ModulesOverLocalRingsForCAP

Category of modules over a local ring modeled by Serre quotients for CAP

2017.04.11

4 April 2017

Sebastian Posur

Sebastian Posur

Email: sebastian.posur@uni-siegen.de

Homepage: https://sebastianpos.github.io

Address: Department Mathematik
Universität Siegen
Walter-Flex-Straße 3
57068 Siegen
Germany

Contents

Chapter 1

Cap category

1.1 Constructors

1.1.1 CategoryOfLeftModulePresentationsOverLocalRing (for IsHomalgRing, IsList)

▷ `CategoryOfLeftModulePresentationsOverLocalRing(`*`arg1, arg2`*`)` (operation)

Insert documentation for your function here

Chapter 2

Examples and Tests

2.1 Gauss

Example

```
gap> Q := HomalgFieldOfRationalsInSingular();;
gap> R := Q * "x,y,z";;
gap> category := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "x", "y", "z" ] );;
gap> M := AsLeftPresentation( HomalgMatrix( [ [ "(x^2 + y^2 + z^2 - 1)^2" ] ], 1, 1, R ) );;
gap> N := AsSerreQuotientCategoryObject( category, M );;
gap> IsZero( N );
true
gap> M := AsLeftPresentation( HomalgMatrix( [ [ "(x^2 + y^2 + z^2)^2" ] ], 1, 1, R ) );;
gap> N := AsSerreQuotientCategoryObject( category, M );;
gap> IsZero( N );
false
gap> M := AsLeftPresentation( HomalgMatrix( "x-1,0,0,0", 2, 2, R ) );;
gap> N := AsSerreQuotientCategoryObject( category, M );;
gap> m := MinimalGeneratorsModel( N );;
gap> IsIsomorphism(m);
true
gap> MinimalNumberOfGenerators( N );
1
```

Example

```
gap> Q := HomalgFieldOfRationalsInSingular();;
gap> R := Q * "x,y";;
gap> R := R / "x^2 - y";;
gap> SetIsIntegralDomain( R, true );;
gap> matrix := HomalgMatrix( "[x,0,0,0,y,0,0,0,x^2-y]", 3, 3, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[ 1,2 ]
gap> matrix := HomalgMatrix( "[1,1,1,1,1,1,1,1,1]", 3, 3, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[ 1 ]
gap> matrix := HomalgMatrix( "[0,0,0,0,0,0,0,0,0,0,0,0]", 3, 4, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[]
gap> matrix := HomalgZeroMatrix( 0, 4, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[]
```

```
gap> matrix := HomalgZeroMatrix( 4, 0, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[]
gap> matrix := HomalgZeroMatrix( 0, 0, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[]
gap> matrix := HomalgMatrix( "[x,y,x,y,y,x,x,x,x^2-y]", 3, 3, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[ 1, 2, 3 ]
gap> matrix := HomalgIdentityMatrix( 9, R );;
gap> REFIndicesForMatricesOverIntegralDomain( matrix );
[ 1, 2, 3, 4, 5, 6, 7, 8, 9 ]
```

Example

```
gap> F5 := HomalgRingOfIntegersInSingular( 5 );;
gap> R := F5 * "x,y,z,v,w";;
gap> category := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "x", "y", "z", "v", "w" ] )
gap> i1 := HomalgMatrix( "[ x-z, y-w ]", 2, 1, R );;
gap> i2 := HomalgMatrix( "[ y^6*v^2*w-y^3*v*w^20+1, x*y^4*z^4*w-z^5*w^5+x^3*y*z^2-1 ]", 2, 1, R )
gap> M1 := AsLeftPresentation( i1 );;
gap> M2 := AsLeftPresentation( i2 );;
gap> OIO := CokernelObject( Annihilator( DirectSum( M1, M2 ) ) );;
gap> j1 := HomalgMatrix( "[ x*z, x*w, y*z, y*w, v^2 ]", 5, 1, R );;
gap> j2 := HomalgMatrix( "[ y^6*v^2*w-y^3*v*w^2+1, x*y^4*z^4*w-z^5*w^5+x^3*y*z^2-1, x^7 ]", 3, 1,
gap> M1 := AsLeftPresentation( j1 );;
gap> M2 := AsLeftPresentation( j2 );;
gap> OJO := CokernelObject( Annihilator( DirectSum( M1, M2 ) ) );;
gap> M := AsSerreQuotientCategoryObject( category, OIO );;
gap> N := AsSerreQuotientCategoryObject( category, OJO );;
gap> Min := FunctorMinimalModel( category );;
gap> M_min := ApplyFunctor( Min, M );;
gap> N_min := ApplyFunctor( Min, N );;
gap> T := TorComplex( M_min, N_min );;
gap> H0 := HomologyFunctor( category, 0 );;
gap> H1 := HomologyFunctor( category, 1 );;
gap> H2 := HomologyFunctor( category, 2 );;
gap> h0 := ApplyFunctor( H0, T );;
gap> MinimalNumberOfGenerators( h0 );
1
gap> h0 := Source( FiltrationByPrimeIdealEmbedding( h0 ) );;
gap> MinimalNumberOfGenerators( h0 );
3
gap> h0 := Source( FiltrationByPrimeIdealEmbedding( h0 ) );;
gap> MinimalNumberOfGenerators( h0 );
2
gap> h0 := Source( FiltrationByPrimeIdealEmbedding( h0 ) );;
gap> MinimalNumberOfGenerators( h0 );
0
gap> h1 := ApplyFunctor( H1, T );;
gap> MinimalNumberOfGenerators( h1 );
1
gap> h1 := Source( FiltrationByPrimeIdealEmbedding( h1 ) );;
gap> MinimalNumberOfGenerators( h1 );
```

```
1
gap> h1 := Source( FiltrationByPrimeIdealEmbedding( h1 ) );;
gap> MinimalNumberOfGenerators( h1 );
0
gap> h2 := ApplyFunctor( H2, T );;
gap> MinimalNumberOfGenerators( h2 );
0
```

Example

```
gap> R := HomalgRingOfIntegers( 4 );;
gap> category := CategoryOfLeftModulePresentationsOverLocalRing( R, [ 2 ] );;
gap> IsRegular( category );
false
```

Example

```
gap> R := HomalgFieldOfRationalsInSingular() * "x,y";;
gap> R := R/ "x^2 - y^3";;
gap> category1 := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "x", "y" ] );;
gap> IsRegular( category1 );
false
gap> category2 := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "x-1", "y-1" ] );;
gap> IsRegular( category2 );
true
```

This is an example from homalg.

2.2 Intersection theory

Example

```
gap> Q := HomalgFieldOfRationalsInSingular( );;
gap> R := Q * "x,y,z,w";;
gap> category := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "x", "y", "z", "w" ] );;
gap> Y1 := HomalgMatrix( "[ x,y ]", 2, 1, R );;
gap> Y1 := AsLeftPresentation( Y1 );;
gap> Y2 := HomalgMatrix( "[ z,w ]", 2, 1, R );;
gap> Y2 := AsLeftPresentation( Y2 );;
gap> Y := CokernelObject( Annihilator( DirectSum( Y1, Y2 ) ) );;
gap> Y := AsSerreQuotientCategoryObject( category, Y );;
gap> ZZ := HomalgMatrix( "[ x-z, y-w ]", 2, 1, R );;
gap> ZZ := AsLeftPresentation( ZZ );;
gap> ZZ := AsSerreQuotientCategoryObject( category, ZZ );;
gap> T := TorComplex( ZZ, Y );;
gap> H0 := HomologyFunctor( category, 0 );;
gap> H1 := HomologyFunctor( category, 1 );;
gap> H2 := HomologyFunctor( category, 2 );;
gap> Min := FunctorMinimalModel( category );;
gap> h0 := ApplyFunctor( H0, T );;
gap> h0 := ApplyFunctor( Min, h0 );;
gap> MinimalNumberOfGenerators( h0 );
1
gap> ph0 := Source( FiltrationByPrimeIdealEmbedding( h0 ) );;
gap> ph0 := ApplyFunctor( Min, ph0 );;
gap> MinimalNumberOfGenerators( ph0 );
```

```
2
gap> pph0 := Source( FiltrationByPrimeIdealEmbedding( ph0 ) );;
gap> IsZero( pph0 );
true
gap> h1 := ApplyFunctor( H1, T );;
gap> h1 := ApplyFunctor( Min, h1 );;
gap> MinimalNumberOfGenerators( h1 );
1
gap> ph1 := Source( FiltrationByPrimeIdealEmbedding( h1 ) );;
gap> IsZero( ph1 );
true
gap> IsZero( ApplyFunctor( H2, T ) );
true
```

Example

```
gap> Q := HomalgFieldOfRationalsInSingular( );;
gap> R := Q * "x,y,z";;
gap> category := CategoryOfLeftModulePresentationsOverLocalRing( R, [ "y","z" ] );;
gap> A := HomalgMatrix( "[ y ]", 1, 1, R );;
gap> Ap := AsLeftPresentation( A );;
gap> A := AsSerreQuotientCategoryObject( category, Ap );;
gap> B := HomalgMatrix( "[ z ]", 1, 1, R );;
gap> Bp := AsLeftPresentation( B );;
gap> B := AsSerreQuotientCategoryObject( category, Bp );;
gap> T := TorComplex( A, B );;
gap> H0 := HomologyFunctor( category, 0 );;
gap> H1 := HomologyFunctor( category, 1 );;
gap> H2 := HomologyFunctor( category, 2 );;
gap> h0 := ApplyFunctor( H0, T );;
gap> MinimalNumberOfGenerators( h0 );
1
gap> ph0 := Source( FiltrationByPrimeIdealEmbedding( h0 ) );;
gap> MinimalNumberOfGenerators( ph0 );
0
gap> h1 := ApplyFunctor( H1, T );;
gap> MinimalNumberOfGenerators( h1 );
0
```

Index