GradedModules

A homalg based package for the Abelian category of finitely presented graded modules over computable graded rings

2024.01-01

10 January 2024

Mohamed Barakat

Sebastian Gutsche

Sebastian Jambor

Markus Lange-Hegermann

Arne Lorenz

Oleksandr Motsak

Mohamed Barakat
Email: mohamed.barakat@uni-siegen.de
Homepage: https://mohamed-barakat.github.io
Address: Walter-Flex-Str. 3
57072 Siegen
Germany

Sebastian Gutsche
Email: gutsche@mathematik.uni-siegen.de
Homepage: https://sebasguts.github.io
Address: Department Mathematik
Universität Siegen
Walter-Flex-Straße 3
57072 Siegen
Germany

Sebastian Jambor
Email: sebastian.jambor@rwth-aachen.de
Homepage: http://wwwb.math.rwth-aachen.de/~sebastian/
Address: Sebastian Jambor
Lehrstuhl B fuer Mathematik, RWTH Aachen
Templergraben 64
52062 Aachen
Germany

Markus Lange-Hegermann
Email: markus.lange-hegermann@hs-owl.de
Homepage: https://www.th-owl.de/eecs/fachbereich/team/markus-lange-hegermann/
Address: Markus Lange-Hegermann
Hochschule Ostwestfalen-Lippe
Liebigstraße 87
32657 Lemgo
Germany

Arne Lorenz
Email: arne.lorenz@rwth-aachen.de
Homepage: http://wwwb.math.rwth-aachen.de/~arne/
Address: Arne Lorenz
Lehrstuhl B fuer Mathematik, RWTH Aachen
Templergraben 64
52062 Aachen
Germany

Oleksandr Motsak
Email: motsak@mathematik.uni-kl.de
Homepage: http://www.mathematik.uni-kl.de/~motsak/

Address: Department of Mathematics
University of Kaiserslautern
67653 Kaiserslautern
Germany

Copyright

Contents

Chapter 1

Installation of the GradedModules Package

To install this package just extract the package's archive file to the GAP `pkg` directory.

By default the GradedModules package is not automatically loaded by GAP when it is installed. You must load the package with

```
LoadPackage("GradedModules");
```

before its functions become available.

Please, send me an e-mail if you have any questions, remarks, suggestions, etc. concerning this package. Also, I would be pleased to hear about applications of this package.

Mohamed Barakat

Chapter 2

Ring Maps

2.1 Ring Maps: Attributes

2.1.1 KernelSubobject

▷ `KernelSubobject(phi)` (method)

Returns: a homalg submodule

The kernel ideal of the ring map `phi`.

2.2 Ring Maps: Operations and Functions

2.2.1 SegreMap

▷ `SegreMap(R, s)` (method)

Returns: a homalg ring map

The ring map corresponding to the Segre embedding of $MultiProj(R)$ into the projective space according to $P(W_1) \times P(W_2) \to P(W_1 \otimes W_2)$.

2.2.2 PlueckerMap

▷ `PlueckerMap(l, n, A, s)` (method)

Returns: a homalg ring map

The ring map corresponding to the Plücker embedding of the Grassmannian $G_l(P^n(A)) = G_l(P(W))$ into the projective space $P(\bigwedge^l W)$, where $W = V^*$ is the A-dual of the free module $V = A^{n+1}$ of rank $n+1$.

2.2.3 VeroneseMap

▷ `VeroneseMap(n, d, A, s)` (method)

Returns: a homalg ring map

The ring map corresponding to the Veronese embedding of the projective space $P^n(A) = P(W)$ into the projective space $P(S^d W)$, where $W = V^*$ is the A-dual of the free module $V = A^{n+1}$ of rank $n+1$.

Chapter 3

GradedModules

3.1 GradedModules: Category and Representations

3.2 GradedModules: Constructors

3.3 GradedModules: Properties

For more properties see the corresponding section (**Modules: Modules: Properties**)) in the documentation of the homalg package.

3.4 GradedModules: Attributes

3.4.1 BettiTable (for modules)

▷ `BettiTable(M)` (attribute)

Returns: a homalg diagram

The Betti diagram of the homalg graded module M.

3.4.2 CastelnuovoMumfordRegularity

▷ `CastelnuovoMumfordRegularity(M)` (attribute)

Returns: an integer

The Castelnuovo-Mumford regularity of the homalg graded module M.

3.4.3 CastelnuovoMumfordRegularityOfSheafification

▷ `CastelnuovoMumfordRegularityOfSheafification(M)` (attribute)

Returns: an integer

The Castelnuovo-Mumford regularity of the sheafification of homalg graded module M.

For more attributes see the corresponding section (**Modules: Modules: Attributes**)) in the documentation of the homalg package.

3.5 LISHV: Logical Implications for GradedModules

3.6 GradedModules: Operations and Functions

3.6.1 MonomialMap

▷ `MonomialMap(d, M)` (operation)

Returns: a homalg map

The map from a free graded module onto all degree d monomial generators of the finitely generated homalg module M.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> m := MonomialMap( 1, M );
<A homomorphism of graded left modules>
gap> Display( m );
x^2,0,0,
x*y,0,0,
x*z,0,0,
y^2,0,0,
y*z,0,0,
z^2,0,0,
0,  x,0,
0,  y,0,
0,  z,0,
0,  0,1

the graded map is currently represented by the above 10 x 3 matrix

(degrees of generators of target: [ -1, 0, 1 ])
```

3.6.2 RandomMatrix

▷ `RandomMatrix(S, T)` (operation)

Returns: a homalg matrix

A random matrix between the graded source module S and the graded target module T.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "a,b,c";;
gap> S := GradedRing( R );;
gap> rand := RandomMatrix( S^1 + S^2, S^2 + S^3 + S^4 );
<A 2 x 3 matrix over a graded ring>
gap> #Display( rand );
gap> #-3*a-b,                                                    -1,
gap> #-a^2+a*b+2*b^2-2*a*c+2*b*c+c^2,                            -a+c,
gap> #-2*a^3+5*a^2*b-3*b^3+3*a*b*c+3*b^2*c+2*a*c^2+2*b*c^2+c^3,-3*b^2-2*a*c-2*b*c+c^2
```

3.6.3 GeneratorsOfHomogeneousPart

▷ `GeneratorsOfHomogeneousPart(d, M)` (operation)

Returns: a homalg matrix

The resulting homalg matrix consists of a generating set (over R) of the d-th homogeneous part of the finitely generated homalg S-module M, where R is the coefficients ring of the graded ring S with $S_0 = R$.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> m := GeneratorsOfHomogeneousPart( 1, M );
<An unevaluated non-zero 7 x 3 matrix over a graded ring>
gap> Display( m );
x^2,0,0,
x*y,0,0,
y^2,0,0,
0,  x,0,
0,  y,0,
0,  z,0,
0,  0,1
(over a graded ring)
```

Compare with `MonomialMap` (3.6.1).

3.6.4 SubmoduleGeneratedByHomogeneousPart

▷ `SubmoduleGeneratedByHomogeneousPart(d, M)` (operation)

Returns: a homalg module

The submodule of the homalg module M generated by the image of the d-th monomial map (→ `MonomialMap` (3.6.1)), or equivalently, by the generating set of the d-th homogeneous part of M.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> n := SubmoduleGeneratedByHomogeneousPart( 1, M );
<A graded left submodule given by 7 generators>
gap> Display( M );
z,  0,    0,
0,  y^2*z,z^2,
x^3,y^2,  z

Cokernel of the map

Q[x,y,z]^(1x3) --> Q[x,y,z]^(1x3),

currently represented by the above matrix
(graded, degrees of generators: [ -1, 0, 1 ])
```

```
gap> Display( n );
x^2,0,0,
x*y,0,0,
y^2,0,0,
0,  x,0,
0,  y,0,
0,  z,0,
0,  0,1

A left submodule generated by the 7 rows of the above matrix

(graded, degrees of generators: [ 1, 1, 1, 1, 1, 1, 1 ])
gap> N := UnderlyingObject( n );
<A graded left module presented by yet unknown relations for 7 generators>
gap> Display( N );
0, 0, z,0, 0,  0,0,
0, z, 0,0, 0,  0,0,
z, 0, 0,0, 0,  0,0,
0, 0, 0,0, -z, y,0,
0, 0, 0,-z,0,  x,0,
0, 0, 0,-y,x,  0,0,
0, -y,x,0, 0,  0,0,
-y,x, 0,0, 0,  0,0,
x, 0, 0,0, y,  0,z,
0, 0, 0,0, y*z,0,z^2

Cokernel of the map

Q[x,y,z]^(1x10) --> Q[x,y,z]^(1x7),

currently represented by the above matrix

(graded, degrees of generators: [ 1, 1, 1, 1, 1, 1, 1 ])
gap> gens := GeneratorsOfModule( N );
<A set of 7 generators of a homalg left module>
gap> Display( gens );
x^2,0,0,
x*y,0,0,
y^2,0,0,
0,  x,0,
0,  y,0,
0,  z,0,
0,  0,1

a set of 7 generators given by the rows of the above matrix
```

3.6.5 RepresentationMapOfRingElement

▷ `RepresentationMapOfRingElement(r, M, d)` (operation)

Returns: a homalg matrix

The graded map induced by the homogeneous degree 1 ring element r (of the underlying homalg graded ring S) regarded as a R-linear map between the d-th and the $(d+1)$-st homogeneous part of the graded finitely generated homalg S-module M, where R is the coefficients ring of the graded ring S with $S_0 = R$. The generating set of both modules is given by `GeneratorsOfHomogeneousPart` (3.6.3). The entries of the matrix presenting the map lie in the coefficients ring R.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> x := Indeterminate( S, 1 );
x
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> m := RepresentationMapOfRingElement( x, M, 0 );
<A "homomorphism" of graded left modules>
gap> Display( m );
1,0,0,0,0,0,0,
0,1,0,0,0,0,0,
0,0,0,1,0,0,0

the graded map is currently represented by the above 3 x 7 matrix

(degrees of generators of target: [ 1, 1, 1, 1, 1, 1, 1 ])
```

3.6.6 RepresentationMatrixOfKoszulId

▷ `RepresentationMatrixOfKoszulId(d, M)` (operation)

Returns: a homalg matrix

It is assumed that all indeterminates of the underlying homalg graded ring S are of degree 1. The output is the homalg matrix of the multiplication map $\mathrm{Hom}(A, M_d) \to \mathrm{Hom}(A, M_{d+1})$, where A is the Koszul dual ring of S, defined using the operation `KoszulDualRing`.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> A := KoszulDualRing( S, "a,b,c" );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> m := RepresentationMatrixOfKoszulId( 0, M );
<An unevaluated 3 x 7 matrix over a graded ring>
gap> Display( m );
a,b,0,0,0,0,0,
0,a,b,0,0,0,0,
0,0,0,a,b,c,0
(over a graded ring)
```

3.6.7 RepresentationMapOfKoszulId

▷ `RepresentationMapOfKoszulId(d, M)` (operation)

Returns: a homalg map

It is assumed that all indeterminates of the underlying homalg graded ring S are of degree 1. The output is the the multiplication map $\mathrm{Hom}(A, M_d) \to \mathrm{Hom}(A, M_{d+1})$, where A is the Koszul dual ring of S, defined using the operation KoszulDualRing.

Example
```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> A := KoszulDualRing( S, "a,b,c" );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ] );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> m := RepresentationMapOfKoszulId( 0, M );
<A homomorphism of graded left modules>
gap> Display( m );
a,b,0,0,0,0,0,
0,a,b,0,0,0,0,
0,0,0,a,b,c,0

the graded map is currently represented by the above 3 x 7 matrix

(degrees of generators of target: [ 4, 4, 4, 4, 4, 4, 4 ])
```

3.6.8 KoszulRightAdjoint

▷ KoszulRightAdjoint(*M, degree_lowest, degree_highest*) (operation)

Returns: a homalg cocomplex

It is assumed that all indeterminates of the underlying homalg graded ring S are of degree 1. Compute the homalg A-cocomplex C of Koszul maps of the homalg S-module M (→ RepresentationMapOfKoszulId (3.6.7)) in the $[$ *degree_lowest* .. *degree_highest* $]$. The Castelnuovo-Mumford regularity of M is characterized as the highest degree d, such that C is not exact at d. A is the Koszul dual ring of S, defined using the operation KoszulDualRing.

Example
```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> A := KoszulDualRing( S, "a,b,c" );;
gap> M := HomalgMatrix( "[ x^3, y^2, z,   z, 0, 0 ]", 2, 3, S );;
gap> M := LeftPresentationWithDegrees( M, [ -1, 0, 1 ], S );
<A graded non-torsion left module presented by 2 relations for 3 generators>
gap> CastelnuovoMumfordRegularity( M );
1
gap> R := KoszulRightAdjoint( M, -5, 5 );
<A cocomplex containing 10 morphisms of graded left modules at degrees
[ -5 .. 5 ]>
gap> R := KoszulRightAdjoint( M, 1, 5 );
<An acyclic cocomplex containing
4 morphisms of graded left modules at degrees [ 1 .. 5 ]>
gap> R := KoszulRightAdjoint( M, 0, 5 );
<A cocomplex containing 5 morphisms of graded left modules at degrees
[ 0 .. 5 ]>
gap> R := KoszulRightAdjoint( M, -5, 5 );
<A cocomplex containing 10 morphisms of graded left modules at degrees
[ -5 .. 5 ]>
```

```
gap> H := Cohomology( R );
<A graded cohomology object consisting of 11 graded left modules at degrees
[ -5 .. 5 ]>
gap> ByASmallerPresentation( H );
<A non-zero graded cohomology object consisting of
11 graded left modules at degrees [ -5 .. 5 ]>
gap> Cohomology( R, -2 );
<A graded zero left module>
gap> Cohomology( R, -3 );
<A graded zero left module>
gap> Cohomology( R, -1 );
<A graded cyclic torsion-free non-free left module presented by 2 relations fo\
r a cyclic generator>
gap> Cohomology( R, 0 );
<A graded non-zero cyclic left module presented by 3 relations for a cyclic ge\
nerator>
gap> Cohomology( R, 1 );
<A graded non-zero cyclic left module presented by 2 relations for a cyclic ge\
nerator>
gap> Cohomology( R, 2 );
<A graded zero left module>
gap> Cohomology( R, 3 );
<A graded zero left module>
gap> Cohomology( R, 4 );
<A graded zero left module>
gap> Display( Cohomology( R, -1 ) );
Q{a,b,c}/< b, a >

(graded, degree of generator: 0)
gap> Display( Cohomology( R, 0 ) );
Q{a,b,c}/< c, b, a >

(graded, degree of generator: 0)
gap> Display( Cohomology( R, 1 ) );
Q{a,b,c}/< b, a >

(graded, degree of generator: 2)
```

3.6.9 HomogeneousPartOverCoefficientsRing

▷ HomogeneousPartOverCoefficientsRing(*d*, *M*) (operation)

Returns: a homalg module

The degree d homogeneous part of the graded R-module M as a module over the coefficient ring or field of R.

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> M := HomalgMatrix( "[ x, y^2, z^3 ]", 3, 1, S );;
gap> M := Subobject( M, ( 1 * S )^0 );
<A graded torsion-free (left) ideal given by 3 generators>
gap> CastelnuovoMumfordRegularity( M );
4
```

```
gap> M1 := HomogeneousPartOverCoefficientsRing( 1, M );
<A graded left vector space of dimension 1 on a free generator>
gap> gen1 := GeneratorsOfModule( M1 );
<A set consisting of a single generator of a homalg left module>
gap> Display( M1 );
Q^(1 x 1)

(graded, degree of generator: 1)
gap> M2 := HomogeneousPartOverCoefficientsRing( 2, M );
<A graded left vector space of dimension 4 on free generators>
gap> Display( M2 );
Q^(1 x 4)

(graded, degrees of generators: [ 2, 2, 2, 2 ])
gap> gen2 := GeneratorsOfModule( M2 );
<A set of 4 generators of a homalg left module>
gap> M3 := HomogeneousPartOverCoefficientsRing( 3, M );
<A graded left vector space of dimension 9 on free generators>
gap> Display( M3 );
Q^(1 x 9)

(graded, degrees of generators: [ 3, 3, 3, 3, 3, 3, 3, 3, 3 ])
gap> gen3 := GeneratorsOfModule( M3 );
<A set of 9 generators of a homalg left module>
gap> Display( gen1 );
x

a set consisting of a single generator given by (the row of) the above matrix
gap> Display( gen2 );
x^2,
x*y,
x*z,
y^2

a set of 4 generators given by the rows of the above matrix
gap> Display( gen3 );
x^3,
x^2*y,
x^2*z,
x*y*z,
x*z^2,
x*y^2,
y^3,
y^2*z,
z^3

a set of 9 generators given by the rows of the above matrix
```

Chapter 4

The Tate Resolution

4.1 The Tate Resolution: Operations and Functions

4.1.1 TateResolution

▷ TateResolution(*M, degree_lowest, degree_highest*) (operation)

Returns: a homalg cocomplex

Compute the Tate resolution of the sheaf *M*.

Example
```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x0..x3";;
gap> S := GradedRing( R );;
gap> A := KoszulDualRing( S, "e0..e3" );;
```

In the following we construct the different exterior powers of the cotangent bundle shifted by 1. Observe how a single 1 travels along the diagnoal in the window $[-3..0]x[0..3]$.

First we start with the structure sheaf with its Tate resolution:

Example
```
gap> O := S^0;
<The graded free left module of rank 1 on a free generator>
gap> T := TateResolution( O, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>
gap> betti := BettiTable( T );
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>>
gap> Display( betti );
total:    35  20  10   4   1   1   4  10  20  35  56   ?   ?   ?
----------|---|---|---|---|---|---|---|---|---|---|---|---|---|
    3:    35  20  10   4   1   .   .   .   .   .   .   0   0   0
    2:     *   .   .   .   .   .   .   .   .   .   .   .   0   0
    1:     *   *   .   .   .   .   .   .   .   .   .   .   .   0
    0:     *   *   *   .   .   .   .   .   1   4  10  20  35  56
----------|---|---|---|---|---|---|---|---S---|---|---|---|---|
twist:    -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5
---------------------------------------------------------------
Euler:   -35 -20 -10  -4  -1   0   0   0   1   4  10  20  35  56
```

The Castelnuovo-Mumford regularity of the *underlying module* is distinguished among the list of twists by the character 'V' pointing to it. It is *not* an invariant of the sheaf (see the next diagram).

The residue class field (i.e. S modulo the maximal homogeneous ideal):

Example

```
gap> k := HomalgMatrix( Indeterminates( S ), Length( Indeterminates( S ) ), 1, S );
<A 4 x 1 matrix over a graded ring>
gap> k := LeftPresentationWithDegrees( k );
<A graded cyclic left module presented by 4 relations for a cyclic generator>
```

Another way of constructing the structure sheaf:

Example

```
gap> U0 := SyzygiesObject( 1, k );
<A graded torsion-free left module presented by yet unknown relations for 4 ge\
nerators>
gap> T0 := TateResolution( U0, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>
gap> betti0 := BettiTable( T0 );
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>>
gap> Display( betti0 );
total:   35  20  10   4   1   1   4  10  20  35  56   ?   ?   ?
----------|---|---|---|---|---|---|---|---|---|---|---|---|---|
    3:   35  20  10   4   1   .   .   .   .   .   .   0   0   0
    2:    *   .   .   .   .   .   .   .   .   .   .   .   0   0
    1:    *   *   .   .   .   .   .   .   .   .   .   .   .   0
    0:    *   *   *   .   .   .   .   .   1   4  10  20  35  56
----------|---|---|---|---|---|---|---|---S---|---|---|---|---|
twist:   -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5
---------------------------------------------------------------
Euler:  -35 -20 -10  -4  -1   0   0   0   1   4  10  20  35  56
```

The cotangent bundle:

Example

```
gap> cotangent := SyzygiesObject( 2, k );
<A graded torsion-free left module presented by yet unknown relations for 6 ge\
nerators>
gap> IsFree( UnderlyingModule( cotangent ) );
false
gap> Rank( cotangent );
3
gap> cotangent;
<A graded reflexive non-projective rank 3 left module presented by 4 relations\
 for 6 generators>
gap> ProjectiveDimension( UnderlyingModule( cotangent ) );
2
```

the cotangent bundle shifted by 1 with its Tate resolution:

Example

```
gap> U1 := cotangent * S^1;
<A graded non-torsion left module presented by 4 relations for 6 generators>
```

```
gap> T1 := TateResolution( U1, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>
gap> betti1 := BettiTable( T1 );
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>>
gap> Display( betti1 );
total:   120   70   36   15    4    1    6   20   45   84  140    ?    ?    ?
-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|
     3:  120   70   36   15    4    .    .    .    .    .    .    0    0    0
     2:    *    .    .    .    .    .    .    .    .    .    .    .    0    0
     1:    *    *    .    .    .    .    .    1    .    .    .    .    .    0
     0:    *    *    *    .    .    .    .    .    .    6   20   45   84  140
-----------|----|----|----|----|----|----|----|----|----S----|----|----|----|
twist:    -8   -7   -6   -5   -4   -3   -2   -1    0    1    2    3    4    5
-----------------------------------------------------------------------------
Euler:  -120  -70  -36  -15   -4    0    0   -1    0    6   20   45   84  140
```

The second power U^2 of the shifted cotangent bundle $U = U^1$ and its Tate resolution:

Example

```
gap> U2 := SyzygiesObject( 3, k ) * S^2;
<A graded rank 3 left module presented by 1 relation for 4 generators>
gap> T2 := TateResolution( U2, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>
gap> betti2 := BettiTable( T2 );
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>>
gap> Display( betti2 );
total:   140   84   45   20    6    1    4   15   36   70  120    ?    ?    ?
-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|
     3:  140   84   45   20    6    .    .    .    .    .    .    0    0    0
     2:    *    .    .    .    .    .    1    .    .    .    .    .    0    0
     1:    *    *    .    .    .    .    .    .    .    .    .    .    .    0
     0:    *    *    *    .    .    .    .    .    .    4   15   36   70  120
-----------|----|----|----|----|----|----|----|----|----S----|----|----|----|
twist:    -8   -7   -6   -5   -4   -3   -2   -1    0    1    2    3    4    5
-----------------------------------------------------------------------------
Euler:  -140  -84  -45  -20   -6    0    1    0    0    4   15   36   70  120
```

The third power U^3 of the shifted cotangent bundle $U = U^1$ and its Tate resolution:

Example

```
gap> U3 := SyzygiesObject( 4, k ) * S^3;
<A graded free left module of rank 1 on a free generator>
gap> Display( U3 );
Q[x0,x1,x2,x3]^(1 x 1)

(graded, degree of generator: 1)
gap> T3 := TateResolution( U3, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>
gap> betti3 := BettiTable( T3 );
```

```
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded left modules at degrees [ -5 .. 5 ]>>
gap> Display( betti3 );
total:    56  35  20  10   4   1   1   4  10  20  35   ?   ?   ?
----------|---|---|---|---|---|---|---|---|---|---|---|---|---|
    3:    56  35  20  10   4   1   .   .   .   .   .   0   0   0
    2:     *   .   .   .   .   .   .   .   .   .   .   .   0   0
    1:     *   *   .   .   .   .   .   .   .   .   .   .   .   0
    0:     *   *   *   .   .   .   .   .   .   1   4  10  20  35
----------|---|---|---|---|---|---|---|---|---S---|---|---|---|
twist:    -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5
----------------------------------------------------------------
Euler:   -56 -35 -20 -10  -4  -1   0   0   0   1   4  10  20  35
```

Another way to construct $U^2 = U^(3-1)$:

Example

```
gap> u2 := GradedHom( U1, S^(-1) );
<A graded torsion-free right module on 4 generators satisfying yet unknown rel\
ations>
gap> t2 := TateResolution( u2, -5, 5 );
<An acyclic cocomplex containing
10 morphisms of graded right modules at degrees [ -5 .. 5 ]>
gap> BettiTable( t2 );
<A Betti diagram of <An acyclic cocomplex containing
10 morphisms of graded right modules at degrees [ -5 .. 5 ]>>
gap> Display( last );
total:    140   84   45   20    6    1    4   15   36   70  120    ?    ?    ?
-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|
    3:    140   84   45   20    6    .    .    .    .    .    .    0    0    0
    2:      *    .    .    .    .    .    1    .    .    .    .    .    0    0
    1:      *    *    .    .    .    .    .    .    .    .    .    .    .    0
    0:      *    *    *    .    .    .    .    .    .    4   15   36   70  120
-----------|----|----|----|----|----|----|----|----|----S----|----|----|----|
twist:     -8   -7   -6   -5   -4   -3   -2   -1    0    1    2    3    4    5
------------------------------------------------------------------------------
Euler:   -140  -84  -45  -20   -6    0    1    0    0    4   15   36   70  120
```

Chapter 5

Examples

5.1 Betti Diagrams

5.1.1 DE-2.2

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x0,x1,x2";;
gap> S := GradedRing( R );;
gap> mat := HomalgMatrix( "[ x0^2, x1^2, x2^2 ]", 1, 3, S );
<A 1 x 3 matrix over a graded ring>
gap> M := RightPresentationWithDegrees( mat, S );
<A graded cyclic right module on a cyclic generator satisfying 3 relations>
gap> M := RightPresentationWithDegrees( mat );
<A graded cyclic right module on a cyclic generator satisfying 3 relations>
gap> d := Resolution( M );
<A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>
gap> betti := BettiTable( d );
<A Betti diagram of <A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>>
gap> Display( betti );
 total:  1 3 3 1
----------------
     0:  1 . . .
     1:  . 3 . .
     2:  . . 3 .
     3:  . . . 1
----------------
degree:  0 1 2 3
gap> ## we are still below the Castelnuovo-Mumford regularity, which is 3:
gap> M2 := SubmoduleGeneratedByHomogeneousPart( 2, M );
<A graded torsion right submodule given by 3 generators>
gap> d2 := Resolution( M2 );
<A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>
gap> betti2 := BettiTable( d2 );
<A Betti diagram of <A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>>
gap> Display( betti2 );
```

```
 total:  3 8 6 1
----------------
     2:  3 8 6 .
     3:  . . . 1
----------------
degree:  0 1 2 3
```

5.1.2 DE-Code

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x0,x1,x2";;
gap> S := GradedRing( R );;
gap> mat := HomalgMatrix( "[ x0^2, x1^2 ]", 1, 2, S );
<A 1 x 2 matrix over a graded ring>
gap> M := RightPresentationWithDegrees( mat, S );
<A graded cyclic right module on a cyclic generator satisfying 2 relations>
gap> d := Resolution( M );
<A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>
gap> betti := BettiTable( d );
<A Betti diagram of <A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>>
gap> Display( betti );
 total:  1 2 1
--------------
     0:  1 . .
     1:  . 2 .
     2:  . . 1
--------------
degree:  0 1 2
gap> m := SubmoduleGeneratedByHomogeneousPart( 2, M );
<A graded torsion right submodule given by 4 generators>
gap> d2 := Resolution( m );
<A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>
gap> betti2 := BettiTable( d2 );
<A Betti diagram of <A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>>
gap> Display( betti2 );
     2:  4 8 4
--------------
degree:  0 1 2
```

5.1.3 Schenck-3.2

This is an example from Section 3.2 in [Sch03].

Example

```
gap> Qxyz := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> mmat := HomalgMatrix( "[ x, x^3 + y^3 + z^3 ]", 1, 2, Qxyz );
<A 1 x 2 matrix over an external ring>
gap> S := GradedRing( Qxyz );;
gap> M := RightPresentationWithDegrees( mmat, S );
```

```
<A graded cyclic right module on a cyclic generator satisfying 2 relations>
gap> Mr := Resolution( M );
<A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>
gap> bettiM := BettiTable( Mr );
<A Betti diagram of <A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>>
gap> Display( bettiM );
 total:  1 2 1
--------------
     0:  1 1 .
     1:  . . .
     2:  . 1 1
--------------
degree:  0 1 2
gap> R := GradedRing( CoefficientsRing( S ) * "x,y,z,w" );;
gap> nmat := HomalgMatrix( "[ z^2 - y*w, y*z - x*w, y^2 - x*z ]", 1, 3, R );
<A 1 x 3 matrix over a graded ring>
gap> N := RightPresentationWithDegrees( nmat );
<A graded cyclic right module on a cyclic generator satisfying 3 relations>
gap> Nr := Resolution( N );
<A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>
gap> bettiN := BettiTable( Nr );
<A Betti diagram of <A right acyclic complex containing
2 morphisms of graded right modules at degrees [ 0 .. 2 ]>>
gap> Display( bettiN );
 total:  1 3 2
--------------
     0:  1 . .
     1:  . 3 2
--------------
degree:  0 1 2
```

5.1.4 Schenck-8.3

This is an example from Section 8.3 in [Sch03].

Example
```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z,w";;
gap> S := GradedRing( R );;
gap> jmat := HomalgMatrix( "[ z*w, x*w, y*z, x*y, x^3*z - x*z^3 ]", 1, 5, S );
<A 1 x 5 matrix over a graded ring>
gap> J := RightPresentationWithDegrees( jmat );
<A graded cyclic right module on a cyclic generator satisfying 5 relations>
gap> Jr := Resolution( J );
<A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>
gap> betti := BettiTable( Jr );
<A Betti diagram of <A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>>
gap> Display( betti );
 total:  1 5 6 2
```

```
----------------
     0:  1 . . .
     1:  . 4 4 1
     2:  . . . .
     3:  . 1 2 1
----------------
degree:  0 1 2 3
```

5.1.5 Schenck-8.3.3

This is Exercise 8.3.3 in [Sch03].

Example

```
gap> Qxyz := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( Qxyz );;
gap> mat := HomalgMatrix( "[ x*y*z, x*y^2, x^2*z, x^2*y, x^3 ]", 1, 5, S );
<A 1 x 5 matrix over a graded ring>
gap> M := RightPresentationWithDegrees( mat, S );
<A graded cyclic right module on a cyclic generator satisfying 5 relations>
gap> Mr := Resolution( M );
<A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>
gap> betti := BettiTable( Mr );
<A Betti diagram of <A right acyclic complex containing
3 morphisms of graded right modules at degrees [ 0 .. 3 ]>>
gap> Display( betti );
 total:  1 5 6 2
----------------
     0:  1 . . .
     1:  . . . .
     2:  . 5 6 2
----------------
degree:  0 1 2 3
```

5.2 Commutative Algebra

5.2.1 Saturate

Example

```
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z";;
gap> S := GradedRing( R );;
gap> m := GradedLeftSubmodule( "x,y,z", S );
<A graded torsion-free (left) ideal given by 3 generators>
gap> I := Intersect( m^3, GradedLeftSubmodule( "x", S ) );
<A graded torsion-free (left) ideal given by 6 generators>
gap> NrRelations( I );
8
gap> Im := SubobjectQuotient( I, m );
<A graded torsion-free rank 1 (left) ideal given by 3 generators>
gap> I_m := Saturate( I, m );
<A graded principal (left) ideal of rank 1 on a free generator>
gap> Is := Saturate( I );
```

```
<A graded principal (left) ideal of rank 1 on a free generator>
gap> Assert( 0, Is = I_m );
```

5.3 Global Section Modules of the Induced Sheaves

5.3.1 Examples of the ModuleOfGlobalSections Functor and Purity Filtrations

Example

```
gap> LoadPackage( "GradedRingForHomalg" );;
gap> Qxyzt := HomalgFieldOfRationalsInDefaultCAS( ) * "x,y,z,t";;
gap> S := GradedRing( Qxyzt );;
gap>
gap> wmat := HomalgMatrix( "[ \
> x*y,  y*z,    z*t,        0,           0,          0,\
> x^3*z,x^2*z^2,0,          x*z^2*t,     -z^2*t^2,   0,\
> x^4,  x^3*z,  0,          x^2*z*t,     -x*z*t^2,   0,\
> 0,    0,      x*y,        -y^2,        x^2-t^2,    0,\
> 0,    0,      x^2*z,      -x*y*z,      y*z*t,      0,\
> 0,    0,      x^2*y-x^2*t,-x*y^2+x*y*t,y^2*t-y*t^2,0,\
> 0,    0,      0,          0,           -1,         1 \
> ]", 7, 6, Qxyzt );;
gap>
gap> LoadPackage( "GradedModules" );;
gap> wmor := GradedMap( wmat, "free", "free", "left", S );;
gap> IsMorphism( wmor );;
gap> W := LeftPresentationWithDegrees( wmat, S );;
gap> HW := ModuleOfGlobalSections( W );
<A graded left module presented by yet unknown relations for 6 generators>
gap> LinearStrandOfTateResolution( W, 0,4 );
<A cocomplex containing 4 morphisms of graded left modules at degrees
[ 0 .. 4 ]>
gap> purity_iso := IsomorphismOfFiltration( PurityFiltration( W ) );
<A non-zero isomorphism of graded left modules>
gap> Hpurity_iso := ModuleOfGlobalSections( purity_iso );
<An isomorphism of graded left modules>
gap> ModuleOfGlobalSections( wmor );
<A homomorphism of graded left modules>
gap> NaturalMapToModuleOfGlobalSections( W );
<A homomorphism of graded left modules>
```

5.3.2 Horrocks Mumford bundle

This example computes the global sections module of the Horrocks-Mumford bundle.

Example

```
gap> LoadPackage( "GradedRingForHomalg" );;
gap> R := HomalgFieldOfRationalsInDefaultCAS( ) * "x0..x4";;
gap> S := GradedRing( R );;
gap> A := KoszulDualRing( S, "e0..e4" );;
gap> LoadPackage( "GradedModules" );;
gap> mat := HomalgMatrix( "[ \
> e1*e4, e2*e0, e3*e1, e4*e2, e0*e3, \
> e2*e3, e3*e4, e4*e0, e0*e1, e1*e2  \
```

```
> ]",
> 2, 5, A );
<A 2 x 5 matrix over a graded ring>
gap> phi := GradedMap( mat, "free", "free", "left", A );;
gap> IsMorphism( phi );
true
gap> M := GuessModuleOfGlobalSectionsFromATateMap( 2, phi );
#I  GuessModuleOfGlobalSectionsFromATateMap uses a heuristic for efficiency;
please check the correctness of the following result

<A graded left module presented by yet unknown relations for 19 generators>
gap> IsPure( M );
true
gap> Rank( M );
2
gap> Display( BettiTable( Resolution( M ) ) );
 total:  19 35 20  2
--------------------
     3:   4  .  .  .
     4:  15 35 20  .
     5:   .  .  .  2
--------------------
degree:   0  1  2  3
gap> Display( BettiTable( TateResolution( M, -5, 5 ) ) );
total:  100  37  14  10   5   2   5  10  14  37 100   ?   ?   ?   ?
----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
    4:  100  35   4   .   .   .   .   .   .   .   .   0   0   0   0
    3:    *   .   2  10  10   5   .   .   .   .   .   .   0   0   0
    2:    *   *   .   .   .   .   .   2   .   .   .   .   .   0   0
    1:    *   *   *   .   .   .   .   .   .   5  10  10   2   .   0
    0:    *   *   *   *   .   .   .   .   .   .   .   .   4  35 100
----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---S
twist:   -9  -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5
-------------------------------------------------------------------
Euler:  100  35   2 -10 -10  -5   0   2   0  -5 -10 -10   2  35 100
gap> M;
<A graded reflexive non-projective rank 2 left module presented by 99 \
relations for 19 generators>
gap> P := ElementOfGrothendieckGroup( M );
( 2*O_{P^4} - 1*O_{P^3} - 4*O_{P^2} - 2*O_{P^1} ) -> P^4
gap> P!.DisplayTwistedCoefficients := true;
true
gap> P;
( 2*O(-3) - 10*O(-2) + 15*O(-1) - 5*O(0) ) -> P^4
gap> chi := HilbertPolynomial( M );
1/12*t^4+2/3*t^3-1/12*t^2-17/3*t-5
gap> c := ChernPolynomial( M );
( 2 | 1-h+4*h^2 ) -> P^4
gap> ChernPolynomial( M * S^3 );
( 2 | 1+5*h+10*h^2 ) -> P^4
gap> ch := ChernCharacter( M );
[ 2-u-7*u^2/2!+11*u^3/3!+17*u^4/4! ] -> P^4
```

```
gap> HilbertPolynomial( ch );
1/12*t^4+2/3*t^3-1/12*t^2-17/3*t-5
gap> List( [ -8 .. 7 ], i -> Value( chi, i ) );
[ 35, 2, -10, -10, -5, 0, 2, 0, -5, -10, -10, 2, 35, 100, 210, 380 ]
gap> HF := HilbertFunction( M );
function( t ) ... end
gap> List( [ 0 .. 7 ], HF );
[ 0, 0, 0, 4, 35, 100, 210, 380 ]
gap> IndexOfRegularity( M );
4
gap> DataOfHilbertFunction( M );
[ [ [ 4 ], [ 3 ] ], 1/12*t^4+2/3*t^3-1/12*t^2-17/3*t-5 ]
```

References

[Sch03] Hal Schenck. *Computational algebraic geometry*, volume 58 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 2003. 19, 20, 21

Index