Computing modular cohomology rings of finite groups

Simon King
Friedrich Schiller University Jena
Joint work with David Green, Graham Ellis, Bettina Eick

July 24, 2019

seit 1558

Software, aim

SageMath package p_group_cohomology

- Documentation:
  `http://users.minet.uni-jena.de/cohomology/documentation`
- Results: `http://users.minet.uni-jena.de/~king/cohomology`

Software, aim

SageMath package p_group_cohomology

- Documentation:
  `http://users.minet.uni-jena.de/cohomology/documentation`
- Results: `http://users.minet.uni-jena.de/~king/cohomology`
- Installation:
  - v3.1: `sage -i p_group_cohomology`
  - v3.2: See `https://trac.sagemath.org/ticket/28204`

Software, aim

SageMath package p_group_cohomology

- Documentation:
  `http://users.minet.uni-jena.de/cohomology/documentation`
- Results: `http://users.minet.uni-jena.de/~king/cohomology`
- Installation:
  - v3.1: `sage -i p_group_cohomology`
  - v3.2: See `https://trac.sagemath.org/ticket/28204`

Aim

Computation of/with modular cohomology rings of finite groups, $H^*(G; \mathbb{F}_p)$, which includes some ring theoretic invariants, induced maps and detection of ring isomorphisms.

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need ~ 8 minutes for order 64

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need $\sim$ 8 minutes for order 64

(J. Carlson needed $\sim$ 8 months comp. time [1997-2001])

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need ~ 8 minutes for order 64

(J. Carlson needed ~ 8 months comp. time [1997-2001])

about 2 months for order 128 (now probably faster).

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need $\sim$ 8 minutes for order 64

(J. Carlson needed $\sim$ 8 months comp. time [1997-2001])

about 2 months for order 128 (now probably faster).

Interesting non prime power groups

Modular cohomology for different primes of (among others)

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need ~ 8 minutes for order 64

(J. Carlson needed ~ 8 months comp. time [1997-2001])

about 2 months for order 128 (now probably faster).

Interesting non prime power groups

Modular cohomology for different primes of (among others)

- Co_3: $H^*(Co_3; \mathbb{F}_2)$ is Cohen-Macaulay (was conjectured by Benson).

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need $\sim$ 8 minutes for order 64

(J. Carlson needed $\sim$ 8 months comp. time [1997-2001])

about 2 months for order 128 (now probably faster).

Interesting non prime power groups

Modular cohomology for different primes of (among others)

- Co_3: $H^*(Co_3; \mathbb{F}_2)$ is Cohen-Macaulay (was conjectured by Benson).
- HS, Janko groups (not J_4), Mathieu groups (not M_{24})
- McL: Correcting result of Adem-Milgram

Results

$H^*(G; \mathbb{F}_2)$ for all 267 groups of order 64 and all 2328 groups of order 128

We need ~ 8 minutes for order 64
(J. Carlson needed ~ 8 months comp. time [1997-2001])
about 2 months for order 128 (now probably faster).

Interesting non prime power groups

Modular cohomology for different primes of (among others)

- Co_3: $H^*(Co_3; \mathbb{F}_2)$ is Cohen-Macaulay (was conjectured by Benson).
- HS, Janko groups (not J_4), Mathieu groups (not M_{24})
- McL: Correcting result of Adem-Milgram
- $Sz(8)$: minimal presentation of $H^*(Sz(8); \mathbb{F}_2)$ has 102 generators of maximal degree 29 and 4790 relations of maximal degree 58.

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Tools we use in SageMath to solve the tasks

1. - D. Green [2001]: "Heady standard bases" (min. proj. resolution of the modular group algebras of prime power groups)

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Tools we use in SageMath to solve the tasks

1.
   - D. Green [2001]: "Heady standard bases" (min. proj. resolution of the modular group algebras of prime power groups)
   - Cartan–Eilenberg [1956]: "Stable element method" (otherwise)

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Tools we use in SageMath to solve the tasks

1.
    - D. Green [2001]: "Heady standard bases" (min. proj. resolution of the modular group algebras of prime power groups)
    - Cartan–Eilenberg [1956]: "Stable element method" (otherwise)
    - SK [2014]: Non-commutative F_5 algorithm hopefully in future

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Tools we use in SageMath to solve the tasks

1.
   - D. Green [2001]: "Heady standard bases" (min. proj. resolution of the modular group algebras of prime power groups)
   - Cartan–Eilenberg [1956]: "Stable element method" (otherwise)
   - SK [2014]: Non-commutative F_5 algorithm hopefully in future
2. Use Cython code, and let Singular compute Gröbner bases.

Computational approaches

General scheme suggested by J. Carlson [2001]

Need to solve the following computational tasks:

1. Given $n \in \mathbb{N}$, compute $H^d(G)$ for all $d \leq n$.
2. Generators/relations $\rightsquigarrow$ Ring approximation $\tau_n H^*(G)$
3. Test if $H^*(G) \cong \tau_n H^*(G)$: Completeness criteria

Tools we use in SageMath to solve the tasks

1. - D. Green [2001]: "Heady standard bases" (min. proj. resolution of the modular group algebras of prime power groups)
   - Cartan–Eilenberg [1956]: "Stable element method" (otherwise)
   - SK [2014]: Non-commutative F_5 algorithm hopefully in future
2. Use Cython code, and let Singular compute Gröbner bases.
3. - D. Benson [2004], D. Green and SK [2011], for prime power groups
   - SK [2013], for non-prime-power groups
   - P. Symonds [2010], for all groups

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \leq U \leq G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$,

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \leq U \leq G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$, determined by stability conditions associated with representatives of $U \setminus G / U$.

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \leq U \leq G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$, determined by stability conditions associated with representatives of $U \setminus G / U$.
- Holt [1985] suggests to use a tower $S = U_0 \leq U_1 \leq ... \leq U_k = G$. Our default: $S \leq N_G(Z(S)) \leq G$.

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \le U \le G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$, determined by stability conditions associated with representatives of $U \setminus G / U$.
- Holt [1985] suggests to use a tower $S = U_0 \le U_1 \le \ldots \le U_k = G$. Our default: $S \le N_G(Z(S)) \le G$.

Mod-2 cohomology of third Conway group [SK, Green, Ellis 2011]

- For $G = Co_3$: $|S| = 1024$ and $|S \setminus G / S| = 484\,680$.

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \leq U \leq G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$, determined by stability conditions associated with representatives of $U \setminus G / U$.
- Holt [1985] suggests to use a tower $S = U_0 \leq U_1 \leq \ldots \leq U_k = G$. Our default: $S \leq N_G(Z(S)) \leq G$.

Mod-2 cohomology of third Conway group [SK, Green, Ellis 2011]

- For $G = Co_3$: $|S| = 1024$ and $|S \setminus G / S| = 484\,680$.
- $S = U_0 \leq U_1 = N_G(\underbrace{Z_2(S)}_{\cong C_4 \times C_2}) \leq U_2 = N_G(C_4) \leq U_3 = N_G(\underbrace{Z(S)}_{\cong C_2}) \leq U_4 = G$

Stable element method of Cartan–Eilenberg

For G not a prime power group and $S \in Syl_p(G)$:

- If $S \leq U \leq G$, then $\mathrm{res}^G_U : H^*(G) \hookrightarrow H^*(U)$, determined by stability conditions associated with representatives of $U \setminus G / U$.
- Holt [1985] suggests to use a tower $S = U_0 \leq U_1 \leq \ldots \leq U_k = G$. Our default: $S \leq N_G(Z(S)) \leq G$.

Mod-2 cohomology of third Conway group [SK, Green, Ellis 2011]

- For $G = Co_3$: $|S| = 1024$ and $|S \setminus G / S| = 484\,680$.
- $S = U_0 \leq U_1 = N_G(\underbrace{Z_2(S)}_{\cong C_4 \times C_2}) \leq U_2 = N_G(C_4) \leq U_3 = N_G(\underbrace{Z(S)}_{\cong C_2}) \leq U_4 = G$
-

i	1	2	3	4
$\lvert U_{i-1} \setminus U_i / U_{i-1} \rvert$	2	3	3	7

In total, only 11 non-trivial stability conditions remain.

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is “large enough” wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$.

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".
  Expl $\mathrm{Syl}_2(Co_3)$: Degrees 8, 12, 14, 15

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".
  Expl $\mathrm{Syl}_2(Co_3)$: Degrees 8, 12, 14, 15 resp. 8, 4, 6, 7

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".
  Expl $\mathrm{Syl}_2(Co_3)$: Degrees 8, 12, 14, 15 resp. 8, 4, 6, 7
- Get smaller last parameter by enumeration $\rightsquigarrow 8, 4, 6, 2$

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".
  Expl $\mathrm{Syl}_2(Co_3)$: Degrees 8, 12, 14, 15 resp. 8, 4, 6, 7
- Get smaller last parameter by enumeration $\rightsquigarrow 8, 4, 6, 2$
- Show that $\exists$ finite field extension $k/\mathbb{F}_2$ so that $H^*(G;\, k)$ has f.r. HSOP in degrees $8, 4, 2, 2$.

Completeness criteria

General scheme

- Find elements of $\tau_n H^*(G)$ guaranteed to be parameters for $H^*(G)$.
- Perform tests on these elements. If they succeed:
- We are done if n is "large enough" wrt. sum of the parameter degrees.

Benson [2004], Green, SK [2011]

- Dickson invariants (maxdeg $\sim p^{\mathrm{rk}_p(G)}$ resp. $\sim p^{\mathrm{rk}_p(G)-\mathrm{rk}(Z(G))}$) yield elements in $\tau_n H^*(G)$. Test if they form a "filter regular HSOP".
  Expl $\mathrm{Syl}_2(Co_3)$: Degrees 8, 12, 14, 15 resp. 8, 4, 6, 7
- Get smaller last parameter by enumeration $\rightsquigarrow 8, 4, 6, 2$
- Show that $\exists$ finite field extension $k/\mathbb{F}_2$ so that $H^*(G; k)$ has f.r. HSOP in degrees $8, 4, 2, 2$.
- Compute filter degree type using parameters of $H^*(G; \mathbb{F}_2)$ but work with parameter degrees of $H^*(G; k)$.

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X \rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X\rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X\rangle\rangle$-module needs to be computed.

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X \rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X \rangle\rangle$-module needs to be computed.
- Usually at least as good as the modified Benson test.

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X \rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X \rangle\rangle$-module needs to be computed.
- Usually at least as good as the modified Benson test.

SK [2013], if $|G|$ is not prime power, $S \leq U \leq G$

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X \rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X \rangle\rangle$-module needs to be computed.
- Usually at least as good as the modified Benson test.

SK [2013], if $|G|$ is not prime power, $S \leq U \leq G$

1. Bound for the generator degrees of $H^*(G)$ in terms of the generating degree of $H^*(U)$ as a $\tau_n H^*(G)$-module.

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X\rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X\rangle\rangle$-module needs to be computed.
- Usually at least as good as the modified Benson test.

SK [2013], if $|G|$ is not prime power, $S \leq U \leq G$

1. Bound for the generator degrees of $H^*(G)$ in terms of the generating degree of $H^*(U)$ as a $\tau_n H^*(G)$-module.
   Very useful: Stability conditions only in *lower* degrees. Expl: $Sz(8)$

Symonds [2010]

- Let $X \subset \tau_n H^*(G)$ be so that $H^*(G)$ is finite over $\langle\langle X\rangle\rangle$.
  E.g., X a subset of a generating set $\rightsquigarrow$ rather small degrees.
- Easy to use: Only the generating degree of $\tau_n H^*(G)$ as a $\langle\langle X\rangle\rangle$-module needs to be computed.
- Usually at least as good as the modified Benson test.

SK [2013], if $|G|$ is not prime power, $S \leq U \leq G$

1. Bound for the generator degrees of $H^*(G)$ in terms of the generating degree of $H^*(U)$ as a $\tau_n H^*(G)$-module.
   Very useful: Stability conditions only in *lower* degrees. Expl: $Sz(8)$
2. Completeness criterion in terms of
   - parameter degrees for $H^*(G; k)$, $k/\mathbb{F}_p$,
   - $\mathrm{depth}\,(H^*(U))$,
   - Hilbert series of $\tau_n H^*(G)$.

Finding graded algebra isomorphisms

Eick, SK [2015]

We provide a complete classification of $H^*(G)$ up to isomorphisms of graded $\mathbb{F}_p$-algebras, for p-groups G, $|G| \leq 81$.

$\|G\|$	#groups	#rings	cum. #groups	cum. #rings
2	1	1	1	1
4	2	2	3	3
8	5	5	8	7
16	14	14	22	18
32	51	48	73	55
64	267	239	340	260
3	1	1	1	1
9	2	2	3	2
27	5	5	8	5
81	15	13	23	14

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, ..., g_n]/Q$.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, ..., g_n]/Q$.
- For any $\{x_1, ..., x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, ..., n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \ldots, g_n]/Q$.
- For any $\{x_1, \ldots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, \ldots, n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \ldots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \ldots, g_n]/Q$.
- For any $\{x_1, \ldots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ ($i = 1, \ldots, n$), we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \ldots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

If $g_i \mapsto x_i$ for all $i \in I \subset \{1, \ldots, n\}$ extends to an isomorphism, then...

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \ldots, g_n]/Q$.
- For any $\{x_1, \ldots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, \ldots, n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \ldots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

If $g_i \mapsto x_i$ for all $i \in I \subset \{1, \ldots, n\}$ extends to an isomorphism, then...

1. equal Hilbert series of $G_I := \langle g_i | i \in I \rangle \lhd R_1$, $X_I := \langle x_i | i \in I \rangle \lhd R_2$.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \dots, g_n]/Q$.
- For any $\{x_1, \dots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, \dots, n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \dots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

If $g_i \mapsto x_i$ for all $i \in I \subset \{1, \dots, n\}$ extends to an isomorphism, then...

1. equal Hilbert series of $G_I := \langle g_i | i \in I \rangle \lhd R_1$, $X_I := \langle x_i | i \in I \rangle \lhd R_2$.
2. substituting x_i for g_i in $Q \cap \langle\langle g_i | i \in I \rangle\rangle \subset \mathbb{F}_p[g_1, \dots, g_n]$ yields zero.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \ldots, g_n]/Q$.
- For any $\{x_1, \ldots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, \ldots, n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \ldots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

If $g_i \mapsto x_i$ for all $i \in I \subset \{1, \ldots, n\}$ extends to an isomorphism, then...

1. equal Hilbert series of $G_I := \langle g_i | i \in I \rangle \lhd R_1$, $X_I := \langle x_i | i \in I \rangle \lhd R_2$.
2. substituting x_i for g_i in $Q \cap \langle\langle g_i | i \in I \rangle\rangle \subset \mathbb{F}_p[g_1, \ldots, g_n]$ yields zero.
3. $\mathrm{Ann}(G_I)$, $\mathrm{Ann}(X_I)$ resp. $\sqrt{G_I}$, $\sqrt{X_I}$ have the same Hilbert series.

Isomorphy of f.p. graded $\mathbb{F}_p$-algebras R_1, R_2 generated in positive degree

Very naive algorithm:

- Let $R_1 \cong \mathbb{F}_p[g_1, \dots, g_n]/Q$.
- For any $\{x_1, \dots, x_n\}$ with $x_i \in R_2^{(|g_i|)}$ $(i = 1, \dots, n)$, we can test if $g_i \mapsto x_i$ extends to a graded isomorphism $R_1 \to R_2$.
- Only *finitely many choices* for $\{x_1, \dots, x_n\}$. Hence we can test in finite time whether or not $R_1 \cong R_2$.

If $g_i \mapsto x_i$ for all $i \in I \subset \{1, \dots, n\}$ extends to an isomorphism, then...

1. equal Hilbert series of $G_I := \langle g_i | i \in I \rangle \lhd R_1$, $X_I := \langle x_i | i \in I \rangle \lhd R_2$.
2. substituting x_i for g_i in $Q \cap \langle\langle g_i | i \in I \rangle\rangle \subset \mathbb{F}_p[g_1, \dots, g_n]$ yields zero.
3. $\mathrm{Ann}(G_I)$, $\mathrm{Ann}(X_I)$ resp. $\sqrt{G_I}$, $\sqrt{X_I}$ have the same Hilbert series.

When we successively increase I, the number of possible mappings of G_I satisfying above criteria often remains fairly small!

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.
- Aim: Compute minimal generating set for M.

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.
- Aim: Compute minimal generating set for M.

"Heady" standard bases [Green 2001]: Similar to Buchberger's algorithm

- Monomial ordering on $\mathcal{P} \rightsquigarrow$ "leading monomials" in $\mathcal{P}$, $\mathcal{A}$, M.

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.
- Aim: Compute minimal generating set for M.

"Heady" standard bases [Green 2001]: Similar to Buchberger's algorithm

- Monomial ordering on $\mathcal{P} \rightsquigarrow$ "leading monomials" in $\mathcal{P}$, $\mathcal{A}$, M.
- For $f \in \mathcal{A}^r$, $G \subset M$: $\mathrm{NF}(f; G) \in \mathcal{A}^r$ (termination?).

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.
- Aim: Compute minimal generating set for M.

"Heady" standard bases [Green 2001]: Similar to Buchberger's algorithm

- Monomial ordering on $\mathcal{P} \rightsquigarrow$ "leading monomials" in $\mathcal{P}$, $\mathcal{A}$, M.
- For $f \in \mathcal{A}^r$, $G \subset M$: $\mathrm{NF}(f; G) \in \mathcal{A}^r$ (termination?).
- "S-polynomials" $\rightsquigarrow G'$ so that $\mathrm{NF}(f; G') = 0 \iff f \in M$.
- By construction, S-polynomials belong to $\mathrm{Rad}(M)$.

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.*, $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.*, M Syzygy module.
- Aim: Compute minimal generating set for M.

"Heady" standard bases [Green 2001]: Similar to Buchberger's algorithm

- Monomial ordering on $\mathcal{P} \rightsquigarrow$ "leading monomials" in $\mathcal{P}$, $\mathcal{A}$, M.
- For $f \in \mathcal{A}^r$, $G \subset M$: $\mathrm{NF}(f; G) \in \mathcal{A}^r$ (termination?).
- "S-polynomials" $\rightsquigarrow G'$ so that $\mathrm{NF}(f; G') = 0 \iff f \in M$.
- By construction, S-polynomials belong to $\mathrm{Rad}(M)$.
- $\mathrm{NF}_h(f; G)$: Only consider radicality preserving reductions.

Minimal generating sets for modules over basic algebras

Setting

- $\mathcal{P}$ path algebra over field K
- $\psi : \mathcal{P} \twoheadrightarrow \mathcal{A}$; *e.g.,* $\mathcal{A}$ basic algebra.
- $\langle g_1, \ldots, g_k \rangle = M \subset \mathcal{A}^r$ right $\mathcal{A}$ module; *e.g.,* M Syzygy module.
- Aim: Compute minimal generating set for M.

"Heady" standard bases [Green 2001]: Similar to Buchberger's algorithm

- Monomial ordering on $\mathcal{P} \rightsquigarrow$ "leading monomials" in $\mathcal{P}$, $\mathcal{A}$, M.
- For $f \in \mathcal{A}^r$, $G \subset M$: $\mathrm{NF}(f; G) \in \mathcal{A}^r$ (termination?).
- "S-polynomials" $\rightsquigarrow G'$ so that $\mathrm{NF}(f; G') = 0 \iff f \in M$.
- By construction, S-polynomials belong to $\mathrm{Rad}(M)$.
- $\mathrm{NF}_h(f; G)$: Only consider radicality preserving reductions.
- **Thm:** If a negative degree ordering is used, the non-radical elements of a heady standard basis form a minimal generating set of M.

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation $\mathrm{ev} : \bigoplus_{i=1}^{k} \mathfrak{e}_i\mathcal{P} \twoheadrightarrow M$, $\mathrm{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i\mathcal{P}$ with $\mathrm{ev}(\tilde{f}) = f \in M$: $\mathrm{Lt}(\tilde{f})$ is an F_5 signature of f.

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation ev : $\bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P} \twoheadrightarrow M$, $\operatorname{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P}$ with $\operatorname{ev}(\tilde{f}) = f \in M$: $\operatorname{Lt}(\tilde{f})$ is an F_5 signature of f.
- Let $\operatorname{NF}_\sigma(f; G)$ be obtained from signature preserving reductions.

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation ev : $\bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P} \twoheadrightarrow M$, $\mathrm{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P}$ with $\mathrm{ev}(\tilde{f}) = f \in M$: $\mathrm{Lt}(\tilde{f})$ is an F_5 signature of f.
- Let $\mathrm{NF}_\sigma(f; G)$ be obtained from signature preserving reductions.
- Disregard all S-polynomials with a signature in lead(ker(ev)).

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation $\mathrm{ev} : \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P} \twoheadrightarrow M$, $\mathrm{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P}$ with $\mathrm{ev}(\tilde{f}) = f \in M$: $\mathrm{Lt}(\tilde{f})$ is an F_5 signature of f.
- Let $\mathrm{NF}_\sigma(f; G)$ be obtained from signature preserving reductions.
- Disregard all S-polynomials with a signature in lead(ker(ev)).
  - *Quotient relations of* $\mathcal{A}$ play the role of trivial Syzygies that are used for the classical commutative F_5.

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation ev : $\bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P} \twoheadrightarrow M$, $\mathrm{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P}$ with $\mathrm{ev}(\tilde{f}) = f \in M$: $\mathrm{Lt}(\tilde{f})$ is an F_5 signature of f.
- Let $\mathrm{NF}_\sigma(f; G)$ be obtained from signature preserving reductions.
- Disregard all S-polynomials with a signature in lead(ker(ev)).
  - *Quotient relations of $\mathcal{A}$* play the role of trivial Syzygies that are used for the classical commutative F_5.
  - Any remaining zero reduction yield non-trivial Syzygies, wich allows to avoid useless S-polynomials later.

Signed standard bases: [SK 2014] inspired by Faugère's F_5 [2002]

Evaluation ev : $\bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P} \twoheadrightarrow M$, $\mathrm{ev}(\mathfrak{e}_i) = g_i$

- If $\tilde{f} \in \bigoplus_{i=1}^{k} \mathfrak{e}_i \mathcal{P}$ with $\mathrm{ev}(\tilde{f}) = f \in M$: $\mathrm{Lt}(\tilde{f})$ is an F_5 signature of f.
- Let $\mathrm{NF}_\sigma(f; G)$ be obtained from signature preserving reductions.
- Disregard all S-polynomials with a signature in lead(ker(ev)).
  - *Quotient relations of $\mathcal{A}$* play the role of trivial Syzygies that are used for the classical commutative F_5.
  - Any remaining zero reduction yield non-trivial Syzygies, wich allows to avoid useless S-polynomials later.

Why we want to use F_5 in future

- **Thm:** If a negative degree ordering is used, a *signed standard basis* allows to read off bases for $\mathrm{Rad}^i(M)$.
- Green's heady algorithm uses only partial information of the F_5-signature that allows to find minimal generating sets but won't avoid useless critical pairs.