Some recent progress with free mathematical software

Viktor Levandovskyy

Universität Kassel, Germany

27.12.2021, Kyiv, online

Where can one publish on the mathematical software?

Everybody knows `https://zbmath.org` but what about `https://swmath.org` – a freely accessible, innovative information service for mathematical software ?

Where to publish or to look for papers on the mathematical software?

- Journal of Symbolic Computation
- Mathematics in Computer Science
- Journal for Software in Algebra and Geometry (JSAG) `https://msp.org/jsag`
- Maple Transactions `www.mapletransactions.org` (Computer-assisted research in mathematics, applications, and education. Use of Maple is not a prerequisite!)
- ...

If you use some software (especially open-source), please cite it properly!

Where can one publish on the mathematical software?

Everybody knows `https://zbmath.org` but what about `https://swmath.org` – a freely accessible, innovative information service for mathematical software ?

Where to publish or to look for papers on the mathematical software?

- Journal of Symbolic Computation
- Mathematics in Computer Science
- Journal for Software in Algebra and Geometry (JSAG) `https://msp.org/jsag`
- Maple Transactions `www.mapletransactions.org` (Computer-assisted research in mathematics, applications, and education. Use of Maple is not a prerequisite!)
- ...

If you use some software (especially open-source), please cite it properly!

- In a recent paper **“Advancing mathematics by guiding human intuition with AI”**, appeared in **Nature**

  `https://www.nature.com/articles/s41586-021-04086-x`

  by Davies et al., incl. Geordie Williamson
- tools and software in the field artificial intelligence were used
- to obtain two novel theoretical results

  in topology and

  in representation theory.

Many problems and questions in modern science can be considered via the **presentation** of modules over associative algebras over a field or even over a ring.

The mother of all associative algebras is the **free associative algebra** $K\langle X\rangle$, where X is a finite alphabet and K is a field. We also consider $R\langle X\rangle$, where R is an associative ring (with 1, often commutative).

A **finitely presented associative algebra** (FPA) is a factor $K\langle X\rangle/I$, where I is a **two-sided ideal** ("of relations"). Example: $\mathbb{Q}(a)[x, y] \cong \mathbb{Q}(a)\langle x, y\rangle/\langle yx - xy\rangle$.

A **finitely presented associative ring** (FPR) is a factor $R\langle X\rangle/I$. Example: $(\mathbb{Z}/2022\mathbb{Z})[x, y] \cong \mathbb{Z}\langle x, y\rangle/\langle yx - xy, 2022\rangle$.

A **module** over an FPA is presented by a matrix. A module can be left, right or bilateral (bimodule).

A free bimodule of rank r over an FPA A is $A\varepsilon_1 A \oplus \ldots \oplus A\varepsilon_r A$.

Many problems and questions in modern science can be considered via the **presentation** of modules over associative algebras over a field or even over a ring.

The mother of all associative algebras is the **free associative algebra** $K\langle X\rangle$, where X is a finite alphabet and K is a field. We also consider $R\langle X\rangle$, where R is an associative ring (with 1, often commutative).

A **finitely presented associative algebra** (FPA) is a factor $K\langle X\rangle/I$, where I is a **two-sided ideal** ("of relations"). Example: $\mathbb{Q}(a)[x,y] \cong \mathbb{Q}(a)\langle x,y\rangle/\langle yx - xy\rangle$.

A **finitely presented associative ring** (FPR) is a factor $R\langle X\rangle/I$. Example: $(\mathbb{Z}/2022\mathbb{Z})[x,y] \cong \mathbb{Z}\langle x,y\rangle/\langle yx - xy, 2022\rangle$.

A **module** over an FPA is presented by a matrix. A module can be left, right or bilateral (bimodule).

A free bimodule of rank r over an FPA A is $A\varepsilon_1 A \oplus \ldots \oplus A\varepsilon_r A$.

Many problems and questions in modern science can be considered via the **presentation** of modules over associative algebras over a field or even over a ring.

The mother of all associative algebras is the **free associative algebra** $K\langle X\rangle$, where X is a finite alphabet and K is a field. We also consider $R\langle X\rangle$, where R is an associative ring (with 1, often commutative).

A **finitely presented associative algebra** (FPA) is a factor $K\langle X\rangle/I$, where I is a **two-sided ideal** ("of relations").
Example: $\mathbb{Q}(a)[x,y] \cong \mathbb{Q}(a)\langle x,y\rangle/\langle yx - xy\rangle$.

A **finitely presented associative ring** (FPR) is a factor $R\langle X\rangle/I$.
Example: $(\mathbb{Z}/2022\mathbb{Z})[x,y] \cong \mathbb{Z}\langle x,y\rangle/\langle yx - xy, 2022\rangle$.

A **module** over an FPA is presented by a matrix. A module can be left, right or bilateral (bimodule).

A free bimodule of rank r over an FPA A is $A\varepsilon_1 A \oplus \ldots \oplus A\varepsilon_r A$.

Many problems and questions in modern science can be considered via the **presentation** of modules over associative algebras over a field or even over a ring.

The mother of all associative algebras is the **free associative algebra** $K\langle X\rangle$, where X is a finite alphabet and K is a field.
We also consider $R\langle X\rangle$, where R is an associative ring (with 1, often commutative).

A **finitely presented associative algebra** (FPA) is a factor $K\langle X\rangle/I$, where I is a **two-sided ideal** ("of relations").
Example: $\mathbb{Q}(a)[x,y] \cong \mathbb{Q}(a)\langle x,y\rangle/\langle yx - xy\rangle$.

A **finitely presented associative ring** (FPR) is a factor $R\langle X\rangle/I$.
Example: $(\mathbb{Z}/2022\mathbb{Z})[x,y] \cong \mathbb{Z}\langle x,y\rangle/\langle yx - xy, 2022\rangle$.

A **module** over an FPA is presented by a matrix. A module can be left, right or bilateral (bimodule).

A free bimodule of rank r over an FPA A is $A\varepsilon_1 A \oplus \ldots \oplus A\varepsilon_r A$.

Problems, when leaving cosy commutative fin. gen. world:

- non-terminating **procedure** (an *algorithm* has termination!)
- absense of canonical forms, hardness of its computation
- decidability issues

Non-commutative computer algebra over free assoc. algebras:

- Gröbner(–Shirshov) bases (two-sided)
- normal forms (remainder after two-sided division)
- Gröbner lifting (book/trace keeping)

Problems, when leaving cosy commutative fin. gen. world:

- non-terminating **procedure** (an *algorithm* has termination!)
- absense of canonical forms, hardness of its computation
- decidability issues

Non-commutative computer algebra over free assoc. algebras:

- Gröbner(–Shirshov) bases (two-sided)
- normal forms (remainder after two-sided division)
- Gröbner lifting (book/trace keeping)

Computer algebra systems for free associative algebras

Legendary systems, still running/downloadable

- GBNP package for GAP by A M. Cohen et al., last update in 2016; `https://gap-packages.github.io/gbnp`
- Bergman by J. Backelin, V. Ufnarovski et al., last update in 2011; `servus.math.su.se/bergman`
- Felix by J. Apel, `felix.hgb-leipzig.de`
- MAS by H. Kredel `krum.rz.uni-mannheim.de/mas/`

Still running/downloadable: **but for how long?**
Problems: discontinued software/packages, violation of vertical compatibility...
Solutions: containering via e.g. Docker ...

Computer algebra systems for free associative algebras

Legendary systems, still running/downloadable

- GBNP package for GAP by A M. Cohen et al., last update in 2016; `https://gap-packages.github.io/gbnp`
- Bergman by J. Backelin, V. Ufnarovski et al., last update in 2011; `servus.math.su.se/bergman`
- Felix by J. Apel, `felix.hgb-leipzig.de`
- MAS by H. Kredel `krum.rz.uni-mannheim.de/mas/`

Still running/downloadable: **but for how long?**
Problems: discontinued software/packages, violation of vertical compatibility...
Solutions: containering via e.g. Docker ...

Computer algebra systems for free associative algebras

Actual systems

- MAGMA: Gröbner bases subject to *deg left lex* ordering
- JAS by H. Kredel `http://krum.rz.uni-mannheim.de/jas`
- NCAlgebra via Mathematica by W. Helton et al., last update in 2017, `http://math.ucsd.edu/~ncalg`,

- Singular:Letterplace 4-2-1: very rich functionality
  - recent papers by Levandovskyy et al. at ISSAC 2020
  - the only system, offering Gröbner bases for modules over $\mathbb{Z}\langle X\rangle$

Computer algebra systems for free associative algebras

Actual systems

- MAGMA: Gröbner bases subject to *deg left lex* ordering
- JAS by H. Kredel `http://krum.rz.uni-mannheim.de/jas`
- NCALGEBRA via MATHEMATICA by W. Helton et al., last update in 2017, `http://math.ucsd.edu/~ncalg`,

- SINGULAR:LETTERPLACE 4-2-1: very rich functionality
  - recent papers by Levandovskyy et al. at ISSAC 2020
  - the only system, offering Gröbner bases for modules over $\mathbb{Z}\langle X\rangle$

Julia and OSCAR

Julia language

- The usage of the modern Julia language https://julialang.org/, backed by MIT (and meanwhile, many others)
- helps to achieve high level of integration of tools from various mathematical areas
- It already offers a developed ecosystem of high performance packages

Julia and OSCAR

OSCAR Project, supported by the German DFG

https://oscar.computeralgebra.de is build in the Julia ecosystem; in turn, it builds on and extends the four cornerstone systems

- GAP – computational discrete algebra
  https://www.gap-system.org
- Singular – commutative and non-commutative algebra, algebraic geometry
  https://www.singular.uni-kl.de
- Polymake – polyhedral geometry
  https://polymake.org/
- Antic (Hecke, Nemo) – number theory
  https://nemocas.org

SageMath `https://www.sagemath.org` among other – *wraps* other systems.

A system HomAlg by M. Barakat (Siegen) et al.

`https://homalg-project.github.io`

- written in the object–oriented language of GAP ("logic")
- supports the construction of various sorts of computable categories and performs homological algebra computations within computable Abelian and triangulated categories
- **delegates** concrete calculations with matrices over rings to various computer algebra systems
- features the CAP project – "Categories, Algorithms, and Programming" and further advanced packages

SageMath `https://www.sagemath.org` among other – *wraps* other systems.

A system HomAlg by M. Barakat (Siegen) et al.

`https://homalg-project.github.io`

- written in the object–oriented language of GAP (“logic”)
- supports the construction of various sorts of computable categories and performs homological algebra computations within computable Abelian and triangulated categories
- **delegates** concrete calculations with matrices over rings to various computer algebra systems
- features the CAP project – “Categories, Algorithms, and Programming” and further advanced packages

What is SINGULAR?

- open source computer algebra system (made in Germany)
- 40+ years of experience, very good with Gröbner bases
- freely available from `http://www.singular.uni-kl.de`
- SAGE and HOMALG can use S. as a backend
- S. is a part of the visionary OSCAR system
- S. has connections with MATHEMATICA, MAPLE etc.

What is LETTERPLACE?

- a subsystem of SINGULAR, providing the manipulations and computations within free associative algebras ...
  - over fields, supported by SINGULAR (a big list)
  - over the ring $\mathbb{Z}$ directly ...
  - and over $\mathbb{Z}\langle Y \rangle / J$ via elimination orderings!
- started around 2007, underwent several releases
- based on **Letterplace** theory by R. La Scala and V. Levandovskyy.

What is SINGULAR?

- open source computer algebra system (made in Germany)
- 40+ years of experience, very good with Gröbner bases
- freely available from `http://www.singular.uni-kl.de`
- SAGE and HOMALG can use S. as a backend
- S. is a part of the visionary OSCAR system
- S. has connections with MATHEMATICA, MAPLE etc.

What is LETTERPLACE?

- a subsystem of SINGULAR, providing the manipulations and computations within free associative algebras ...
  - over fields, supported by SINGULAR (a big list)
  - over the ring $\mathbb{Z}$ directly ...
  - and over $\mathbb{Z}\langle Y\rangle/J$ via elimination orderings!
- started around 2007, underwent several releases
- based on **Letterplace** theory by R. La Scala and V. Levandovskyy.


Viktor Levandovskyy
Kassel
Tobias Metzlaff
INRIA Sophia Antipolis
Hans Schönemann
TU Kaiserslautern
Karim Abou Zeid
RWTH Aachen

SINGULAR letterplace

1. V. Levandovskyy, H. Schönemann and K. Abou Zeid. LETTERPLACE — a Subsystem of SINGULAR for Computations with Free Algebras via Letterplace Embedding. In *Proc. of the Int. Symposium on Symbolic and Algebraic Computation (ISSAC'20)*, ACM Press, 305–311, 2020.

2. V. Levandovskyy, T. Metzlaff and K. Abou Zeid. Computations of free non-commutative Gröbner Bases over $\mathbb{Z}$ with SINGULAR:LETTERPLACE. In *Proc. ISSAC'20*, 312–319, 2020.

2'. Extended and enhanced version of **2** appears soon in J. of Symbolic Computation.

3. L. Schmitz and V. Levandovskyy. Formally Verifying Proofs for Algebraic Identities of Matrices. In *Intelligent Computer Mathematics* Springer LNCS, LNAI 222–236, 2020.

Gröbner Technology = Gröbner Trinity + Gröbner Basics

Gröbner Trinity consists of three components

1. STD/GB Gröbner basis $\mathcal{G}$ of a module M
2. SYZ Gröbner basis of the syzygy module of M
3. LIFT the transformation matrix between two bases $\mathcal{G}$ and M

The function LIFTSTD computes all the trinity data at once.

Gröbner Trinity should be formulated separately for one-sided (left and right) and for two-sided modules (bimodules).

Implementation: we offer `twostd` and `rightstd` functions.

Gröbner Technology = Gröbner Trinity + Gröbner Basics

Gröbner Trinity consists of three components

1. STD/GB Gröbner basis $\mathcal{G}$ of a module M
2. SYZ Gröbner basis of the syzygy module of M
3. LIFT the transformation matrix between two bases $\mathcal{G}$ and M

The function LIFTSTD computes all the trinity data at once.

Gröbner Trinity should be formulated separately for one-sided (left and right) and for two-sided modules (bimodules).

Implementation: we offer `twostd` and `rightstd` functions.

Live example over $\mathbb{Z}$: Iwahori–Hecke algebra

An Iwahori–Hecke algebra is associated to Coxeter group.
It is constructed by means of finite presentation over $\mathbb{Z}[q, q^{-1}]$ where q will later be specialized, most frequently to the root of unity over a finite field.

Iwahori–Hecke algebra of type A_3

It is presented as the factor-algebra of $\mathbb{Z}[q, q^{-1}]\langle x, y, z\rangle$ modulo

$$\langle x^2 + (1-q)x - q, y^2 + (1-q)y - q, z^2 + (1-q)z - q,$$

$$zx - xz, yxy - xyx, zyz - yzy\rangle.$$

A Gröbner basis of the above contains just one new element: $xyzx - yxyz$.
Further: we specialize q to the third primitive root of unity.
We conclude that, specialized over any field $\mathbb{K}$, the Iwahori-Hecke algebra of type A_3 is of dimension 24. **Thank you!**

Live example over $\mathbb{Z}$: Iwahori–Hecke algebra

An Iwahori–Hecke algebra is associated to Coxeter group.
It is constructed by means of finite presentation over $\mathbb{Z}[q, q^{-1}]$ where q will later be specialized, most frequently to the root of unity over a finite field.

Iwahori–Hecke algebra of type A_3

It is presented as the factor-algebra of $\mathbb{Z}[q, q^{-1}]\langle x, y, z\rangle$ modulo

$$\langle x^2 + (1-q)x - q, y^2 + (1-q)y - q, z^2 + (1-q)z - q,$$

$$zx - xz, yxy - xyx, zyz - yzy\rangle.$$

A Gröbner basis of the above contains just one new element: $xyzx - yxyz$.
Further: we specialize q to the third primitive root of unity.
We conclude that, specialized over any field $\mathbb{K}$, the Iwahori-Hecke algebra of type A_3 is of dimension 24. **Thank you!**