theorem (Suslin's Lemma[Rot 08])

Let B *be a commutative ring, let* $s \geq 1$, *and consider polynomials in* $B[y]$*:*

$$\begin{aligned} f(x) &= y^s + a_1 y^{s-1} + \ldots + a_s \\ g(x) &= b_1 y^{s-1} + \ldots + b_s \end{aligned}$$

Then, for each j *with* $1 \leq j \leq s-1$, *the ideal* $(f, g) \subset B[y]$ *contains a polynomial of degree atmost* $s-1$ *having leading coefficient* b_j.

Using this result, Suslin gave a proof of the following result by Horrocks:

The following result by Quillen plays a crucial role in proving the Quillen-Suslin theorem.

Theorem (Suslin's Lemma[Rot 08])

Let B be a commutative ring, let $s \geq 1$, and consider polynomials in $B[y]$:

$$f(x) = y^s + a_1 y^{s-1} + \ldots + a_s$$
$$g(x) = b_1 y^{s-1} + \ldots + b_s$$

Then, for each j with $1 \leq j \leq s-1$, the ideal $(f, g) \subset B[y]$ contains a polynomial of degree atmost $s-1$ having leading coefficient b_j.

Using this result, Suslin gave a proof of the following result by Horrocks:

Theorem (Horrocks)

Let $R = B[y]$, where B is a local ring, and let $\alpha = (a_1, \dots, a_n) \in R^n$ be a unimodular column. If some a_i is monic, then α is the first column of some invertible matrix in $GL(n, R)$.

The following result was proved by Vaserstein:

Theorem ([Rot 08])

Let B be a domain, let $R = B[y]$, and let $\alpha(y)$ be a unimodular column at least one of whose coordinates is monic, say, $\alpha(y) = \alpha_1(y), \ldots, \alpha_n(y)$. Then

$$\alpha(y) = M(y) \cdot \beta$$

where $M(y) \in GL_n(R)$ and β is a unimodular column over B.

Using these results, one can prove Serre's conjecture inductively, using induction on the number of variables. i.e. reducing one variable at every step.

Theorem (Quillen-Suslin)

If k is a field, then every finitely generated projective $k[x_1, \ldots, x_m]$-module is free.

In 1992, Logar and Sturmfels, gave algorithmic proof of the Quillen-Suslin theorem.

This algorithm uses induction on the number of variables n, and it consists of two main parts:

Local Loop: which generates solutions for finitely many suitable local rings

Patching: in which all these "local" solutions are patched together to get a global solution.

In 1992, Logar and Sturmfels, gave algorithmic proof of the Quillen-Suslin theorem.
This algorithm uses induction on the number of variables n, and it consists of two main parts:

Local Loop: which generates solutions for finitely many suitable local rings

Patching: in which all these "local" solutions are patched together to get a global solution.

After proving Serre's conjecture, Suslin in 1977, proved the following K_1-analogue of Serre's conjecture:

Theorem (Suslin's stability theorem[Sus 77])

Let R be a commutative Noetherian ring. Let $n \geq \max(3, \dim(R) + 2)$. Let $A = (f_{ij})$ be an $n \times n$-matrix of determinant 1 with entries in the polynomial ring $R[x_1 \ldots, x_m]$. Then A can be written as a product of elementary matrices over $R[x_1 \ldots, x_m]$.

In other words,

$$SL_n(R[x_1 \ldots, x_m]) = E_n(R[x_1 \ldots, x_m]),$$

for $n \geq \max(3, \dim(R) + 2)$ where $E_n(R[x_1 \ldots, x_m])$ denotes the subgroup of elementary matrices over $R[x_1 \ldots, x_m]$.

After proving Serre's conjecture, Suslin in 1977, proved the following K_1-analogue of Serre's conjecture:

Theorem (Suslin's stability theorem[Sus 77])

Let R be a commutative Noetherian ring. Let $n \geq \max(3, \dim(R) + 2)$. Let $A = (f_{ij})$ be an $n \times n$-matrix of determinant 1 with entries in the polynomial ring $R[x_1 \ldots, x_m]$. Then A can be written as a product of elementary matrices over $R[x_1 \ldots, x_m]$.

In other words,

$$SL_n(R[x_1 \ldots, x_m]) = E_n(R[x_1 \ldots, x_m]),$$

for $n \geq \max(3, \dim(R) + 2)$ where $E_n(R[x_1 \ldots, x_m])$ denotes the subgroup of elementary matrices over $R[x_1 \ldots, x_m]$.

In 1995, H. Park and C. Woodburn[PW 95] gave an algorithmic proof of this equality. i.e. given any matrix $A \in SL_n(R[x_1 \dots, x_m])$, the algorithm produces a sequence $E_1, \dots, E_k$ of matrices in $E_n(R[x_1 \dots, x_m])$ such that $A = E_1 \cdot E_2 \cdot \dots \cdot E_k$.

During 2004–2009, Anna Fabiańska implemented the Logar-Sturmfels algorithm in a computer algebra system **MAPLE**. The implementation is through packages called `QuillenSuslin` and `involutive` [Fab].

The main advantage of this implementation is that the result (Quillen-Suslin algorithm) can be applied to the unimodular matrices over a polynomial ring whose coefficients ring can be a finite field, number field or ring of integers.

However it cannot be applied to the field of complex numbers as mentioned in Logar-Sturmfels algorithm.

During 2004–2009, Anna Fabiańska implemented the Logar-Sturmfels algorithm in a computer algebra system **MAPLE**. The implementation is through packages called `QuillenSuslin` and `involutive` [Fab].

The main advantage of this implementation is that the result (Quillen-Suslin algorithm) can be applied to the unimodular matrices over a polynomial ring whose coefficients ring can be a finite field, number field or ring of integers.

However it cannot be applied to the field of complex numbers as mentioned in Logar-Sturmfels algorithm.

During 2004–2009, Anna Fabiaǹska implemented the Logar-Sturmfels algorithm in a computer algebra system **MAPLE**. The implementation is through packages called `QuillenSuslin` and `involutive` [Fab].
The main advantage of this implementation is that the result (Quillen-Suslin algorithm) can be applied to the unimodular matrices over a polynomial ring whose coefficients ring can be a finite field, number field or ring of integers.
However it cannot be applied to the field of complex numbers as mentioned in Logar-Sturmfels algorithm.

In 2012, I, along with Dr. Mohamed Barakat, started a project called LessGenerators, to implement the Quillen-Suslin algorithm using computer algebra systems SINGULAR and GAP.

- The package is based on the homalg project. The aim of the package LessGenerators is to provide a tool for finding a minimal generating set for a given module.
- The package provides a partial support for the localization of the baserings at prime ideals. e.g. $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$
- Using this, we implement the Suslin Lemma, theorem of Horrocks and patching of local solutions as mentioned in Logar-Sturmfels algorithm for all computable fields of char 0 in this package.
- The structure of the package is in sync with the base concept of the homalg project and provides universal implementation in the sense of CASs. i.e. it can use any CAS supported by the homalg project for ring arithmetic.

- The package is based on the homalg project. The aim of the package LessGenerators is to provide a tool for finding a minimal generating set for a given module.
- The package provides a partial support for the localization of the baserings at prime ideals. e.g. $k[X_1, \dots, X_{n-1}]_{\wp}[X_n]$
- Using this, we implement the Suslin Lemma, theorem of Horrocks and patching of local solutions as mentioned in Logar-Sturmfels algorithm for all computable fields of char 0 in this package.
- The structure of the package is in sync with the base concept of the homalg project and provides universal implementation in the sense of CASs. i.e. it can use any CAS supported by the homalg project for ring arithmetic.

- The package is based on the `homalg` project. The aim of the package `LessGenerators` is to provide a tool for finding a minimal generating set for a given module.
- The package provides a partial support for the localization of the baserings at prime ideals. e.g. $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$
- Using this, we implement the Suslin Lemma, theorem of Horrocks and patching of local solutions as mentioned in Logar-Sturmfels algorithm for all computable fields of char 0 in this package.
- The structure of the package is in sync with the base concept of the `homalg` project and provides universal implementation in the sense of CASs. i.e. it can use any CAS supported by the `homalg` project for ring arithmetic.

- The package is based on the homalg project. The aim of the package LessGenerators is to provide a tool for finding a minimal generating set for a given module.
- The package provides a partial support for the localization of the baserings at prime ideals. e.g. $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$
- Using this, we implement the Suslin Lemma, theorem of Horrocks and patching of local solutions as mentioned in Logar-Sturmfels algorithm for all computable fields of char 0 in this package.
- The structure of the package is in sync with the base concept of the homalg project and provides universal implementation in the sense of CASs. i.e. it can use any CAS supported by the homalg project for ring arithmetic.

The package depends mainly upon the following packages from homalg project:

- Modules
- homalg
- RingsForHomalg
- LocalizeRingForHomalg
- MatricesForHomalg
- GAPDoc

- Using the package LocalizeRingForHomalg, one can compute the localization of a polynomial ring by a maximal ideal.
- The Logar-Strumfels algorithm uses local rings, which are obtained using localization at prime ideals. The functionality of localization at prime ideals is partially added to LocalizeRingForHomalg. Through this, one can use the polynomial ring over local ring $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$.

- Using the package `LocalizeRingForHomalg`, one can compute the localization of a polynomial ring by a maximal ideal.
- The Logar-Strumfels algorithm uses local rings, which are obtained using localization at prime ideals. The functionality of localization at prime ideals is partially added to `LocalizeRingForHomalg`. Through this, one can use the polynomial ring over local ring $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$.

- Using the package LocalizeRingForHomalg, one can compute the localization of a polynomial ring by a maximal ideal.
- The Logar-Strumfels algorithm uses local rings, which are obtained using localization at prime ideals. The functionality of localization at prime ideals is partially added to LocalizeRingForHomalg. Through this, one can use the polynomial ring over local ring $k[X_1, \ldots, X_{n-1}]_{\wp}[X_n]$.

gap> Q := HomalgFieldOfRationalsInSingular( );;

gap> R := (Q * "x") * "y";

Singular output supressed...

Q[x][y]

gap> m := HomalgMatrix("[

$2*x^2+2*x*y+y^2+1$,$x*y+y^2+x$,$x+y$, $x*y+y^2+x$,

y^2+1, y]", 2, 3, R);

<A 2 x 3 matrix over an external ring>

gap> M := LeftPresentation(m);

<A non-torsion left module presented by 2 relations for 3 generators>

gap> IsStablyFree(M);

true

gap> M;

<A free left module of rank 1 on 3 non-free generators satisfying 3 relations>

gap> Q := HomalgFieldOfRationalsInSingular( );;
gap> R := (Q * "x") * "y";
Singular output supressed...

Q[x][y]

gap> m := HomalgMatrix("[
$2*x^2+2*x*y+y^2+1$,$x*y+y^2+x$,$x+y$, $x*y+y^2+x$,
y^2+1, y]", 2, 3, R);

<A 2 x 3 matrix over an external ring>

gap> M := LeftPresentation(m);

<A non-torsion left module presented by 2 relations for 3 generators>

gap> IsStablyFree(M);

true

gap> M;

<A free left module of rank 1 on 3 non-free generators satisfying 3 relations>

gap> Q := HomalgFieldOfRationalsInSingular( );;

gap> R := (Q * "x") * "y";

Singular output supressed...

Q[x][y]

gap> m := HomalgMatrix("[$2*x^2+2*x*y+y^2+1$,$x*y+y^2+x$,$x+y$, $x*y+y^2+x$, y^2+1, y]", 2, 3, R);

<A 2 x 3 matrix over an external ring>

gap> M := LeftPresentation(m);

<A non-torsion left module presented by 2 relations for 3 generators>

gap> IsStablyFree(M);

true

gap> M;

<A free left module of rank 1 on 3 non-free generators satisfying 3 relations>

gap> Q := HomalgFieldOfRationalsInSingular( );;

gap> R := (Q * "x") * "y";

Singular output supressed...

Q[x][y]

gap> m := HomalgMatrix("[

$2*x^2+2*x*y+y^2+1$,$x*y+y^2+x$,$x+y$, $x*y+y^2+x$,

y^2+1, y]", 2, 3, R);

<A 2 x 3 matrix over an external ring>

gap> M := LeftPresentation(m);

<A non-torsion left module presented by 2 relations for 3 generators>

gap> IsStablyFree(M);

true

gap> M;

<A free left module of rank 1 on 3 non-free generators satisfying 3 relations>

gap> Q := HomalgFieldOfRationalsInSingular( );;

gap> R := (Q * "x") * "y";

Singular output supressed...

Q[x][y]

gap> m := HomalgMatrix("[$2 * x^2 + 2 * x * y + y^2 + 1$,$x * y + y^2 + x$,$x + y$, $x * y + y^2 + x$, $y^2 + 1$, y]", 2, 3, R);

<A 2 x 3 matrix over an external ring>

gap> M := LeftPresentation(m);

<A non-torsion left module presented by 2 relations for 3 generators>

gap> IsStablyFree(M);

true

gap> M;

<A free left module of rank 1 on 3 non-free generators satisfying 3 relations>

```
gap> Q := HomalgFieldOfRationalsInSingular( );;
gap> R := ( Q * "x" ) * "y";
Singular output supressed...
Q[x][y]
gap> m := HomalgMatrix( "[
```

$2*x^2+2*x*y+y^2+1$,$x*y+y^2+x$,$x+y$, $x*y+y^2+x$,
y^2+1, y]", 2, 3, R);

```
<A 2 x 3 matrix over an external ring>
gap> M := LeftPresentation( m );
<A non-torsion left module presented by 2 relations for 3
generators>
gap> IsStablyFree( M );
true
gap> M;
<A free left module of rank 1 on 3 non-free generators satisfying 3
relations>
```

```
gap> LoadPackage( "LessGenerators" );

true

gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y'
$\ldots$ Singular output supressed$\ldots$
Q[x,y]
gap> m := HomalgMatrix( "[ \
> 2*x^2+2*x*y+y^2+1,x*y+y^2+x,x+y,\
> x*y+y^2+x,          y^2+1,    y   \
> ]", 2, 3, R );
<A 2 x 3 matrix over an external ring>
gap> M := LeftPresentation( m );
<A non-torsion left module presented by 2 relations fo
gap>
```

- A. Fabiańska, QUILLEN SUSLIN project: A package for computing bases of free modules over polynomial rings, `http://wwwb.math.rwth-aachen.de/QuillenSuslin`, 2003-2009.
- A. Fabiańska *Algorithmic analysis of presentations of groups and modules*, Ph.D. Thesis, RWTH Aachen, 2009.
- A. Fabiańska, A. Quadrat, *Applications of the Quillen-Suslin theorem to multidimensional systems theory*, Radon Series Comp. Appl. Math 3, (2007), pp. 23–106. (Available also as INRIA report 6126, February 2007, from `http://hal.inria.fr/inria-00131035`).
- M. Barakat, *The homalg Project* – GAP4, 2007–2013, `http://homalg.math.rwth-aachen.de/index.php`.

- B. Barwick and B. Stone, *The QuillenSuslin Package for Macaulay2*, 2013. (Available at http://arxiv.org/abs/1107.4383v2).
- The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.6.5*, 2013, http://www.gap-system.org.
- T. Y. Lam, *Serre's Problem on Projective Modules*, Springer Monographs in Mathematics, Springer-Verlag, Berlin, 2006.
- R. C. Laubenbacher, C. J. Woodburn. *An algorithm for the Quillen-Suslin theorem for monoid rings*, Algorithms for algebra (Eindhoven, 1996). J. Pure Appl. Algebra 117/118 (1997), pp. 395–429.

- R. C. Laubenbacher, C. J. Woodburn. *A new algorithm for the Quillen-Suslin theorem*, Beiträge Algebra Geom. 41 (2000), no. 1, pp. 23–31.
- A. Logar, B. Sturmfels, *Algorithms for the Quillen-Suslin theorem*, J. Algebra, 145 (1992), pp. 231–239.
- H. Park, C. Woodburn, *An algorithmic proof of Suslin's stability theorem for polynomial rings*, J. Algebra, 178 (1995), pp. 227–298.
- D. Quillen, *Projective modules over polynomial rings*, Invent. Math., 36 (1976), pp. 167–171.
- J. Rotman, *An Introduction to Homological Algebra*, 2nd Ed., Springer, (2008).

- J.-P. Serre, *Faisceaux algébriques cohérents*, Annals of Mathematics. Second Series. 61(2)(1955) pp. 197–278,
- A. A. Suslin, *Projective modules over polynomial rings are free*, Dokl. Akad. Nauk. S.S.S.R., 229 (1976), (Soviet Math. Dokl., 17 (1976), pp. 1160–1164).
- A. A. Suslin, *On the structure of the special linear group over polynomial rings*. Math. USSR Izv. 11, (1977), pp. 221–238.