

Oktober 2021

# Computeralgebra Rundbrief

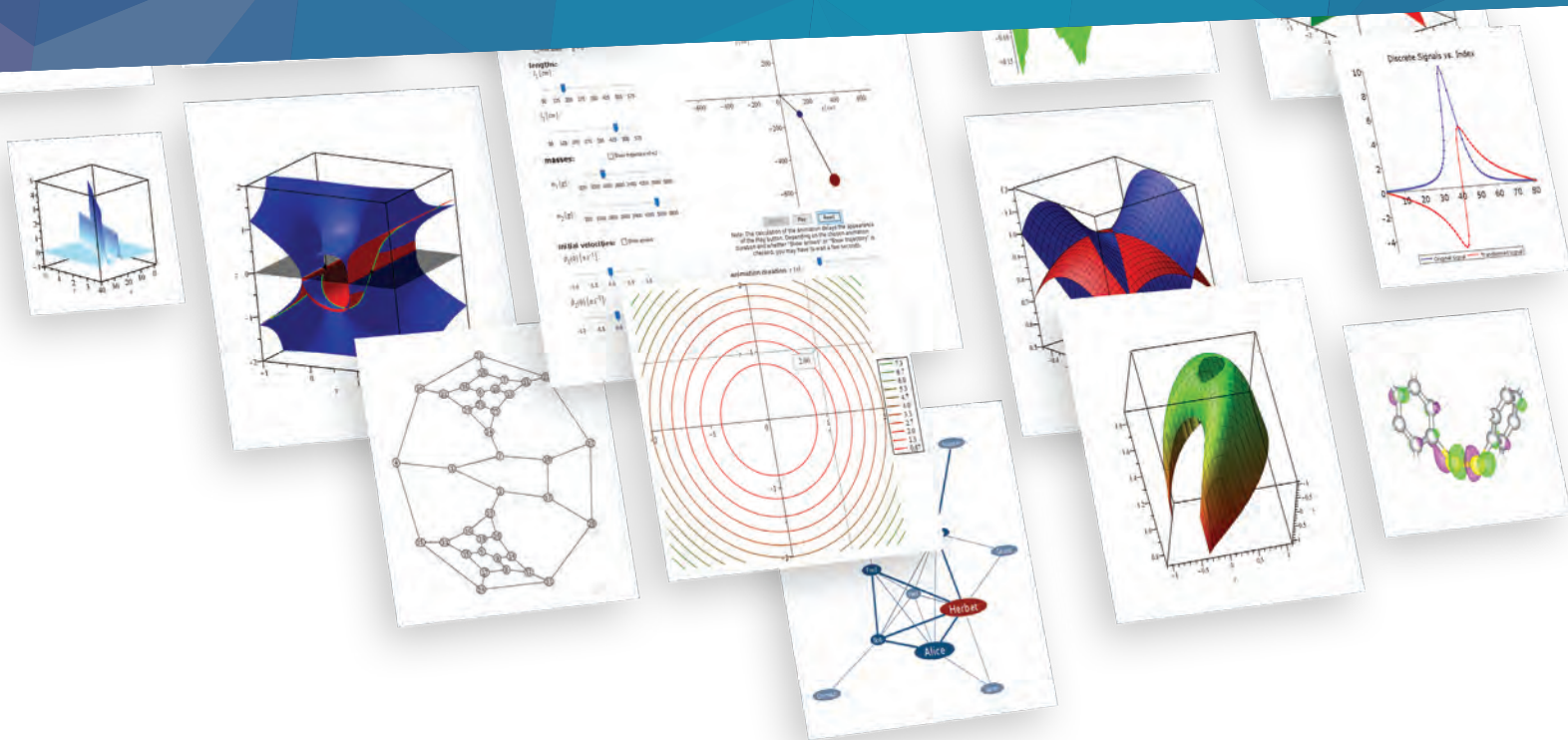
> Ausgabe 69

- ▶ Computeralgebra-Tagung der Fachgruppe
- ▶ Combinatorial Objects with Orbiter
- ▶ Algebra and Condensed Matter
- ▶ Industrietagung der Fachgruppe



# Sie nutzen Maple 2021 noch nicht?

Sehen Sie, was Sie bisher versäumt haben



## Neues Maple 2021 jetzt verfügbar!

Von kleinen, aber unglaublich nützlichen Verbesserungen der Benutzeroberfläche bis hin zu ganz neuen Bereichen der Mathematik bietet Maple 2021 eine Reihe von Erweiterungen im gesamten Produkt.

Probieren Sie Maple kostenlos für 15 Tage ohne Verpflichtungen  
[www.maplesoft.com/CAR2021](http://www.maplesoft.com/CAR2021)



## Inhaltsverzeichnis

<b>Inhalt</b> . . . . .	3
<b>Impressum</b> . . . . .	4
<b>Mitteilungen der Sprecher</b> . . . . .	5
<b>Tagungen der Fachgruppe</b> . . . . .	6
<b>Themen und Anwendungen</b> . . . . .	10
<i>Algebra and Condensed Matter</i> (R. M. Geilhufe) . . . . .	10
<b>Neues über Systeme</b> . . . . .	16
<i>Classification of Combinatorial Objects with Orbiter</i> (A. Betten) . . . . .	16
<b>Publikationen über Computeralgebra</b> . . . . .	23
<b>Besprechungen zu Büchern der Computeralgebra</b> . . . . .	24
<i>B. Fine et al.: Topic in Infinite Group Theory</i> (M. Kreuzer) . . . . .	24
<i>L. Bokut et al.: Gröbner Shirshov Bases</i> (M. Kreuzer) . . . . .	25
<b>Berufungen</b> . . . . .	26
<b>Promotionen in der Computeralgebra</b> . . . . .	27
<b>Berichte von Konferenzen</b> . . . . .	28
<b>Hinweise auf Konferenzen</b> . . . . .	30
<b>Fachgruppenleitung Computeralgebra 2020–2023</b> . . . . .	31

## Impressum

Der Computeralgebra-Rundbrief wird herausgegeben von der Fachgruppe Computeralgebra der GI in Kooperation mit der DMV und der GAMM (verantwortlicher Redakteur: Dr. Fabian Reimers [car@mathematik.de](mailto:car@mathematik.de))

Der Computeralgebra-Rundbrief erscheint halbjährlich, Redaktionsschluss 15.02. und 15.09. ISSN 0933-5994. Mitglieder der Fachgruppe Computeralgebra erhalten je ein Exemplar dieses Rundbriefs im Rahmen ihrer Mitgliedschaft. Fachgruppe Computeralgebra im Internet: <http://www.fachgruppe-computeralgebra.de>.

Konferenzankündigungen, Mitteilungen, einzurichtende Links, Manuskripte und Anzeigenwünsche bitte an den verantwortlichen Redakteur.

**GI** (Gesellschaft für  
Informatik e.V.)  
Wissenschaftszentrum  
Ahrstr. 45  
53175 Bonn  
Telefon 0228-302-145  
Telefax 0228-302-167  
[bonn@gi.de](mailto:bonn@gi.de)  
<https://gi.de>



**DMV** (Deutsche Mathematiker-  
Vereinigung e.V.)  
Mohrenstraße 39  
10117 Berlin  
Telefon 030-20377-306  
Telefax 030-20377-307  
[dmv@wias-berlin.de](mailto:dmv@wias-berlin.de)  
<https://www.mathematik.de>



**GAMM** (Gesellschaft für Angewandte  
Mathematik und Mechanik e.V.)  
Technische Universität Dresden  
Institut für Statik und Dynamik der  
Tragwerke  
01062 Dresden  
Telefon 0351-463-33448  
Telefax 0351-463-37086  
[GAMM@mailbox.tu-dresden.de](mailto:GAMM@mailbox.tu-dresden.de)  
<https://www.gamm-ev.de>



---

## Mitteilungen der Sprecher

---

*Liebe Mitglieder der Fachgruppe Computeralgebra,*

*noch immer liegt die Pandemie wie ein schwerer Schatten über vielen unserer Aktivitäten. So waren im vergangenen Semester erneut zahlreiche Veranstaltungen auf ein virtuelles oder bestenfalls hybrides Format beschränkt. Auch unsere Industrietagung, von der wir auf Seite 8 berichten, konnte trotz der Verschiebung um ein ganzes Jahr nicht in Präsenz stattfinden, was aber andererseits durch Vortragende aus dem Ausland, die sonst sicher nicht angereist wären, auch einen positiven Effekt hatte.*

*Umso mehr freut man sich dann, den einen oder anderen Kollegen, die eine oder andere Kollegin bei ersten Präsenzveranstaltungen endlich einmal persönlich wiederzutreffen. Vorträge, Lehre und Zusammenarbeit über elektronische Medien sind zwar möglich und haben für die meisten von uns auch mehr oder minder geklappt, aber gerade die informellen Gespräche in kleinen Grüppchen mit der Kaffeetasse in der Hand bringen doch ganz andere Themen und manchmal auch ganz unerwartete Ideen zu Tage. Hoffen wir also, dass die Fachgruppentagung, die von 2021 auf den März 2022 verschoben wurde, uns allen wieder eine solche Möglichkeit bieten wird! Sie ist definitiv als Präsenz-Veranstaltung geplant. Wir konnten auch diesmal wieder Hauptvortragende aus sehr verschiedenen Bereichen gewinnen, von denen wir interessante Schlaglichter erwarten. Wie man in der Ankündigung auf Seite 6 sieht, weichen Zeit und Ort der Tagung diesmal etwas von bisher Gewohntem ab. Um so mehr möchten wir Sie bitten, Werbung für die Tagung zu machen und insbesondere den wissenschaftlichen Nachwuchs in den Arbeitsgruppen zur Teilnahme und vor allem zu eigenen Vorträgen zu ermuntern.*

*Inhaltlich beschäftigt sich der Rundbrief diesmal mit dem Einsatz von Computeralgebra in der Physik kondensierter Materie (ab Seite 10) sowie mit dem auf kombinatorische Objekte ausgelegten Computeralgebra System Orbiter (ab Seite 16). Abgerundet wird die aktuelle Ausgabe dann wie immer von Buchbesprechungen sowie Konferenzberichten und -hinweisen.*

*Falls es Ihnen so geht wie uns und Sie den Bereich Computeralgebra und Schule in diesem Rundbrief vermissen, dann sind Sie vielleicht ein Adressat für unseren Aufruf, sich genau für diesen Bereich in der Fachgruppe zu engagieren. Tatsächlich sind uns Beiträge für den Rundbrief willkommen, und die Fachgruppenleitung würde gerne bald die vakante Position einer Fachexpertin oder eines Fachexperten für Schule und Didaktik füllen.*

*Wir wünschen Ihnen eine angenehme und anregende Lektüre.*

*Anne Frühbis-Krüger*

*Gregor Kemper*

### Tagung der Fachgruppe Computeralgebra

#### München

**9. bis 11. März 2022**

<http://www.fachgruppe-computeralgebra.de/muenchen-2022>

In Fortsetzung der erfolgreichen Tagungen 2003, 2005, 2009, 2012, 2014, 2017, 2019 in Kassel und 2007 in Kaiserslautern führt die Fachgruppe ihre 9. Computeralgebra-Tagung im kommenden Jahr diesmal in München durch. Turnusgemäß für 2021 geplant, jedoch wegen der COVID-19-Pandemie verschoben, soll die Tagung 2022 wieder ganz in Präsenz stattfinden. Neben dem Wechsel des Tagungsortes von Kassel zu München gibt es auch beim Termin eine Neuerung. So wird die Tagung von Mittwoch (9.3.) bis Freitag (11.3.) im März abgehalten statt wie bisher üblich von Donnerstag bis Samstag im Mai.



Ansonsten knüpfen wir an die bewährte Tradition dieser Tagungsreihe an. So bleibt es das Ziel, einerseits einige Hauptvortragende zu gewinnen, die Übersichtsvorträge über wichtige Gebiete der Computeralgebra und über Computeralgebra-Software geben, andererseits aber auch Nachwuchswissenschaftler|innen zu ermöglichen ihre Ergebnisse vorzustellen. Vorträge sind auf Englisch und Deutsch gleichermaßen willkommen. Die Fachgruppe vergibt wieder einen mit 500 € dotierten Preis für den besten Vortrag einer Nachwuchswissenschaftler|in.

Hierzu sind Nachwuchswissenschaftler|innen (**Doktoranden, Postdocs**) aufgefordert, sich bis zum **31.1.2022** mit einem **Vortrag anzumelden**.

Als Hauptvortragende konnten wir folgende Wissenschaftler|innen gewinnen:

- **Melanie Harms** (RWTH Aachen)
- **Tommy Hofmann** (Universität Siegen)
- **Lukas Kühne** (MPI Leipzig)
- **Veronika Pillwein** (RISC Linz)
- **Timo de Wolff** (TU Braunschweig)



Münchener Frauenkirche und Neues Rathaus, Foto Wikipedia, lizenziert unter CC BY-SA 3.0.



**Website:** <http://www.fachgruppe-computeralgebra.de/muenchen-2022>

**Termin und Ort:** Die Tagung findet in der Zeit vom 9. – 11. März 2022 am Max-Planck-Institut für Physik in München statt. Sie wird am 9. März 2022 um circa 13:00 Uhr eröffnet (Anreisetag) und endet am 11. März 2022 um circa 12:30 Uhr (Abreisetag).

**Anmeldung:** Die Anmeldung eines Vortrags ist bis 31. Januar 2022 möglich. Die Anmeldung ohne Vortrag ist bis 28. Februar 2022 möglich. Details zur Anmeldung finden Sie auf der Website der Tagung.

**Konferenzgebühren:** Jedes Nichtmitglied der Fachgruppe entrichtet vor Ort einen Unkostenbeitrag in Höhe von 20 € für die Kaffeepausen, alternativ kann man vor Ort zum Jahresbeitrag von 9 € Mitglied der Fachgruppe werden.

**Nachwuchspreis:** Die Fachgruppe Computeralgebra vergibt für den besten Vortrag eines Nachwuchswissenschaftlers wieder einen mit 500 € dotierten Nachwuchspreis. Verbunden mit dem Geldpreis ist die Einladung auf der nächsten Tagung der Fachgruppe einen Hauptvortrag zu halten.



*Foto der letzten Tagung 2019 in Kassel*



## **Industrietagung der Fachgruppe**

Oldenburg, 29.09.-30.09.2021

[www.fachgruppe-computeralgebra.de/industrietagung2021](http://www.fachgruppe-computeralgebra.de/industrietagung2021)

Nach über 10-jähriger Pause fand am 28. und 29. September 2021 in Oldenburg wieder eine Industrietagung der Fachgruppe Computeralgebra statt. Diesmal war der Fokus ganz aktuell gesetzt auf kryptographischen Themen und IT-Sicherheit – passend zu dem gerade laufenden Standardisierungsverfahren für PostQuanten-Kryptographie beim NIST. Trotz Corona-bedingter Verschiebung von 2020 auf dieses Jahr und eigentlich vielversprechender Vorzeichen für eine Tagung in Präsenz zwang Corona am Ende doch zu einer reinen Online-Tagung, was natürlich gerade den gewünschten informellen Austausch am Rande stark einschränkte.

Gleich zu Beginn des Workshops führte Manfred Lochter (Bundesamt für Sicherheit in der Informationstechnik) in die aktuelle Lage ein, bekannte und noch weitgehend unbekannte Bedrohungen kamen zur Sprache, ebenso gängige und post-quanten-sichere Algorithmen. Dabei gab er den Zuhörern mit auf den Weg, dass in Zukunft gerade das Zusammenspiel von physikalischen Methoden und mathematischer Post-Quanten-Kryptographie von Bedeutung sein wird.

Im Themenschwerpunkt zu Post-Quanten Kryptographie führte Joppe Bos in FrodoKEM ein und gab einen Einblick in die Problematik bei Embedded Devices, während Edoardo Persichetti den Schwerpunkt seiner Präsentation auf Code-based Cryptography legte. Algebraische Kryptanalyse stand im Zentrum der Vorträge von Virendra Sule und Peter Nonnenmann.

Die Brücke zwischen Wissenschaft und Industrie schlug Marc Kaplan in seinem Vortrag zu Quantennetzwerken. Ganz auf der Industrieseite bewegten sich die Vorträge von Stjepan Kovac und Andy Jenkinson zu Cybersecurity-Aspekten und der Vortrag von Benjamin Sinram zum Zusammenspiel von Produktentwicklung, Softwareentwicklung und Kryptographie als Sicherheitstechnologie.

Insgesamt war es ein rundes Programm mit vielen verschiedenen Aspekten, aus dem gerade die jüngeren Teilnehmer auch den ein oder anderen Einblick in die Arbeit jenseits der Universität mitnehmen konnte. Hoffen wir, dass bis zum nächsten Mal nicht wieder zehn Jahre vergehen werden.

Anne Frühbis-Krüger (Oldenburg)



# Online-Workshop Computational Algebra 2021

Hannover, Oldenburg (Online), 21.05.2021

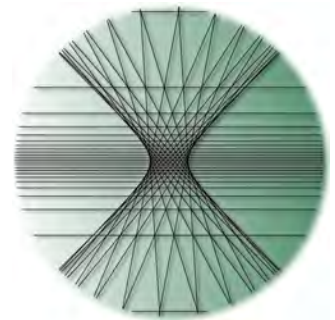
[www.fachgruppe-computeralgebra.de/ca2021](http://www.fachgruppe-computeralgebra.de/ca2021)

Da die Fachgruppentagung um ein Jahr verschoben werden musste, und wir dennoch regelmäßig eine Plattform zur Kommunikation der aktuellen Forschung in der Computeralgebra benötigen, fand im Frühling 2021 wieder ein Online-Kurzworkshop der Fachgruppe statt. Der erste solche Workshop im Herbst 2020 wurde sehr positiv aufgenommen, der zweite Workshop verlief deshalb nach demselben Schema.

Da der erste Online-Workshop durch Zusammenlegen zweier Minisymposia der DMV-Tagung entstanden war und somit auch Publikum aus dem angrenzenden Gebiet „Algebraic Methods for the Sciences“ teilgenommen hatte, war die Beteiligung beim zweiten Workshop etwas geringer ausgefallen. Natürlich fehlte außerdem der übliche Austausch zwischen den Vorträgen; das Programm war aber trotz allem genauso bereichernd und topaktuell wie immer: Vortragende waren Simon Brandhorst (Saarbrücken), Jean Philippe Labbé (Berlin), Timo de Wolff (Braunschweig), Jonathan Kliem (Berlin).

Insgesamt konnten wir durch die beiden Online-Workshops die Zeit bis zur nächsten Fachgruppentagung im März 2022 (siehe Ankündigung auf Seite 6) gut überbrücken.

Michael Cuntz (Hannover)



## Antrag auf Mitgliedschaft in der Fachgruppe Computeralgebra

Die Fachgruppe Computeralgebra sieht es als ihre Aufgabe an, Lehre, Forschung, Entwicklung, Anwendungen, Informationsaustausch und Zusammenarbeit auf dem Gebiet der Computeralgebra in Deutschland zu fördern.

Eine Mitgliedschaft in der Fachgruppe Computeralgebra gibt es bereits ab 7,50 € pro Jahr (für Mitglieder von DMV, GI oder GAMM; ansonsten 9 €).

### Vorteile einer Mitgliedschaft:

- Sie fördern durch Ihren Beitrag die Workshops, Seminare, Tagungen und andere Aktivitäten auf dem Gebiet der Computeralgebra, die die Fachgruppe organisiert und unterstützt.
- Sie erhalten zweimal im Jahr den Computeralgebra-Rundbrief mit vielen interessanten Informationen rund um die Computeralgebra frei Haus.
- Sie verleihen unserer Stimme an Gewicht, die wir aktiv in Diskussionen um die Stellung der Computeralgebra in der Ausbildung in Schule und Hochschule einbringen.

Wir würden uns sehr über Ihre Unterstützung freuen. Die Mitgliedschaft in der Fachgruppe steht allen offen. Weiter Informationen zur Mitgliedschaft und einen Aufnahmeantrag finden Sie auf unserer Webseite unter folgender Adresse, oder scannen Sie einfach den QR-Code.

<https://fachgruppe-computeralgebra.de/aufnahmeantrag>



### Algebra and Condensed Matter

R. Matthias Geilhufe

(Nordita, Stockholm University, KTH - Royal Institute of Technology)

matthias.geilhufe@su.se



---

#### Introduction

---

Condensed matter physics deals with the properties of matter at low temperatures. It aims to describe the microscopic origin of macroscopic phenomena. The macroscopic properties are strongly bound to the phase of matter, for example: solid, characterized by structural rigidity; ferromagnet, characterized by a finite magnetization; superconductor, characterized by a vanishing electrical resistivity. A phase transition denotes the sudden change of a macroscopic property depending on a control parameter, e.g., temperature, pressure, chemical composition.

The difficulty of the subject lies in the complexity of the many-body problem. For example, 1 cm<sup>3</sup> iron contains  $\approx 8.5 \times 10^{22}$  iron atoms with  $\approx 2.2 \times 10^{24}$  electrons in total. At the same time, the complexity leads to an incredible richness of physical phenomena.

Among others, algebraic methods help in the classification of states of matter, the formulation of effective theories, or determining degenerate levels in spectra. The aim of the present paper is to introduce these ideas to a broad audience of mathematics background. We will focus on a few key examples.

We start with a brief outline of the mathematical framework of quantum mechanics. Next, we show how symmetries in quantum mechanical systems determine level degeneracies. We motivate this result with two examples, first discussing an atomic system and second a crystal, i.e., a lattice-periodic solid. Afterwards, we show how symmetries can be used to formulate an effective theory for electronic states in materials. Such effective theories are often sufficient to study the low temperature properties of complex materials. At the final part of this survey, we introduce the phenomenological theory of phase transitions. Here, again, symmetry is a key ingredient.

By no means does this review cover the full spectrum of relevant topics in condensed matter theory nor the potential computer algebra has in the field. Already, the few examples picked are only tiny bits of entire fields of study. Often, I do not provide a fair account of rele-

vant references. Whenever possible I refer to a review article which might guide the interested reader.

---

#### Quantum mechanics

---

Phenomena of submicroscopic particles are governed by quantum mechanics. Its development about 100 years ago counts as one of the most significant breakthroughs in physics history. In quantum mechanics, a system is described by a state vector  $|\psi\rangle \in \mathcal{H}$  in a Hilbert space  $\mathcal{H}$ . Typically,  $\mathcal{H}$  is defined over the field of the complex numbers  $\mathbb{C}$ . We use the so-called Dirac or bra-ket notation. Following the Riesz–Fréchet representation theorem, each vector  $|\psi\rangle$  is associated a dual vector  $\langle\psi|$ .  $\langle\psi|$  is a linear functional constituting a map  $\mathcal{H} \rightarrow \mathbb{C}$ . Or, in simple notation, we write the inner product as the *braket*  $\langle\psi|\psi\rangle$ , which motivates the terms bra-vector  $\langle\psi|$  and ket vector  $|\psi\rangle$ .

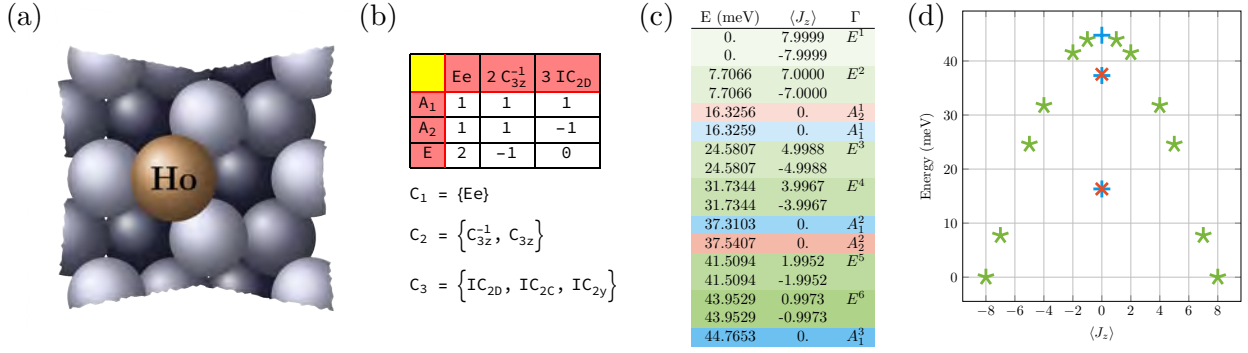
Observables, such as momentum or angular momentum are represented by linear operators  $O : \mathcal{H} \rightarrow \mathcal{H}$ . Traditionally, the operator is chosen to be Hermitian to ensure a real valued spectrum as measured in experiments. However, this restriction has been lifted, e.g., by the development of the PT-symmetric quantum mechanics by Bender and Boetcher [1] (P is the parity operation mapping the position vector  $\mathbf{r} \rightarrow -\mathbf{r}$ ; T is the time-reversal  $t \rightarrow -t$ ) or in the so-called non-Hermitian quantum mechanics [2] describing open quantum systems. In the latter, non-Hermitian components represent e.g. source and drain terms.

The dynamics of a quantum system is governed by the Schrödinger equation

$$H|\psi\rangle = i\hbar \frac{\partial}{\partial t} |\psi\rangle. \quad (1)$$

Here, H is an energy-valued operator called Hamilton operator. We choose H to be Hermitian. The time is denoted by  $t$  and  $\hbar$  is the Planck constant. In the case of a static system, i.e., with no explicit time dependence, the Schrödinger equation becomes an eigenvalue equation

$$H|\psi\rangle = E|\psi\rangle, \quad (2)$$



**Figure 1:** A Holmium atom on a platinum surface experiences a trigonal potential from the surrounding Pt atoms. As a consequence, the lowest lying Ho state splits with eigenvalues in agreement with the irreducible representations of the group  $C_{3v}$ . (a) surface structure. (b) character table of  $C_{3v}$ , calculated with GTPack [3, 4]. (c) calculated energy levels. (d) energy levels versus expectation value of the total angular momentum. More details can be found in Ref. [5].

with  $E$  being the energy eigenstate of  $|\psi\rangle$ . A symmetry  $U$  of the Hamiltonian  $H$  corresponds to a similarity transformation of  $H$  mediated by an invertible operator  $U$ ,

$$UHU^{-1} = H. \quad (3)$$

The set of all symmetries forms a group  $\mathcal{G}$ . The identity is given by  $U = 1$ . From equation (3), it follows if  $U \in \mathcal{G}$  also  $U^{-1} \in \mathcal{G}$ . Furthermore, for  $U, V \in \mathcal{G}$  it follows from direct computation  $UV \in \mathcal{G}$ . Applying  $U$  from the left to the Schrödinger equation (2) and inserting  $1 = U^{-1}U$ , gives

$$UHU^{-1}U|\psi\rangle = HU|\psi\rangle = EU|\psi\rangle. \quad (4)$$

Hence, if  $|\psi\rangle$  is an eigenfunction of  $H$ , then also  $U|\psi\rangle$  is an eigenfunction. Assuming  $\mathcal{G}$  to be a finite group and applying all  $U \in \mathcal{G}$  to  $|\psi\rangle$ , we can project at most  $d$  linearly independent vectors  $|\psi_1\rangle, \dots, |\psi_d\rangle$ . The matrices given by

$$U_{ij} = \langle \psi_i | U | \psi_j \rangle, \quad (5)$$

form an irreducible representation of  $\mathcal{G}$ . As a consequence, the degeneracy of the energy levels in a quantum system are tightly bound to the dimensions of the irreducible representations of the underlying symmetry group.

### Example: degenerate levels

Group theory allows for estimating expected level degeneracy in spectra. This can be of great importance for functionalizing materials by symmetry breaking, e.g., by applying pressure. It also plays a significant role in quantum chemistry. In Figure 1, we show an example for the ground state level splitting of a single Ho atom on a highly conducting Pt surface. Here, the huge magnetic moment of Ho can be switched between two possible states, with measured lifetimes in the order of minutes [5]. On the (111) surface of a Pt crystal, a single Ho atom experiences a trigonal potential from the surrounding Pt atoms. While the ground state of Ho is a high spin state with total angular momentum  $J = 8$  for the free atom with spherical symmetry, this state splits

into the irreducible representations of  $C_{3v}$ , the symmetry group on the surface.  $C_{3v}$  is the symmetry group composed of 3-fold rotations about the Cartesian  $z$ -axis, and three vertical mirror planes. The group has six elements and three irreducible representations, denoted by  $A_1$  (trivial representation),  $A_2$ , and  $E$ . According to the Mulliken notation [4],  $A_1$  and  $A_2$  are 1-dimensional and  $E$  a 2-dimensional representation. The splitting of the 17-dimensional reducible representation of the  $J = 8$  state  $D^8$  is given by

$$D^8 \sim 3A_1 \oplus 2A_2 \oplus 6E. \quad (6)$$

The corresponding numerical values for the Ho energy levels on the surface are given in Figure 1 (c) and (d).

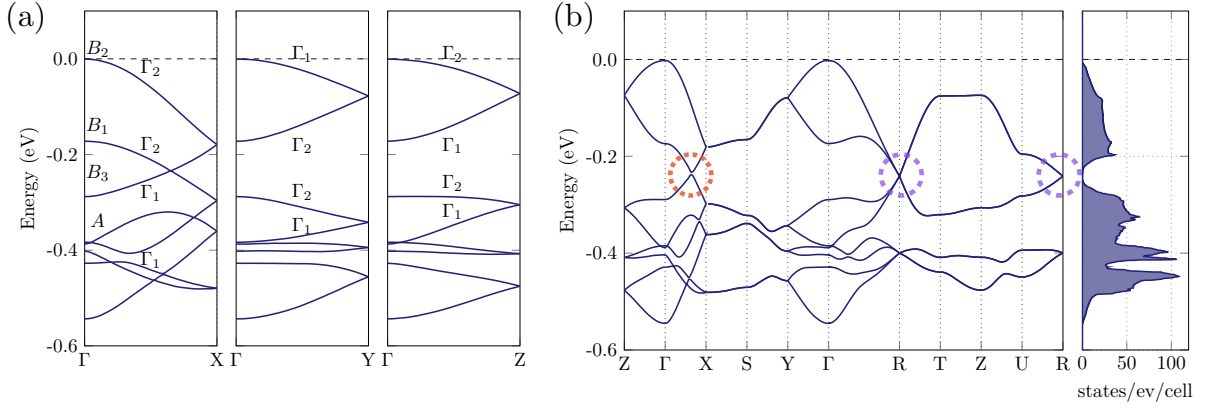
### Band structures

Crystals are periodic structures with a lattice spanned by the vectors  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ . The possible symmetry groups of 3-dimensional (nonmagnetic) crystals are the 230 crystallographic space groups [7]. The irreducible representations of space groups can be denoted by a combined index including the pseudo-momentum  $\mathbf{k}$  and an additional index running over inequivalent irreducible representations at  $\mathbf{k}$ .

Degenerate levels in band structures can emerge due to two mechanisms [8, 6]:

- **Degenerate levels protected by symmetry.** These are  $d$ -fold degenerate levels belonging to a  $d$ -dimensional irreducible representation.
- **Degenerate levels protected by band connectivity.** These are  $p + q$ -fold degenerate *accidental crossings* belonging to two inequivalent irreducible representations of dimensions  $p$  and  $q$ . They are enforced to occur due to continuous  $\mathbf{k}$ -dependence of the bands and the compatibility relations.

An example for the two kinds of crossings is shown in Figure 2. The band structure corresponds to an organic molecular crystal in space group #19. Here a four-fold



**Figure 2:** Electronic band structure of an organic molecular crystal. In the band structure (b), two different kinds of crossings can be found. Degenerate states due to higher dimensional irreducible representations (protected by symmetry) are highlighted in purple. Degenerate states protected by the connectivity of the bands are highlighted in red. (a) shows the dispersion of the electronic band structure together with the corresponding irreducible representations along three paths in the Brillouin zone. While bands belonging to the same representation anti-cross (i.e., form a gap), representations of different representations cross. Therefore, the crossing protected by the band connectivity can not be removed by swapping the 4 relevant electronic states at  $\Gamma$  ( $A, B_1, B_2, B_3$ ). Plots are taken from Ref. [6].

degenerate irreducible representation at the  $R$  point on the Brillouin zone boundary leads to a double Dirac crossing (Figure 2 (b)). Additionally, a Dirac crossing can be seen at the path  $\bar{\Gamma}\bar{X}$ , which is protected by the connectivity of one-fold degenerate bands emerging at  $\Gamma$  and two-fold degenerate bands in  $X$  (Figure 2 (a)).

In the past years, the topological properties of band structures have been intensively investigated. Of particular interest are: topological insulators [9], i.e., materials which are insulating in the bulk, but conducting on the surface; Dirac and Weyl semimetals [10, 11], i.e., materials where low-energy excitations behave as massless fully relativistic fermions; line node semimetals [12, 8], i.e., materials where bands cross in a (closed) line at the Fermi level; systems hosting exotic fermionic excitations [13, 14], e.g., due to tripple band crossings; higher order topological materials, exhibiting non-trivial states at crystal edges or corners [15, 16]. Computer algebraic methods can be used to scan through the massive phase space of allowed Hamiltonians in the 230 space groups. The irreducible representations of space groups can be calculated, e.g., using the Bilbao crystallographic server [17] or GTPack [3, 4].

## Effective Hamiltonians

Electrons are fermions. They occupy allowed states following the Pauli principle. At zero temperature, all states up to the Fermi level are occupied. Typical band widths in materials are in the order of a few eV. However, 1 meV corresponds to a temperature of 10 K. As a consequence, only a tiny regime of the band structure is of physical relevance at low temperatures. Therefore, it often suffices to describe a material in terms of an effective Hamiltonian. This can be done in two ways: i) one starts with an actual band structure and performs a low energy expansion around a  $\mathbf{k}$ -point of interest; ii) one starts with the symmetry of the material and generates

a symmetry constraint Hamiltonian. Computer algebra can be applied for the latter. Codes allowing for generating effective Hamiltonians are e.g. QSYMM [18], GTPack [3, 4].

To outline the method, we consider a two-dimensional surface with  $C_{3v}$  symmetry. We take into account two bands belonging to two different spin-channels. In spin space, an  $SU(2)$  rotation matrix can be expressed as  $e^{i\frac{\phi}{2}\sigma \cdot \mathbf{n}}$ , where  $\phi$  is the rotation angle and  $\mathbf{n}$  the rotation axis. Hence, we obtain

$$C_{3z} = \begin{pmatrix} e^{i\frac{\pi}{3}} & 0 \\ 0 & e^{-i\frac{\pi}{3}} \end{pmatrix} \quad (7)$$

for the counterclockwise three-fold rotation about the  $z$ -axis. Spin is an axial vector similar to the angular momentum and therefore is not affected by inversion. Hence, mirror operations in spin space are given by ordinary rotation matrices. For the case of a reflection  $M = IC_{2y}$  one obtains

$$M = C_{2y} = i\sigma_y. \quad (8)$$

Furthermore, we assume time-reversal symmetry, with the time-reversal operation  $T = i\sigma_y K$ .  $K$  is the complex conjugation. From the form of the matrices and the corresponding rotations in the 2-dimensional  $\mathbf{k}$ -space, we can summarize the action of the symmetry elements on  $\mathbf{k}$  and  $\sigma$  as,

$$C_{3z} : k_{\pm} \rightarrow e^{\pm i\frac{2\pi}{3}} k_{\pm}; \quad (9)$$

$$\sigma_{\pm} \rightarrow e^{\pm i\frac{2\pi}{3}} \sigma_{\pm}; \sigma_z \rightarrow \sigma_z \quad (10)$$

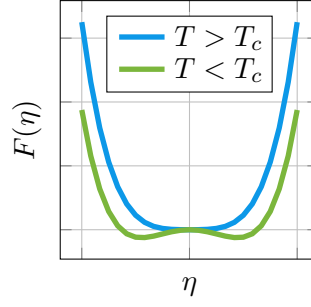
$$M : k_{\pm} \rightarrow -k_{\mp}; \quad (11)$$

$$\sigma_{x,z} \rightarrow -\sigma_{x,z}; \sigma_z \rightarrow \sigma_z \quad (12)$$

$$T : \mathbf{k} \rightarrow -\mathbf{k}; \sigma \rightarrow -\sigma. \quad (13)$$

We use the abbreviations  $k_{\pm} = k_x \pm ik_y$  and  $\sigma_{\pm} = \sigma_x \pm i\sigma_y$ . The Hamiltonian is invariant under all the symmetries of  $C_{3v}$  and time-reversal symmetry. A generic





**Figure 3:** Phenomenological theory of phase transitions with order parameter  $\eta$ . Below the critical temperature  $T < T_c$ , the free energy has minima for  $\eta \neq 0$ . As a result, a system undergoes the transition into an ordered phase.  $\eta$  breaks the symmetry.

$2 \times 2$  Hamiltonian can be written as

$$H(\mathbf{k}) = \epsilon(\mathbf{k}) + \begin{pmatrix} h(\mathbf{k}) & g(\mathbf{k}) \\ g^*(\mathbf{k}) & -h(\mathbf{k}) \end{pmatrix}. \quad (14)$$

We require  $h^*(\mathbf{k}) = h(\mathbf{k})$  for the Hamiltonian to be Hermitian. Furthermore, the symmetry operations constrain the Hamiltonian to satisfy the following conditions,

$$TH(\mathbf{k})T^{-1} = H(-\mathbf{k}), \quad (15)$$

$$C_3H(k_{\pm})C_3^{-1} = H(e^{\pm i\frac{2\pi}{3}}k_{\pm}), \quad (16)$$

$$MH(k_{\pm})M^{-1} = H(-k_{\mp}). \quad (17)$$

Time-reversal symmetry leads to  $h(-\mathbf{k}) = -h(\mathbf{k})$  and  $g(-\mathbf{k}) = -g(\mathbf{k})$ . The three-fold rotational symmetry gives  $h(k_{\pm}) = h(e^{\mp i\frac{2\pi}{3}}k_{\pm})$  and  $e^{\pm i\frac{2\pi}{3}}g(k_{\pm}) = g(e^{\mp i\frac{2\pi}{3}}k_{\pm})$ , and, last but not least, the mirror symmetry enforces  $-h(k_{\pm}) = h(-k_{\mp})$  and  $-g^*(k_{\pm}) = g(-k_{\mp})$ . To obtain an explicit expression for  $h(\mathbf{k})$  and  $g(\mathbf{k})$  we expand both functions in powers of  $k_{\pm}$ ,  $h(\mathbf{k}) = a_0 + a_1k_+ + a_2k_- + a_3k_+^2 + \dots$ , and  $g(\mathbf{k}) = b_0 + b_1k_+ + b_2k_- + b_3k_+^2 + \dots$ . Enforcing the conditions above, we deduce  $h(\mathbf{k}) = \frac{\lambda}{2}(k_+^3 + k_-^3)$  and  $g(\mathbf{k}) = -ik_+$ . The resulting low-energy expansion is

$$H(\mathbf{k}) = \epsilon(\mathbf{k}) + v_D(\sigma_x k_y - i\sigma_y k_x) + \frac{\lambda}{2}(k_+^3 + k_-^3) + \dots \quad (18)$$

Here, the first term gives the linear dispersion of the Dirac cone and the corresponding spin-texture. The second term induces hexagonal warping effects. Such hexagonal warping effects can be seen experimentally for the surface states of topological insulators [19].

## Phase transitions

According to the Landau-Ginzburg-Wilson theory of phase transitions, a phase is associated to an order parameter introducing a symmetry breaking. For example, the transition into a ferromagnet with finite magnetization breaks time-reversal symmetry. The transition of a normal metal into a superconductor breaks  $U(1)$  gauge symmetry. The transition into a ferroelectric material breaks inversion symmetry.

We consider a symmetry group  $\mathcal{G}$  with irreducible representations  $D^p$  of dimension  $d_p$ . The multicomponent order parameter  $\eta_i^p$ ,  $i = 1, \dots, d_p$  is a set of basis functions transforming like  $D^p$ , i.e.,  $\forall g \in \mathcal{G} : g\eta_i = \sum_j D_{ji}^p(g)\eta_j$ . The free energy  $F$  is a scalar function of all present order parameters.  $F$  is invariant under all transformations  $g \in \mathcal{G}$  and can be written as a polynomial in the  $\eta_i^p$ .

For example, the simplest case is a scalar order parameter  $\eta$  with the following free energy (see Figure ),

$$F(\eta) = a(T - T_c)\eta^2 + b\eta^4, \quad a, b, > 0. \quad (19)$$

Here,  $T$  is the temperature. The phase transition takes place at the critical temperature  $T_c$ . In terms of (19), for  $T > T_c$  the minimum of the free energy is given for  $\eta = 0$ . This regime is denoted as the normal phase. Below the critical temperature,  $T < T_c$ , the free energy has the form of a double well potential with minima at  $\eta = \pm\eta_0$ . Hence, the system minimizes the free energy by establishing an ordered phase characterized by  $\eta$ .

More complex expressions can be obtained for multicomponent order parameters. For example, a ferroelectric is characterized by a nonzero polarization vector  $\mathbf{P}$ . In a cubic normal phase, the free energy takes the following form up to fourth order

$$F(\mathbf{P}) = \alpha_1 \mathbf{P}^2 + \alpha_2 (P_x P_y + P_x P_z + P_y P_z) + \beta (P_x^4 + P_y^4 + P_z^4). \quad (20)$$

The complexity of expressions for the free energy increases even further by lowering the symmetry of the normal phase or by coupling various order parameters describing different phenomena, e.g., coexistence or competition of ferroelectricity and superconductivity [20].

Computer algebra can be applied to automatically generate expressions for the free energy. Algorithmically, this is achieved by modeling the space of all relevant order parameters by the reducible representation  $D = D^{p_1} \oplus D^{p_2} \oplus \dots$ . At each order  $n$ , we generate the representation

$$D^{\otimes n} = \underbrace{D \otimes D \otimes \dots}_{n \text{ times}}. \quad (21)$$



$D^{\otimes n}$  is reducible and can be written in terms of the irreducible representations  $D^p$ ,

$$D^{\otimes n} \sim n_1 D^1 \oplus n_2 D^2 \oplus \dots \quad (22)$$

$D^1$  denotes the identity representation. The free energy  $F$  transforms as  $D^1$ . Therefore, we need to generate linearly independent terms of the  $n_1$  products of  $\eta_j^{p_i}$  which transform as  $D^1$ . The  $n_1$  basis functions transforming as  $D^1$  are constructed using Clebsch-Gordan coefficients. The Clebsch-Gordan coefficients mediate the similarity transformation between the reducible representation  $D^{\otimes n}$  and its block diagonal form. Clebsch-Gordan coefficients can be calculated, e.g., using the algorithm of van Den Broek and Cornwell [21].

---

### Intertwined order

---

Complex materials can exhibit a rich phase diagram with several phase boundaries. By now it has been understood that fluctuations in one phase can mediate the mechanism to stabilize another phase. For example, fluctuations around the quantum critical point to a ferroelectric transition can mediate superconductivity [22]. Furthermore, instead of considering several phases as competing independent states, many cases are known with intertwined order [23, 24]. In such materials, neither of the two phases  $\eta_i^p$  nor  $\eta_j^q$  are present, but instead the composite order  $\eta_i^p \eta_j^q$ . While  $\eta_i^p$  and  $\eta_j^q$  are basis functions of the irreducible representation  $D^p$  and  $D^q$ , the composite order parameter  $\eta_i^p \eta_j^q$  transforms as the generally reducible representation  $D^{p \otimes q} = D^p \otimes D^q$ . This representation can be decomposed into the irreducible representations

$$D^{p \otimes q} \sim n_1 D^1 \oplus n_2 D^2 \oplus \dots \quad (23)$$

Each of the irreducible representations on the right-hand side represent an independent composite order parameter. While the basis functions of the occurring irreducible representations have the same properties as the single phases discussed in the previous section, the observable physical phenomena strongly depend on the composition of the parent phases. Exploring and classifying allowed composite phases in the 230 space groups is a tedious task which can be automatized using the methods of computer algebra.

---

### Conclusion and outlook

---

The properties of matter are determined from a complex interacting quantum many-body problem. Computer algebra can be applied to formulate effective theories or to classify and discuss allowed solutions. Symmetry is the key ingredient for formulating, e.g., an effective Hamiltonian or free energy expansion.

Several branches of condensed matter physics rely on algebraic methods. For example, with the discovery of topological properties of matter in the 1980s

[25, 26, 27] and the experimental realization of topological insulators in the 2000s [28], huge interest has emerged in classifying electronic band structures in 2- and 3-dimensional solids for the past two decades. Recently, this effort was extended, e.g.: i) to dimensions  $> 3$ , due to the correspondence of higher dimensional periodic lattices to 2- and 3-dimensional quasiperiodic structures [29]; ii) space- and time-periodic systems, i.e., Floquet-Bloch spectra [30].

Time-dependence also plays a significant role in discovering novel time-dependent or non-equilibrium orders, e.g., time crystals [31, 32], odd-frequency pairing [33], transient superconductivity [34]. The complexity of non-equilibrium phenomena clearly calls for novel algebraic techniques.

### References

- [1] Carl M Bender and Stefan Boettcher. Real spectra in non-Hermitian Hamiltonians having  $\mathcal{PT}$  symmetry. *Physical Review Letters*, 80(24):5243, 1998.
- [2] Yuto Ashida, Zongping Gong, and Masahito Ueda. Non-hermitian physics. *Advances in Physics*, 69(3):249–435, 2020.
- [3] R. M. Geilhufe and Wolfram Hergert. GTPack: A Mathematica Group Theory Package for Application in Solid-State Physics and Photonics. *Frontiers in Physics*, 6:86, 2018.
- [4] Wolfram Hergert and R. M. Geilhufe. *Group Theory in Solid State Physics and Photonics: Problem Solving with Mathematica*. Wiley-VCH, 2018. ISBN: 978-3-527-41133-7.
- [5] Toshio Miyamachi, Tobias Schuh, Tobias Märkl, Christopher Bresch, Timofey Balashov, Alexander Stöhr, Christian Karlewski, Stephan André, Michael Marthaler, Martin Hoffmann, et al. Stabilizing the magnetic moment of single holmium atoms by symmetry. *Nature*, 503(7475):242–246, 2013.
- [6] R Matthias Geilhufe, Stanislav S Borysov, Adrien Bouhon, and Alexander V Balatsky. Data mining for three-dimensional organic Dirac materials: focus on space group 19. *Scientific reports*, 7(1):1–7, 2017.
- [7] Carolyn Pratt Brock, T Hahn, H Wondratschek, U Müller, U Shmueli, E Prince, A Authier, V Kopskỳ, DB Litvin, E Arnold, et al. International tables for crystallography volume A: Space-group symmetry, 2016.
- [8] R Matthias Geilhufe, Adrien Bouhon, Stanislav S Borysov, and Alexander V Balatsky. Three-dimensional organic Dirac-line materials due to nonsymmorphic symmetry: A data mining approach. *Physical Review B*, 95(4):041103, 2017.
- [9] M Zahid Hasan and Charles L Kane. Colloquium: topological insulators. *Reviews of Modern Physics*, 82(4):3045, 2010.

- [10] Tim O Wehling, Annica M Black-Schaffer, and Alexander V Balatsky. Dirac materials. *Advances in Physics*, 63(1):1–76, 2014.
- [11] NP Armitage, EJ Mele, and Ashvin Vishwanath. Weyl and Dirac semimetals in three-dimensional solids. *Reviews of Modern Physics*, 90(1):015001, 2018.
- [12] A. A. Burkov, M. D. Hook, and Leon Balents. Topological nodal semimetals. *Physical Review B*, 84:235126, Dec 2011.
- [13] Barry Bradlyn, Jennifer Cano, Zhijun Wang, MG Vergniory, C Felser, Robert Joseph Cava, and B Andrei Bernevig. Beyond Dirac and Weyl fermions: Unconventional quasiparticles in conventional crystals. *Science*, 353(6299), 2016.
- [14] Zhijun Wang, Aris Alexandradinata, Robert J Cava, and B Andrei Bernevig. Hourglass fermions. *Nature*, 532(7598):189–194, 2016.
- [15] Flore K Kunst, Guido van Miert, and Emil J Bergholtz. Lattice models with exactly solvable topological hinge and corner states. *Physical Review B*, 97(24):241405, 2018.
- [16] Sayed Ali Akbar Ghorashi, Tianhe Li, and Taylor L. Hughes. Higher-Order Weyl Semimetals. *Physical Review Letters*, 125:266804, Dec 2020.
- [17] Mois I Aroyo, Asen Kirov, Cesar Capillas, JM Perez-Mato, and Hans Wondratschek. Bilbao crystallographic server. ii. representations of crystallographic point groups and space groups. *Acta Crystallographica Section A: Foundations of Crystallography*, 62(2):115–128, 2006.
- [18] Dániel Varjas, Tómas Ö Rosdahl, and Anton R Akhmerov. Qsymm: Algorithmic symmetry finding and symmetric Hamiltonian generation. *New Journal of Physics*, 20(9):093026, 2018.
- [19] Zhanybek Alpichshev, J. G. Analytis, J.-H. Chu, I. R. Fisher, Y. L. Chen, Z. X. Shen, A. Fang, and A. Kapitulnik. STM Imaging of Electronic Waves on the Surface of  $\text{Bi}_2\text{Te}_3$ : Topologically Protected Surface States and Hexagonal Warping Effects. *Physical Review Letters*, 104:016401, Jan 2010.
- [20] Joseph L. Birman and Meir Weger. Theory of co-existence of superconductivity and ferroelectricity: A dynamical symmetry model. *Physical Review B*, 64:174503, Oct 2001.
- [21] P. M. van Den Broek and J. F. Cornwell. Clebsch-gordan coefficients of symmetry groups. *physica status solidi (b)*, 90(1):211–224, 1978.
- [22] Quantum critical origin of the superconducting dome in  $\text{SrTiO}_3$ , author=Edge, Jonathan M and Kedem, Yaron and Aschauer, Ulrich and Spaldin, Nicola A and Balatsky, Alexander V. *Physical Review Letters*, 115(24):247002, 2015.
- [23] Rafael M Fernandes, Peter P Orth, and Jörg Schmalian. Intertwined vestigial order in quantum materials: Nematicity and beyond. *Annual Review of Condensed Matter Physics*, 10:133–154, 2019.
- [24] Eduardo Fradkin, Steven A Kivelson, and John M Tranquada. Colloquium: Theory of intertwined orders in high temperature superconductors. *Reviews of Modern Physics*, 87(2):457, 2015.
- [25] K. v. Klitzing, G. Dorda, and M. Pepper. New Method for High-Accuracy Determination of the Fine-Structure Constant Based on Quantized Hall Resistance. *Physical Review Letters*, 45:494–497, Aug 1980.
- [26] R. B. Laughlin. Quantized hall conductivity in two dimensions. *Physical Review B*, 23:5632–5633, May 1981.
- [27] D. J. Thouless, M. Kohmoto, M. P. Nightingale, and M. den Nijs. Quantized Hall Conductance in a Two-Dimensional Periodic Potential. *Physical Review Letters*, 49:405–408, Aug 1982.
- [28] YL Chen, James G Analytis, J-H Chu, ZK Liu, S-K Mo, Xiao-Liang Qi, HJ Zhang, DH Lu, Xi Dai, Zhong Fang, et al. Experimental realization of a three-dimensional topological insulator,  $\text{Bi}_2\text{Te}_3$ . *Science*, 325(5937):178–181, 2009.
- [29] Yaacov E Kraus, Yoav Lahini, Zohar Ringel, Mor Verbin, and Oded Zilberberg. Topological states and adiabatic pumping in quasicrystals. *Physical Review Letters*, 109(10):106402, 2012.
- [30] Alvaro Gómez-León and Gloria Platero. Floquet-bloch theory and topology in periodically driven lattices. *Physical Review Letters*, 110(20):200403, 2013.
- [31] Krzysztof Sacha and Jakub Zakrzewski. Time crystals: a review. *Reports on Progress in Physics*, 81(1):016401, 2017.
- [32] Xiao Mi, Matteo Ippoliti, Chris Quintana, Amy Greene, Zijun Chen, Jonathan Gross, Frank Arute, Kunal Arya, Juan Atalaya, Ryan Babbush, et al. Observation of Time-Crystalline Eigenstate Order on a Quantum Processor. *arXiv:2107.13571*, 2021.
- [33] Jacob Linder and Alexander V Balatsky. Odd-frequency superconductivity. *Reviews of Modern Physics*, 91(4):045005, 2019.
- [34] Daniele Fausti, RI Tobey, Nicky Dean, Stefan Kaiser, A Dienst, Matthias C Hoffmann, S Pyon, T Takayama, H Takagi, and Andrea Cavalleri. Light-induced superconductivity in a stripe-ordered cuprate. *Science*, 331(6014):189–191, 2011.



# Classification of Combinatorial Objects with Orbiter

Anton Betten (Colorado State University)

Anton.Betten@colostate.edu

---

## Introduction

Orbiter [7] is a computer algebra system devoted to the classification of combinatorial objects. For our purposes, combinatorial object means element of a set with a fixed group action. Two objects are equivalent (or isomorphic) if they belong to the same orbit under the group. The classification problem is the problem of determining the orbits of the group action, for instance by listing one representative from each orbit. This is different from enumerative combinatorics, where the goal is to determine the number of orbits, for instance using enumerative formulae. In most cases, construction of orbit representatives includes determining the stabilizer groups of the orbit representatives. Using the orbit-stabilizer lemma (orbit length equals index of stabilizer in the group that acts), we can then determine the number of objects total. This is often useful as a way to double check the classification, for instance when an enumerative formula is known for the total number of objects.

Related problems that fall within the scope of this research are the following. The isomorphism problem: Given two objects of the same type, determine whether or not they belong to the same orbit. If so, determine a group element that maps one to the other. The automorphism group problem: Given one object, determine the automorphism group. The recognition problem: Given an object, identify the unique element in a previously constructed transversal that is isomorphic to it.

The purpose of this article is to introduce the computer algebra system Orbiter which is devoted to the classification of combinatorial objects. We wish to illustrate how Orbiter can be used to solve these problems for some types of objects. For the sake of space, we restrict our focus to a problem from algebraic geometry, namely the classification of cubic surfaces with 27 lines (cf. [6]). A further application is the classification of smooth quartic curves (cf. [10]). Many further applications can be found in the User's Guide. We will discuss how Orbiter is installed and used. After

that, we will show how mathematical objects can be created, including groups, finite fields, and projective spaces. We will then turn to the problem of classifying cubic surfaces with 27 lines over a finite field. Finally, we will comment on some computer graphics capabilities, mathematical databases, and the use of Orbiter in parallel computing.

---

## What is Orbiter?

Orbiter is an open source software package written in C++. It can be used as class library or as a standalone computer algebra system. Orbiter does not have an interactive shell interface. Instead, Orbiter takes commands from the command line. Command sequences can be collected in unix shell scripts or makefiles. Orbiter offers functionality in group theory, in particular the field of finite permutation groups. Based on group theoretic algorithms, combinatorial objects can be classified. This article will illustrate some of the capabilities of Orbiter by showing examples of what it can do. Many further examples can be found in the User's guide, which can be found at [4]. A survey paper about Orbiter is [7].

Many researchers are working on the construction of various types of combinatorial objects. McKay's canonical augmentation procedure from [21] is very popular, as is his software package Nauty [20] (see also [22]). The book [19] is devoted to the classification problem for codes and designs. Many other contributions cannot be listed here for reasons of space.

Orbiter uses partially ordered sets with group action to classify combinatorial objects. For background, see [23]. For the algorithmic aspects, the work of Schmalz [27] and [26] is relevant ("Leiterspiel algorithm"). Before Orbiter, there was DISCRETA [8]. Before DISCRETA, there was Bernd Schmalz's program DCC ("double coset constructor") to compute double cosets in groups.

The goal of this document is to show how Orbiter can be used from the command line. We will not address the use of Orbiter as a C++ class library.

---

## Why do we need it?

---

Orbiter overlaps with some well-established computer algebra systems, in particular GAP [17] and Magma [12]. The goal is not to replace these systems but rather to offer something that cannot be done or that would be difficult to do in the existing systems. The classification of combinatorial objects is often difficult, and requires tedious computations, for which GAP or Magma may not be the best platform (see [7] for some case studies). Orbiter is trying to fill a gap (no pun intended!) where the existing systems become inefficient. We would also like to point out that there is considerable overlap with the GAP package fining [2]. Again, fining is not the system of choice when it comes to classification, so there is no conflict. In addition, Orbiter offers a way to collect mathematical data by means of a feedback loop. More on that in Section “Mathematical Data” below.

---

## Running Orbiter

---

There are two ways to run Orbiter: Native and Docker. Native means that Orbiter is compiled from scratch, using the source code from the github repository (cf. [5]). Docker [14] is a system to run preconfigured software in an encapsulated way on various platform, including Windows. We describe using Orbiter through unix *makefiles*, which are run through the tool *make* (cf. [16]). This is a software tool that allows collecting short command snippets in the form of text files that can easily be handled. However, the conventions in the tool involve some subtleties regarding the use of whitespace, which can cause problems to novice users. We will point out possible pitfalls along the way. Note that it is not necessary to use makefiles. Another possibility would be to use shell scripts. Ultimately, it would be possible to type out all commands into a terminal window. This could be a little tedious though, considering the fact that most Orbiter commands expect lengthy parameters from the command line.

Let us start by discussing how to run Orbiter as a native application. To do so, a unix-like compile environment is required, including a modern C++ compiler and the tools *git* and *make*. Windows users may need to install Cygwin [13]. The following steps are required: Using *git*, *clone* the repository. Then enter the directory *orbiter* and type

```
make
```

Once compiled, the Orbiter executable is

```
src/apps/orbiter/orbiter.out
```

within the Orbiter directory. We then recommend creating a separate work directory *not within the orbiter directory*. For the following, we assume the following directory tree structure:

```
└─ orbiter
   └─ work
```

In the work directory, create a small makefile like so:

```
OP=../orbiter
ORBITER_PATH=$(OP)/src/apps/orbiter/

test:
    $(ORBITER_PATH)orbiter.out
```

Different directory structures can be accommodated by changing the first line. Next, typing

```
make test
```

within the work directory will invoke Orbiter. Here, *test* is the makefile “target.” The makefile target must appear in the makefile. In the example above, the block

```
test:
    $(ORBITER_PATH)orbiter.out
```

is the makefile target “test.” It is important that the indentation after the makefile target is done using tab characters (no spaces). There can be multiple targets in one makefile, as long as they are separated by an empty line. for more information about the syntax of makefiles, see [16].

A second way to run Orbiter is through Docker [14]. This does not require a compile environment. However, it comes at a small performance cost when running Orbiter commands that are computationally heavy. Orbiter has already been precompiled (by the Orbiter developer) into an *image*, which is a completely self-sustained copy of a unix-environment that can run by the user under the docker front-end. The image is stored on a docker server under the name *abetten/orbiter*. Docker will receive the name of the image from the command line, pull a local copy of the image, and run the image in an encapsulated environment called a *container*. A copy of the image is stored locally, so that subsequent calls to Orbiter can be satisfied using the local copy, which increases turnaround speed. For instance, the following bare-bones makefile sets up Orbiter for use through Docker:

```
DOCKER_OPTIONS=run -it \
    --volume ${PWD}:/mnt -w \
    /mnt abetten/orbiter
ORBITER_PATH=docker $(DOCKER_OPTIONS)

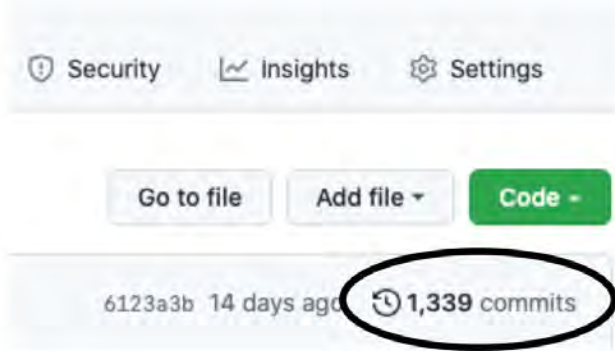
test:
    $(ORBITER_PATH)orbiter.out
```

In this file, there is a space character in line three after *abetten/orbiter* which is important (and unfortunately cannot be seen). By typing

```
make test
```

into a terminal window, Docker starts up and pulls a copy of Orbiter to the local machine, which is then executed. Orbiter will start up, produce a few messages and then shut down. Interestingly, this will work on a Windows machine also (using *supershell* as terminal). The *make* command is passed through to the container, which contains the unix-like software environment, including *make*. The associated *makefile* resides on the local machine, as do input and output files.

Orbiter comes with a version numbering system called a build number. The build number should match the commit number on the github tree, shown in Figure 1.



**Figure 1:** The commit number

When Orbiter starts up, the build number is displayed. In order to update to a more recent version of Orbiter, Docker needs to be instructed to discard the local image. To do so, the command

```
docker rmi -f abetten/orbiter
```

is used. After that, any new invocation of Orbiter will cause Docker to pull the latest Orbiter *image* from the Docker repository. It is convenient to combine the Docker and Native compile environment into a single makefile and use the comment symbol (hash #) to switch between the two modes (the line numbers are not part of the file).

```
1 OP=~/orbiter
2 OP2=$(OP)/src/apps/orbiter/
3 DOCKER_OPTS=run--it.\
4 ---volume:${PWD}:/mnt--w.\
5 ../mnt.abetten/orbiter.
6 #ORBITER_PATH=docker.${DOCKER_OPTS}
7 ORBITER_PATH=$(OP2)
```

Here, whitespace characters can be seen: (spaces are shown as dots, and tab is a little triangle pointing to the right). Please observe the space at the end of line 5 and that the line(s) after the target(s) must start with a tab symbol (and no spaces). Also, the backslash signs are used to break long lines. Please make sure that there are no spaces after the backslash sign. In order to switch to Docker mode, the hash symbol can be removed in line 6

and instead put at the beginning of line 7. In the following examples, we assume that the 7 lines just shown are present at the beginning of the makefile. For brevity, we will only show the commands and their labels. These snippets must come after the top part.

## Group Theory

Orbiter distinguishes between a group and the group action. One group can have different actions. Every group must have at least one action. Let us look at an example: The following makefile command is used to create the group PTL(3, 8).

```
PGGL_3_8:
> $(ORBITER_PATH)orbiter.out--v.3.\
> > -define.G--linear_group.\
> > -PGGL.3.8--end
```

The command PGGL refers to the group PTL, and the two parameters afterwards are the dimension and the order of the field of coefficients. Not much happens! This is because though we told Orbiter to create the group, we did not ask that anything be done with it. Orbiter simply creates the group and exits. Note that Orbiter creates an object of type group under the name *G*. Using this label, it is possible to do something with the group later. The syntax is

```
--with LABEL -do ...--end
```

where LABEL is the label of any Orbiter object (such as *G* in case of the group PTL(3, 8)). Suppose we want to do something with the group. To this end, we will use a group theoretic activity. One possibility is to print a latex report. The next command does just that:

```
PGGL_3_8_report:
> $(ORBITER_PATH)orbiter.out--v.3.\
> > -define.G--linear_group.\
> > > -PGGL.3.8--end.\
> > -with.G--do.\
> > -group_theoretic_activities.\
> > > -report.\
> > -end
> pdflatex.PGGL_3_8_report.tex
> open.PGGL_3_8_report.pdf
```

The Orbiter command produces a report in the form of a latex file. The file name is generated automatically based on the group that was created. Next, the pdflatex command is used to translate the latex file. The open command is Macintosh specific and opens the pdf file in this case. The last two commands may have to be replaced by the appropriate local latex specific commands.

## Indexing

Orbiter can create many types of permutation groups. Each of these groups can be considered in many actions. Starting from basic groups with their standard actions, new actions can be induced or combined or otherwise



modified to create new actions. Basic groups often come with a natural permutation action. This could be the action on points of an underlying projective space, or an affine space, or an orthogonal space, a hermitian space etc. In order to create these basic actions, the permutation representation needs to be established. To this end, Orbiter uses fixed bijections between the space on which we act and the integer interval  $0, \dots, N - 1$  where  $N$  is the degree of the action. In GAP [17], this technique is realized by enumerators. In Magma [12], it is called indexing. The bijections selected for indexing are fixed once and for all. For objects in projective spaces over finite fields which are not prime fields, this requires the choice of a polynomial to create the field. Orbiter has built-in polynomials, stored in the form of tables. This guarantees that the results from one computation can be repeated at any time later and do not change. It is however possible to override the polynomial used to create extension fields. This allows compatibility with other computer algebra systems. Indexing exists for many types of objects. Besides points in projective space, Orbiter has indexing for the grassmannian, orthogonal spaces, tensor spaces, direct products, the wedge product, and many other objects.

Let us consider the example from above. In order to create the group  $\text{PTL}(3, 8)$ , Orbiter first establishes indexing for the 73 points of  $\text{PG}(2, 8)$ . It is possible to produce a latex report which shows the bijection. Because the field  $\mathbb{F}_8$  is an extension field, we first need to enumerate the elements in the field  $\mathbb{F}_8$ . To this end, we pick an irreducible polynomial and represent each field element as polynomial with coefficients over  $\mathbb{F}_2$ . For instance, Orbiter picks  $X^3 + X^2 + 1$  over  $\mathbb{F}_2$  to create  $\mathbb{F}_8$ . Next, we use the binary representation of numbers to map field elements to integers in the interval  $0, \dots, 7$  like so:

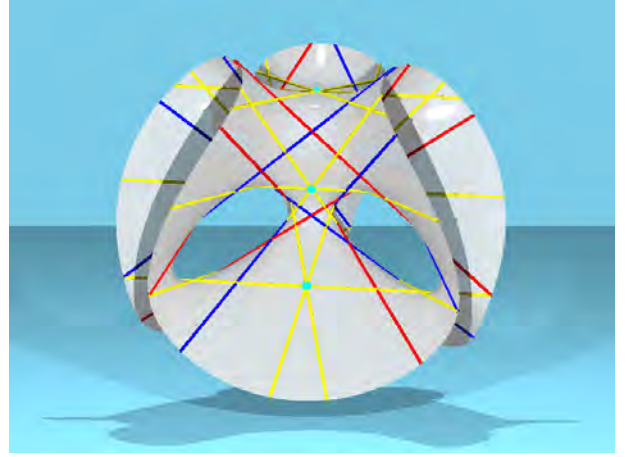
0	000	0
1	001	1
2	010	$\alpha$
3	011	$\alpha + 1$
4	100	$\alpha^2$
5	101	$\alpha^2 + 1$
6	110	$\alpha^2 + \alpha$
7	111	$\alpha^2 + \alpha + 1$

In order to index the points of projective space, Orbiter uses a variant of the lexicographic ordering. Namely, the points of the standard frame are given preference over all other points. In  $\text{PG}(n, q)$ , a frame has size  $n + 2$  and is given the ranks  $0, \dots, n$ . The remaining elements are grouped by the number of trailing zeros and then enumerated lexicographically, assuming that the last non-zero element is one.

## Algebraic Geometry

A cubic surface is an algebraic variety of degree three in  $\text{PG}(3, \mathbb{F})$  (the three-dimensional projective space over the field  $\mathbb{F}$ ). One of the first observations about cubic surfaces is that over an algebraically closed field, a

smooth surface has exactly 27 lines on them. It is possible that the 27 lines can be obtained over fields that are not algebraically closed, but this is the exception and not the rule. An example is shown in Figure 2, which shows the Clebsch surface over the real numbers in an affine chart.



**Figure 2:** *The Clebsch surface*

In this example, all 27 lines are real, but only 24 are visible. This is because three lines lie in the plane at infinity. Another example with all 27 lines real is the Eckardt surface. The coloring of the lines is done according to a Schläfli double six. The six red lines and the six blue lines represent the double six. The 15 yellow lines (3 are at infinity) represent the 15 “diagonal” lines. It is possible that three of the lines are concurrent in a single point. Such a point is called an Eckardt point. The Clebsch surface is characterized by the fact that it has exactly 10 Eckardt points (shown in turquoise in the picture, but not all are visible in the affine chart). The number of Eckardt points is between 0 and 45, but not every number in the interval is possible. Besides the Clebsch surface, there are many other surfaces. Two surfaces are equivalent (or isomorphic) if there is a collineation of the projective space which maps one to the other. It is of interest to determine how many nonisomorphic cubic surfaces with 27 lines exist over a given field, in particular over any finite field. Let us consider an example. In order to classify the cubic surfaces with 27 lines over  $\mathbb{F}_{13}$ , we use Orbiter with the following makefile command:

```
surface_classify_q13:
▷ $(ORBITER_PATH)orbiter.out.-v.5.\
▷ -define.F--finite_field.\
▷ -q.13--end.\
▷ -define.P--projective_space.\
▷ -3.F--end.\
▷ -with.P--do.\
▷ -projective_space_activity.\
▷ -classify_surfaces_with.\
double_sixes.Surf27.-W.-end.\
▷ -end.\
▷ -with.Surf27.-do.\
```

```

> > -classification_of_cubic\
surfaces_with_double_sixes\
activity\
> > > -report--end\
> > -end\
> > -print_symbols
> pdflatex·Surfaces_q13.tex
> open·Surfaces_q13.pdf

```

The computations show that there are exactly 4 isomorphism types of cubic surfaces with 27 lines over  $\mathbb{F}_{13}$ . They are the Eckardt surface with 6 Eckardt points, the Fermat surface with 18 Eckardt points, and two further surfaces with 9 and 4 Eckardt points, respectively. Detailed information about each surface can be found in the latex report (which has 265 pages). Let us take a quick look at the command. The command sequence defines three objects, called  $F$ ,  $P$  and  $Surf27$ . The objects are maintained in a symbol table. The line

```
-define F -finite_field -q 13 -end
```

creates the field  $\mathbb{F}_{13}$  as object  $F$ . The line

```
-define P -projective_space 3 F -end
```

creates the three-dimensional projective space over the field  $F$  and defines the object  $P$ . The sequence

```

-with P -do \
  -projective_space_activity \
  -classify_surfaces_with\
double_sixes Surf27 -W -end \
-end

```

classifies all cubic surfaces with 27 lines in the projective space  $P$  and defines the object  $Surf27$ . The sequence

```

-with Surf27 -do \
  -classification_of_cubic\
surfaces_with_double_sixes\
activity \
  -report -end \
-end

```

creates the latex report from the classification. The algorithm from [11] is used to classify cubic surfaces with 27 lines. This algorithm relies on the well-known relation between cubic surfaces with 27 lines and Schläfli double sixes (cf. [25]). The double sixes in turn are classified using smaller structures called five-plus-ones. These are five pairwise skew lines in  $PG(3, \mathbb{F})$  with a common transversal and with the property that none of the five lies in the regulus determined by three of the other four.

Consider the four surfaces over  $\mathbb{F}_{13}$ . The automorphism group orders and the number of Eckardt points are summarized below, together with the family name:

Surface	#Eckardt	Stab. Order	Name
0	4	12	Eckardt [15]
1	6	24	
2	9	108	
3	18	648	Fermat

The group  $PTL(4, 13)$  has order 50858076935877120. The orbit stabilizer lemma applied to the classification gives that the number of cubic surfaces with 27 lines over  $\mathbb{F}_{13}$  is

$$50858076935877120 \left( \frac{1}{12} + \frac{1}{24} + \frac{1}{108} + \frac{1}{648} \right)$$

= 6906652423390720. On the other hand, according to [18], the number of cubic surfaces with 27 lines over  $\mathbb{F}_q$  is

$$\frac{q^6(q^2 - 1)(q^3 - 1)(q^4 - 1)(q - 2)(q - 3)(q - 5)^2}{51840}.$$

For  $q = 13$ , this confirms the result from the classification.

---

## Computer Graphics

---

Through its interface to the ray-tracing software Povray [24], Orbiter can create three-dimensional graphics and animations. For instance, the code below creates the Kummer surface:

Kummer\_surface:

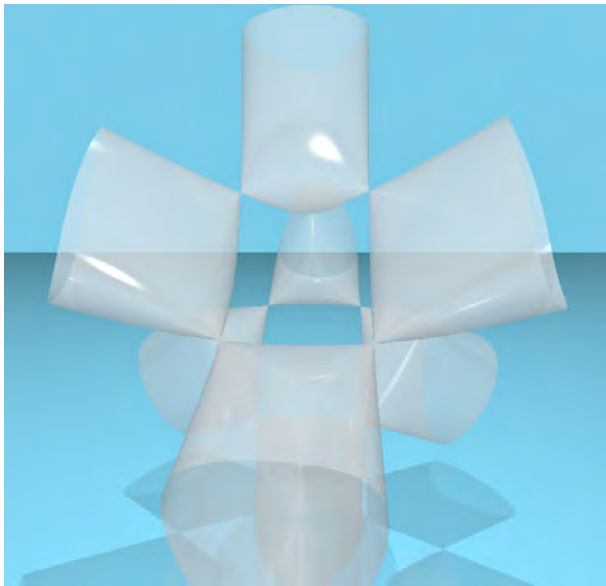
```

> $(ORBITER_PATH)orbiter.out--v.2\
> > -povray\
> > -round.0--nb_frames_default.30\
> > -output_mask_Kummer.%d.%03d.pov\
> > -video_options--W.1024--H.768\
> > -global_picture_scale.0.9\
> > -default_angle.75\
> > -clipping_radius.2.4\
> > -camera.0."1,1,1"-3,1,3"\
> > > "0.12,0.12,0.12""\
> > -end\
> > -scene_objects\
> > > -quartic_lex_35."-2,0,0,0,2,\
0,0,2,0,2,0,0,0,0,0,0,0,0,0,0,\
0,0,2,0,2,0,0,0,0,-2,0,2,0,-2"\
> > > -group_of_things."0"\
> > > -quartics.0."texture{\
pigment{White*0.5*transmit.0.5}\
finish{ambient.0.4*diffuse.0.5\
roughness.0.001*reflection.0.1\
specular.8}\}""\
> > -scene_objects_end\
> > -povray_end

```

The code prepares 30 different frames, showing the surface as it rotates along a vertical axis. The frames can be assembled to an animation using tools like ffmpeg [3]. The surface is defined using the coefficient vector. The monomials of degree 4 are arranged in lexicographic order. Exactly 35 coefficients are given after the `-quartic_lex_35` option. The remaining options are used to specify things like camera viewpoint

and surface color. An image created from this command is shown in Figure 3.



**Figure 3:** *The Kummer surface*

---

## Mathematical Data

The computations necessary to perform a classification involve potentially large numbers of CPU-cycles. For this reason, the mathematical data produced by Orbiter should be conserved for later use. One way to do so is by means of the feedback loop. The output of a classification is written to a C++ source file, which can then be compiled into a later version of Orbiter. This way, the classification becomes available instantaneous to later users of Orbiter. It is also possible to create web-databases of mathematical data. The mathdata project [9] is underway, which will allow access to the data by clicking through tables. At present, Orbiter has databases for cubic surfaces, quartic curves, BLT-sets, spreads and translation planes, packings, dual hyperovals, arcs, linear codes, irreducible polynomials over finite fields and others. For a more in-depth discussion of the feedback loop, see [7].

---

## High Performance Computing

Orbiter is well-suited for high performance computing using compute clusters. The Orbiter workflow is file based: There is a makefile, and there are input and output files to every job. This kind of workflow fits well with parallel processing on compute clusters, where little command scripts are used to start up jobs. These command scripts have information about the desired hardware configuration and about the job demands. The orbiter makefile commands go into the command scripts. We acknowledge use of the NSF funded machine Summit which is run jointly by Colorado State University and CU Boulder [1].

## References

- [1] Jonathon Anderson, Patric J. Burns, Daniel Milroy, Peter Ruprecht, Thomas Hauser, and Howard Jay Siegel. Deploying RMACCS Summit: An HPC Resource for the Rocky Mountain Region. In *Proceedings of PEARC 17, New Orleans, LA, USA, July 09-13, 2017*, 7 pages.
- [2] John Bamberg, Anton Betten, Philippe Cara, Jan De Beule, Michel Lavrauw, and Max Neunhöffer. *FinInG – Finite Incidence Geometry, Version 1.4.1*, 2018.
- [3] Fabrice Bellard. FFmpeg <https://www.ffmpeg.org>.
- [4] A. Betten. Orbiter: <https://www.math.colostate.edu/~betten/orbiter/>.
- [5] Anton Betten. Orbiter – A program to classify discrete objects, 2019, <https://github.com/abetten/orbiter>.
- [6] Anton Betten. Classifying cubic surfaces over finite fields using orbiter. In *ICMS 2018—Proceedings of the International Congress on Mathematical Software; James H. Davenport, Manuel Kauers, George Labahn, Josef Urban (ed.)*, pages 55–61. Springer, 2018.
- [7] Anton Betten. The orbiter ecosystem for combinatorial objects. In *ISSAC 2020—Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*, pages 30–37. ACM, New York, 2020.
- [8] Anton Betten, Evi Haberberger, Reinhard Laue, and Alfred Wassermann. *DISCRETA – A program system for the construction t-designs*. Lehrstuhl II für Mathematik, Universität Bayreuth, 1999. <http://www.mathe2.uni-bayreuth.de/~discreta>.
- [9] Anton Betten, Nathan Kaplan, Fatma Karaoglu, and Oznur Oztunc. MATHDATA <https://www.math.colostate.edu/~betten/mathdata/>.
- [10] Anton Betten and Fatma Karaoglu. Isomorphism Testing of Algebraic Varieties using Canonical Forms, In: *Computer Algebra in Scientific Computing, CASC 2021, September 13-17, 2021, Sochi, Russia, Sirius Mathematics Center*, [https://siriusmathcenter.ru/pr\\_img/1918100371/20210914/13241784/Program\\_010w.pdf](https://siriusmathcenter.ru/pr_img/1918100371/20210914/13241784/Program_010w.pdf), pp 14-27.
- [11] Anton Betten and Fatma Karaoglu. Cubic surfaces over small finite fields. *Des. Codes Cryptogr.*, 87(4):931–953, 2019.
- [12] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

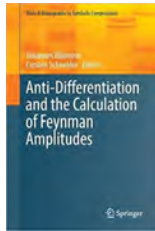
- [13] Cygwin. <https://www.cygwin.com>.
- [14] Docker. <https://www.docker.com>.
- [15] F.E. Eckardt. Ueber diejenigen Flächen dritten Grades, auf denen sich drei gerade Linien in einem Punkte schneiden, *Math. Ann.* 10 (1876), 227–272.
- [16] Free Software Foundation. GNU make <https://www.gnu.org/software/make/manual/make.html>.
- [17] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.8.7*, 2017.
- [18] Fatma Karaoglu and Anton Betten. The number of cubic surfaces with 27 lines over a finite field. To appear in *Journal of Algebraic Combinatorics*.
- [19] P. Kaski and P. Östergård. *Classification algorithms for codes and designs*, volume 15 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 2006.
- [20] Brendan McKay. Nauty and Traces (Version 2.7r1), Australian National University, 2020.
- [21] Brendan D. McKay. Isomorph-free exhaustive generation. *J. Algorithms*, 26(2):306–324, 1998.
- [22] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *J. Symbolic Comput.*, 60:94–112, 2014.
- [23] Wilhelm Plesken. Counting with groups and rings. *J. Reine Angew. Math.*, 334:40–68, 1982.
- [24] POV-RAY Developers. POV-RAY, Persistence of Vision Raytracer Pty. Ltd. <http://povray.org>, accessed 1/23/2021.
- [25] L. Schläfli. An attempt to determine the twenty-seven lines upon a surface of the third order and to divide such surfaces into species in reference to the reality of the lines upon the surface, *Quart. J. Math.* 2 (1858), 55–110.
- [26] B. Schmalz.  $t$ -Designs zu vorgegebener Automorphismengruppe. *Bayreuth. Math. Schr.*, 41:1–164, 1992. Dissertation, Universität Bayreuth, Bayreuth, 1992.
- [27] Bernd Schmalz. Verwendung von Untergruppenleitern zur Bestimmung von Doppelnebenklassen. *Bayreuth. Math. Schr.*, (31):109–143, 1990.

---

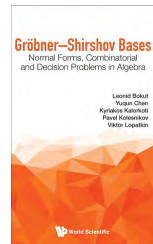
## Publikationen über Computeralgebra

---

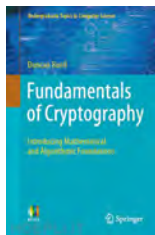
### Neuerscheinungen:



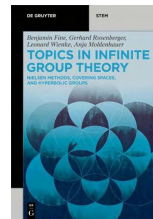
J. Blümlein, C. Schneider (Hrsg.),  
*Anti-Differentiation and the Calculation of Feynman Amplitudes*,  
Texts & Monogr. in Symb. Comp.,  
Springer Int. Publ., Cham 2021,  
640+x Seiten,  
ISBN 978-3-030-80218-9



L. Bokut, Y. Chen, K. Kalorkoti,  
P. Kolesnikov, V. Lopatkin,  
*Gröbner-Shirshov Bases*,  
World Scientific, Singapur 2021,  
285+xxii Seiten,  
ISBN 978-9-814-61948-6



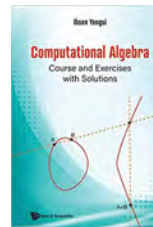
Duncan Buell,  
*Fundamentals of Cryptography*,  
*Math. and Alg. Foundations*,  
Springer Int. Publ., Cham 2021,  
279+xv Seiten,  
ISBN 978-3-030-73491-6



B. Fine, A. Moldenhauer,  
G. Rosenberger, L. Wienke,  
*Topics in Infinite Group Theory*  
de Gruyter, Berlin 2021,  
382+ix Seiten,  
ISBN 978-3-11-067334-0



Wolfram Koepf,  
*Computer Algebra*,  
Springer Int. Publ., Cham 2021,  
384+xii Seiten,  
ISBN 978-3-030-78016-6



Ihsen Yengui,  
*Computational Algebra*,  
World Scientific, Singapur 2021,  
282 Seiten,  
ISBN 978-9-811-23930-4

Die Rubrik Publikationen ist nicht allein auf eine Liste von Neuerscheinungen und Neuauflagen beschränkt. Sie lebt vor allem von fundierten Rezensionen von Fachgruppenmitgliedern für Fachgruppenmitglieder, die wir an dieser Stelle gerne abdrucken. Sollte eines der oben genannten Bücher, insbesondere eine der Neuerscheinungen, Ihr Interesse geweckt haben, und Sie möchten dieses für den Computeralgebra-Rundbrief besprechen, nehmen Sie bitte Kontakt zu Jürgen Klüners oder Martin Kreuzer ([klueners@math.uni-paderborn.de](mailto:klueners@math.uni-paderborn.de), [martin.kreuzer@uni-passau.de](mailto:martin.kreuzer@uni-passau.de)) auf.



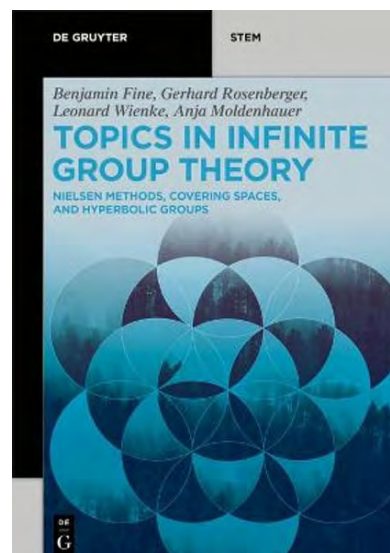
### Benjamin Fine, Anja Moldenhauer, Gerhard Rosenberger und Leonard Wienke Topics in Infinite Group Theory

Es mag auf den ersten Blick etwas verwunderlich erscheinen, warum hier eine Buchbesprechung eines Werkes erscheint, das ganz und gar in der theoretischen Gruppentheorie verankert zu sein scheint. Dennoch hat dieser Band wohl mehr algorithmischen Gehalt und mehr Relevanz für die praktische Implementation von Computeralgebrasystemen für die Gruppentheorie als so manches Werk, das sich mit Beiworten wie „algorithmisch“ oder „Gröbner-Basen“ ziert, jedoch für echte Implementationen und Berechnungen nur wenig Brauchbares bietet.

Das Buch ist in drei Kapitel gegliedert, die sich mit Nielsen Methoden, mit Überlagerungsräumen und mit hyperbolischen Gruppen beschäftigen. Besonders das erste und das dritte Kapitel sind dabei reich an effizienten und wichtigen Rechenverfahren für die algorithmische Gruppentheorie. Diese werden jedoch in der klassischen mathematischen Art und Weise mit „Definition - Satz - Beweis“ präsentiert, und nicht wie heutzutage gelegentlich anzutreffen mit Pseudocode und Pseudobeweisen. Der Vorteil für den praktischen Implementator ist, dass die dem Algorithmus unterliegende Mathematik klar, vollständig und korrekt eingeführt wird; der Nachteil ist, dass man die konkreten Schritte der Algorithmen oftmals aus den Beweisen extrahieren muss.

Betrachten wir nun die einzelnen Kapitel etwas genauer. Die sogenannte Nielsen-Methode, die das Thema des ersten Kapitels darstellt, basiert auf Nielsen Transformationen welche besonders einfache Automorphismen freier Gruppen darstellen. Mit ihnen kann man endliche Mengen von Wörtern in eine Nielsen reduzierte Form transformieren und zum Beispiel einfach zeigen, dass endlich erzeugte Untergruppen von freien Gruppen wieder frei sind. Weitere Anwendungen der Nielsen-Methode sind die Konstruktion von Testelementen, mit deren Hilfe man Gruppenhomomorphismen auf Bijektivität prüfen kann und der Beweis von Nielsen's Theorem, dass Automorphismen endlich erzeugter freier Gruppen endlich präsentierbar sind.

Weitere algorithmische Methoden in diesem Kapitel basieren auf Tietze Transformationen, der Reidemeister-Schreier Methode und der Todd-Coxeter Methode. Diese Techniken stellen die zentralen Algorithmen dar, mit deren Hilfe endlich präsentierte Gruppen algorithmisch untersucht werden können. Das Buch liefert dann auch eine ganze Reihe von solchen Anwendungen, die anderswo so kaum zu finden sind, z. B. freie Produkte mit Amalgamation, Gleichungen in HNN-Erweiterungen sowie spezielle Eigenschaften von Ein-Relator-Gruppen.



Das zweite Kapitel bildet die Brücke zur geometrischen Gruppentheorie und bietet mit den Fundamentalgruppen und Kantenpfadgruppen von CW-Komplexen wichtige Beispiele für die algorithmische Behandlung. Das dritte und letzte Kapitel betrifft hyperbolische Gruppen. Diese sind durch eine Eigenschaft der Cayley-Graphen definiert, die den Gegebenheiten in einem negativ gekrümmten Raum entsprechen. Vom algorithmischen Standpunkt sind sie sehr angenehm zu handhaben, da sie einen Dehn-Algorithmus besitzen, d. h. man kann eine Präsentation finden mit deren Hilfe das Wortproblem und viele weitere algorithmische Aufgaben effizient lösbar sind. Für solche Gruppen werden viele Beispiele konstruiert und klassische, aber auch neueste Resultate bewiesen, wobei die meisten dieser Beweise sehr algorithmisch ausfallen und zur expliziten Implementation einladen.

Insgesamt bietet das Buch einen tiefen und detaillierten Einblick in die algorithmischen Grundlagen der geometrischen und kombinatorischen Gruppentheorie, die von den klassischen Sätzen bis hin zu modernen Anwendungen (z. B. in der Zahlentheorie) und aktuellen Forschungsergebnissen reichen. Gerade im Bereich der hyperbolischen Gruppen existieren viele wichtige offene Vermutungen, die einem experimentellen Zugang über die hier beschriebenen algorithmischen Verfahren zugänglich erscheinen. Das Buch ist klar und gut lesbar geschrieben, wenn auch die Angabe, dass es für Studierende ab dem dritten Studienjahr geeignet sei, leicht optimistisch erscheint.

Martin Kreuzer (Passau)

# Leonid Bokut, Yuqun Chen, Kyriakos Kalorkoti, Pavel Kolesnikov und Viktor Lopatkin

## Gröbner-Shirshov Bases

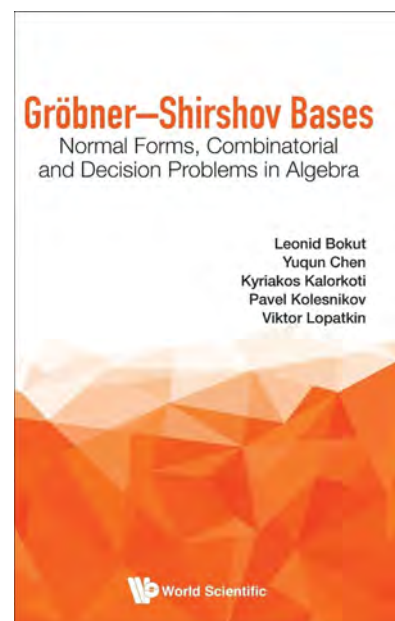
Ein bekanntes Sprichwort besagt „Der Erfolg hat viele Väter, der Misserfolg ist ein Waisenkind“. Das stimmt auch für viele mathematische Entdeckungen, und die Theorie der Gröbner-Basen ist ein gutes Beispiel. Diese Theorie kam so richtig in Fahrt mit der Dissertation [1] von Bruno Buchberger, in der er den nach ihm benannten Algorithmus erfand und auch die Bezeichnung „Gröbner-Basen“ einführte, und der Arbeit [2] über die Knuth-Bendix-Vervollständigung von Termersetzungssystemen.

Schon lange davor hatte es viele Hinweise auf die Existenz einer solchen Theorie gegeben: im Fall homogener Polynome 1916 durch Francis Macaulay [3], im Fall differentieller Polynome 1920 durch Maurice Janet [4], bezüglich der Frage der Berechenbarkeit bei Polynomidealen durch Grete Hermann 1926 in [5], in einer kurzen Notiz aus dem Jahr 1962 zum Fall von Lie-Algebren durch Anatoly Shirshov [6], und im Fall von Standardbasen für Potenzreihenideale durch Heisuke Hironaka in seiner berühmten Arbeit [7], um nur einige Vorläufer zu nennen. In einem langen, technischen Vorwort legen die Autoren dieses Buchs dar, dass die Theorie der Gröbner-Basen für nicht-kommutative Polynomringe (auch bekannt als freie assoziative Algebren) implizit in der dreiseitigen Arbeit von Shirshov enthalten sei und sie folgern daraus, dass alle Gröbner-Basen in nicht-kommutativen Ringen damit Gröbner-Shirshov-Basen heißen sollten. Wie es ein Kollege etwas überspitzt ausdrückte: der einzige Mathematiker, der die Gröbner-Basen anscheinend nicht erfunden hat, ist Wolfgang Gröbner.

Sei es wie es ist, für die Zwecke dieser Besprechung wollen wir dem in diesem Buch verwendeten Namen GS-Basen folgen. Neben der Darlegung des historischen Rahmens und der Beiträge von Shirshov enthält das Buch viel interessantes Material. Im ersten Kapitel beginnen die Autoren mit dem euklidischen Algorithmus für univariate Polynome und kommen über die Gaußsche Elimination für lineare Polynome zur Lösung polynomialer Gleichungssysteme mittels Buchbergers Algorithmus. Diese Tour de Force folgt dem wohlbekannten Pfad, der zum Beispiel sehr schön in dem Buch [8] von Joachim von zur Gathen dargelegt ist. Die hiesige Präsentation leidet allerdings unter gelegentlichen Schludrigkeiten: die Bedingung  $f = 0$  und  $g = 0 \Rightarrow F = 0$  ist natürlich *nicht* gleichbedeutend mit  $F \in \langle f, g \rangle$ .

Im zweiten Kapitel werden die zentralen Objekte des Buchs eingeführt: nicht-kommutative und kommutative Polynomringe sowie freie Lie-Algebren. Für diese Strukturen werden dann im dritten Kapitel GS-Basen definiert. Für den wohlbekannten Fall der kommutativen Polynomringe wählen die Autoren dabei eine recht eigenartige Vorgehensweise: sie verwenden

das Buchberger-Kriterium zur Definition der Gröbner-Basen und zeigen dann, dass die übliche Definition dazu äquivalent ist. Für nicht-kommutative Polynome folgen sie einem ähnlichen Ansatz und verwenden zum Äquivalenzbeweis dann Bergmans „Diamond Lemma“, das sie allerdings „Composition-Diamond Lemma“ nennen und Shirshov zuschreiben. Für die freien Lie-Algebren reduzieren die Autoren die Theorie auf die nicht-kommutativen Polynome unter Verwendung der „Lyndon-Shirshov Basis“, die anderswo auch als Chen-Fox-Lyndon Basis bekannt ist.



Ab Kapitel 4 werden Anwendungen, Erweiterungen und Spezialfälle von GS-Basen betrachtet: für freie Produkte, für Moduln über assoziativen Algebren, verschiedene Arten von Lie-Algebren und Entscheidungsprobleme für Gruppen. Das letztere Thema wird in Kapitel 6 behandelt, welches sich deutlich von den anderen Kapiteln unterscheidet: der notwendige Hintergrund aus der Berechenbarkeitstheorie und der Gruppentheorie wird eingeführt, bevor die Unentscheidbarkeit des Wortproblems und des Konjugationsproblems in endlich präsentierten Gruppen erläutert wird. Dieses Kapitel ist wesentlich wortreicher und ausführlicher gehalten als viele andere. Im siebten Kapitel kommen schließlich u. a. Anwendungen der GS-Basen auf die algebraische diskrete Morse-Theorie, auf Zopfgruppen und auf Erweiterungen von Algebren zur Sprache.

Große Teile dieses Buchs sind sehr komprimiert geschrieben und verlangen dem Leser ein hohes Niveau ab. Wie sich aus obiger Beschreibung ergibt, wird eine riesige Menge an Material abgedeckt. Nichtsdestotrotz fehlen jegliche Hinweise auf eine Reihe anderer bedeutender Beiträge zur Theorie der nicht-kommutativen

Gröbner-Basen, z.B. die Arbeiten von Teo Mora, wie sie etwa in seinem Übersichtsartikel [9] zusammengefasst sind, die Konstruktion nicht-kommutativer Gröbner-Basen (z. B. für Pfadalgebren) durch Ed Green, wie sie z. B. in seinem Buch [10] erklärt werden, oder die Letterplace-Technik, die von Roberto La Scala und Viktor Levandovskyy startend mit ihrer Arbeit [11] entwickelt wurde und die z.B. der Implementation im Computeralgebrasystem *Singular* zugrunde liegt.

Die Autoren erklären nicht, was das Ziel und die Absicht ihres Werkes sind und sie deuten auch die Zielgruppe unter den potentiellen Lesern nicht an. In Anbetracht des weitestgehend sehr dichten Schreibstils und der weit gefassten Themenbereiche scheint sich das Buch am ehesten als Quelle der Inspiration für einschlägige Forscher zu eignen. Es wird sicher seinen Platz unter den Standardwerken über nicht-kommutative Gröbner-Basen finden.

Martin Kreuzer (Passau)

#### *Literatur:*

[1] B. Buchberger, Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal, Dissertation, Universität Innsbruck 1965

[2] D. Knuth and P. Bendix, Simple word problems in universal algebras, in J. Leech (Ed), Computational Problems in Abstract Algebra, Pergamon Press, Oxford 1970, S. 263-297

[3] F.S. Macaulay, The algebraic theory of modular systems, Cambridge Univ. Press, 1916

[4] M. Janet, Sur les systemes d'equations aux derivees partielles, J. Math. Pure Appl. 3 (1920), 65-151

[5] Grete Hermann, Die Frage der endlich vielen Schritte in der Theorie der Polynomideale, Math. Ann. 95 (1926), 736-788

[6] A.I. Shirshov, Certain algorithmic problems for Lie algebras (auf Russisch), Siberian Math. J. 3 (1962), 292-296

[7] H. Hironaka, Resolution of singularities of an algebraic variety over a field of characteristic zero, Ann. Math. 79 (1964), 109-326

[8] J. von zur Gathen, Modern computer algebra, Cambridge Univ. Press, 2003

[9] Teo Mora, An introduction to commutative and noncommutative Gröbner bases, Theor. Comp. Sci. 134 (1994), 131-173

[10] Ed Green, An introduction to noncommutative Gröbner bases, Taylor and Francis, 1994

[11] R. La Scala und Viktor Levandovskyy, Letterplace ideals and noncommutative Gröbner bases, J. Symb. Comput. 44 (2009), 1374-1393

---

## Berufungen

---

**Prof. Dr. Georg Regensburger** hat zum 1.10.2021 eine Professur für Computeralgebra an der Universität Kassel angetreten. Seine Arbeitsgebiete sind algebraische und algorithmische Methoden in der Analysis mit Anwendungen für Differential- und Funktionalgleichungen und in der Systembiologie im Zusammenhang mit Reaktionsnetzwerken.

**Sebastian König: 0-Hecke Algebras of the Symmetric Groups: Centers and Modules Associated to Quasisymmetric Schur Functions**

**Betreuer:** Christine Bessenrodt (Hannover)

**Weitere Gutachter:** Meinolf Geck (Stuttgart), Andrew Mathas (Sydney)

**September 2021**

**Abstract:** We consider two aspects of the 0-Hecke algebras of the symmetric groups: their centers and modules associated to quasisymmetric Schur functions.

In 2015 Tewari and van Willigenburg constructed 0-Hecke modules that are mapped to the skew quasisymmetric Schur functions by the quasisymmetric characteristic. These include straight modules that correspond to the ordinary quasisymmetric Schur functions of Haglund, Luoto, Mason and van Willigenburg. The modules admit a natural direct sum decomposition. We study the summands and provide combinatorial rules for their tops and socles. Moreover, we show that they are indecomposable in the straight case. This is a difference to the general skew case where the summands can be decomposable. For a certain kind of skew modules, we describe a decomposition into indecomposable projective modules.

Vector space bases of the centers of the 0-Hecke algebras of the symmetric groups were described by He in 2015. These bases are indexed by certain equivalence classes of permutations whose explicit description is rather complicated. Even their number is not obvious. Building on work of Geck, Kim and Pfeiffer we obtain a set of representatives. This leads to a parametrization of the equivalence classes by certain compositions called maximal. Moreover, we develop an explicit combinatorial description for the equivalence classes indexed by maximal compositions whose odd parts form a hook. We infer that apart from the identity the elements of He's basis corresponding to these equivalence classes annihilate all simple 0-Hecke modules belonging to the nontrivial block of their 0-Hecke algebra.

---

**Laura Maaßen: Representation categories of compact matrix quantum groups**

**Betreuer:** Gerhard Hiß (Aachen) und Moritz Weber (Saarbrücken)

**Weiterer Gutachter:** Amaury Freslon (Paris)

**Juli 2021**

**Abstract:** One key result obtained from the investigation of compact matrix quantum groups is a Tannaka-Krein type duality, by which any compact matrix quantum group can be fully recovered from its representation category. Following

this idea, easy quantum groups are defined through a combinatorial description of their representation categories. In this thesis, we study the representation categories of so-called group-theoretical quantum groups and show that they can be described by a combinatorial calculus similar to that used for easy quantum groups. Furthermore, we analyse the structure of abstract tensor categories that interpolate the representation categories of easy quantum groups. This thesis thus concerns research problems at the intersection of the theory of compact quantum groups, combinatorics and category theory with links to group theory.

The first part of this thesis concerns group-theoretical quantum groups. We define an analogue of orthogonal group-theoretical quantum groups in the unitary setting and show that their description as semi-direct product quantum groups can be generalised. We describe their representation categories, both in the easy and the non-easy case. For this purpose, we introduce modified versions of categories of partitions, which model the 'group-theoretical structure' of the diagonal subgroups of group-theoretical quantum groups. Moreover, we define a modified fiber functor linked with the classical fiber functor via Moebius inversion. Subsequently, we show that the application of the Tannaka-Krein duality yields the desired description of the representation categories of group-theoretical quantum groups.

Next, we restrict our attention to the orthogonal case. Although it is known that uncountably many orthogonal group-theoretical easy quantum groups exist, almost no concrete examples have been studied. We compute various examples with small generators, including in particular a new series of easy quantum groups between the hyperoctahedral series and higher hyperoctahedral series. We conclude our analysis of orthogonal group-theoretical quantum groups by an improved version of a de Finetti theorem by Raum and Weber.

In the second part of this thesis, we study interpolating partition categories in the framework of Deligne's interpolation categories. Interpolating partition categories are the categorical abstraction of categories of partitions together with a complex interpolation parameter. We explain that their semisimplifications interpolate the representation categories of easy quantum groups. Next, we show that the semisimplicity of an interpolating partition category is encoded in the determinants of certain Gram matrices. We compute the set of interpolation parameters yielding semisimple interpolating partition categories for all group-theoretical easy quantum groups. Moreover, we parametrise the indecomposable objects in all interpolating partition categories by an explicitly constructible system of finite groups and exhibit their Grothendieck rings as filtered deformations. We apply these results to orthogonal easy groups and free orthogonal easy quantum groups.

### GAMM-Jahrestagung 2020@21 Minisymposium zur Computeralgebra

Kassel, 15.03. – 19.03.2021

[jahrestagung.gamm-ev.de](http://jahrestagung.gamm-ev.de)

Nachdem die GAMM-Jahrestagung 2020 pandemiebedingt abgesagt werden musste, wurde sie im Zeitraum 15.-19. März 2021 in einem Online-Format nachgeholt. Am Dienstagnachmittag standen vier Minisymposia auf dem Programm, darunter eines zum Thema „Symbolic computation methods for differential equations, dynamical systems, and control theory“. Die Sitzung begann mit einem vierzigminütigen Hauptvortrag von François Boulier (Uni Lille) zur Beziehung zwischen der von Ritt und Kolchin begründeten Differentialalgebra und der tropischen Differentialgeometrie.

Es folgten vier zwanzigminütige Beiträge. Daniel Robertz (Uni Plymouth) berichtete über einen algorithmischen Zugang zur starken Konsistenzanalyse von Finite-Differenzen-Approximationen bei nichtlinearen partiellen Differentialgleichungen, der auf einem Differenzen-Analogon der differentiellen Thomas-Zerlegung basiert. Er widmete seinen Vortrag dem Andenken an seinen Ko-Autor Vladimir Gerdt, der im Januar an den Folgen einer Corona-Infektion verstorben war. Georg Regensburger (JKU Linz) stellte computeralgebraische Methoden für lineare (Integro-Differential-) Operatoren mit Matrix-Koeffizienten vor. Insbesondere betrachtete er Normalformen solcher Operatoren sowie das Finden und Beweisen von Operatoridentitäten. Veronika Pillwein (JKU Linz) sprach über Algorithmen für die Klasse der *DD-endlichen* Funktionen, die eine natürliche Verallgemeinerung der D-endlichen (oder holonomen) Funktionen darstellen. Letztere sind Lösungen linearer Differentialgleichungen mit polynomiellen Koeffizienten, und bilden ihrerseits die zulässige Koeffizientenmenge für die DD-endlichen Funktionen. Schließlich präsentierte Alban Quadrat (INRIA Paris) algorithmische Aspekte des algebraischen Parameterschätzproblems nach Fliess, das er auf lineare Differentialgleichungen mit polynomiellen Koeffizienten erweitert hat und das sich mit nichtkommutativen Gröbnerbasismethoden lösen lässt. Im Anschluss an die Vorträge und die Diskussion gab es noch Gelegenheit zu einer kleinen informellen Nachsitzung im virtuellen Raum.

Werner Seiler (Kassel) und Eva Zerz (Aachen)

### MEGA 2021

Tromsø (virtuell), Norwegen, 07.06. – 11.06.2021

[puremath.no/mega2021](http://puremath.no/mega2021)

Die Mitternachtssonne des arktischen Tromsø schien bei der diesjährigen MEGA nur virtuell, denn ein anderes Austragungsformat war angesichts der andauernden Pandemielage nicht planbar. Die Organisatoren taten jedoch alles, um aus der Not eine Tugend zu machen. So wurde das Online-Konferenztool **gather.town** benutzt, um die von der Konferenz benutzten Räume, Gebäude und deren Umgebung virtuell nachzubilden, einschließlich einem plätschernden Brunnen vor dem Hauptgebäude. Als Avatar, mit Namensschild wie in einer echten Konferenz, musste man dann seinen Weg finden zu den Frage- und Postersitzungen. Dies hatten die Organisatoren perfekt implementiert, ein beeindruckender Erfolg. All dies wurde geboten, ohne dass eine Tagungsgebühr verlangt wurde. Allerdings verlangte das

Konzept einiges an Internetbandbreite, und der Berichterstatter fand es ein klein wenig geisterhaft, wenn er auf der Wegstrecke in die Nähe zweier Avatare geriet und auf einmal ganz klar und deutlich deren Zwiegespräch über Berufspläne und Chancen auf ein Postdoc an verschiedenen Orten mitzuhören bekam. (Aber sicherlich erschien Neulingen vor hundert Jahren das Telefon mindestens ebenso geisterhaft.)

Nun aber zu den Punkten, die ein Konferenzbericht eigentlich enthalten soll. MEGA ist ein Akronym für **E**ffektive **M**ethoden in **A**lgebraischer **G**eometrie, dessen Buchstaben sich nach Übersetzung in eine romanische Sprache in der richtigen Reihenfolge zusammensetzen. Die Konferenz findet regelmäßig alle zwei Jahre statt, und beinhaltet eingela-dene sowie eingereichte Vorträge, Posterpräsentationen und Software-Demonstrationen. So war es auch diesmal, wobei die zehn eingeladenen Vorträge synchron per Zoom abgehalten wurden, während die knapp 40 eingereichten Vorträge als Aufnahmen zur Verfügung standen, angereichert mit „Teasern“. Bemerkenswert war, dass sechs der zehn eingeladenen Vortragenden Frauen waren. Auch in diesem Jahr gibt es anlässlich der Konferenz einen Sonderband des Journal of Symbolic Computation, für den bis zum 30.09. Beiträge eingereicht werden konnten, auch von Forscherinnen und Forschern, die nicht auf der MEGA vorgetragen hatten. Selbstverständlich durchlaufen die Einreichungen den ganz normalen Begutachtungsprozess, so wie auch die bei der Konferenz eingereichten Vorträge vorab begutachtet wurden.

Die nächste MEGA findet bereits im nächsten Jahr statt, und zwar vom 22. bis 24. Juni 2022 in Krakau. Geplant ist, diese wieder als Präsenzkonferenz abzuhalten.

Gregor Kemper (München)

### ISSAC 2021

Hybridformat, St. Petersburg, 18.07. – 23.07.2021

[www.issac-conference.org/2021](http://www.issac-conference.org/2021)

Ein zweites Mal stand die diesjährige ISSAC, wie jede andere Konferenz, im Schatten der Pandemie. Die Organisatoren wählten ein Hybridformat, wobei allerdings von 261 Teilnehmern nur 13 tatsächlich vor Ort waren. Die Organisatoren haben ganze Arbeit geleistet, um den technischen und organisatorischen Ablauf der Konferenz reibungslos und “benutzerfreundlich” zu gestalten. So umfasste die „Gebrauchsanleitung“ für Teilnehmer diesmal nur vier Seiten. Die verminderten Kosten, die bei einer weitgehend virtuellen Konferenz anfallen, wurden an die Teilnehmer weitergegeben: Die Tagungsgebühren betrugen lediglich \$15 für Online-Teilnehmer.

Somit sollten die Hauptorganisatoren hier auch namentlich erwähnt werden. Als General Chair war Frédéric Chyzak für alle Fragen und Probleme immer ansprechbar. Nikolai Vasiliev fungierte als Local Arrangements Chair, und George Labahn leitete das Programm-Komitee, das unter anderem aus 70 eingereichten Vorträgen 43 ausgewählt hat, natürlich nach Einholung von Gutachten (insgesamt 177).

Entsprechend dem Hybridformat wurden sämtliche Vorträge synchron über Zoom, also diesmal nicht als Aufzeichnungen, abgehalten. Trotzdem mussten auf Rücksicht auf Teilnehmer von Japan bis Kalifornien die Zeiten auf den St. Petersburger Nachmittag bis frühen Abend konzentriert



werden. Daher die untypisch lange Dauer der diesjährigen ISSAC.

Neben den eingereichten Vorträgen gab es drei eingeladene Hauptvorträge, die synchron über Zoom abgehalten wurden. Auch gab es diesmal wieder drei Tutorials, von denen eins von unserer Sprecherin, Anne Frühbis Krüger, gemeinsam mit Janko Böhm, abgehalten wurde. Es gab auch vier Software Demos und zehn Poster, aber weil „Poster“ im virtuellen Format wenig Sinn ergibt, wurden diese als „Short Communications“ gehandelt. Für weitere Details, Titel und Autoren sei auf die wie immer gut organisierte Homepage der Konferenz verwiesen.

Am Rande der ISSAC fand ein halbtägiger Workshop zu Ehren von Vladimir Gerdt statt, der am 5. Januar verstorben war. Dies war sehr passend, denn Vladimir Gerdt war immer eng mit der ISSAC verbunden und wurde insbesondere in diesem Jahr und an diesem Standort schmerzlich vermisst. Ich kann mich gut erinnern, wie Vladimir bei der 2019er ISSAC in Peking den „Werbevortrag“ für den Austragungsort St. Petersburg präsentierte.

Die Fachgruppe Computeralgebra ist mit der ISSAC verbunden unter anderem durch die von uns vergebenen Preise. Diese sind der Distinguished Software Demonstration Award und der Distinguished Poster Award, wobei letzterer konsequenterweise in diesem Jahr zu Distinguished Short Communication Award umbenannt wurde. Erneut wurden für die Preisträger durch die ohnehin bestehenden Poster- und Software-Komitees ermittelt, wofür wir uns sehr bedanken. Der Preis für die **beste Short Communication** ging an:

**Autor:** Andrei Matveiak.

**Titel:** Discovering multiple polylogarithm equations via symbolic computations.

Als **beste Software-Demo** wurde ausgezeichnet:

**Autoren:** Ilia Ilmer, Alexey Ovchinnikov und Gleb Pogudin.

**Titel:** Maple application for structural identifiability analysis of ODE models.

Wie immer gab es das ISSAC Business Meeting, welches durch das vorherige Verteilen von Materialien bestens vorbereitet war. Einer der Punkte war die Wahl eines neuen Mitglieds des Steering Committee. Hierfür traten drei Kandidatinnen und Kandidaten an, von denen Mohab Safey El Din (Sorbonne, Paris) gewählt wurde. Auch diesmal lagen wieder mehr als eine Bewerbung für den Austragungsort für das Jahr 2023 vor: Notre Dame (Indiana, USA) und Tromsø (Norwegen). In der Abstimmung setzte sich Tromsø recht

deutlich durch. Die nächste Auflage der ISSAC findet jedoch vom 4. bis 7. Juli 2022 in Lille (Frankreich) statt. Die Veranstalter hoffen, dann wieder eine echte Präsenz-Konferenz abhalten zu können, ein Wunsch, dem ich mich ausdrücklich anschließen möchte.

Gregor Kemper (München)

## Gemeinsame Jahrestagung der DMV und ÖMG

Passau (online), 27.09. – 01.10.2021

[www.uni-passau.de/dmv-oemg-2021](http://www.uni-passau.de/dmv-oemg-2021)

Vom 27.9. bis zum 1.10.2021 fand in Passau (virtuell) die gemeinsame Jahrestagung der DMV und der ÖMG statt. Die Computeralgebra war auf mehrere Arten dabei vertreten.

Zum einen gab es seit Jahren wieder einmal eine eigene Sektion Computeralgebra, die vom Manuel Kauers (Linz) und dem Berichterstatter geleitet wurde. Dort wurden eine Reihe neuer und neuester Entwicklungen in der Computeralgebra vorgestellt, unter anderem: Mohamed Barakat erklärte, wie so die Kategorientheorie eigentlich eine Programmiersprache für die Computeralgebra ist, Georg Regensburger und Viktor Levandovskyy zeigten Beweise von Operatoridentitäten mit Hilfe nicht-kommutativer Gröbner-Basen, Christoph Koutschan und Marta Panizzut besprachen neue Anwendungen in der diskreten Mathematik, Le Ngoc Long in der algebraischen Geometrie, Daniel Robertz in der Invariantentheorie, Werner Seiler auf Berechnungen für Differentialgleichungen, und Christian Eder stellte das neue Paket `msolve` zum Lösen 0-dimensionaler polynomialer Gleichungssysteme vor, das alles bisher Dagewesene um Längen schlägt.

Des Weiteren hielt Max Horn einen Vortrag *Towards confirmable computer algebra* in der Sektion über das MaRDI Projekt. Schließlich kamen einzelne Aspekte und Anwendungen der Computeralgebra auch in einer Reihe weiterer Sektionen und Vorträge vor, z. B. der Klassifikation algebraischer Varietäten, der isogeniebasierten Kryptographie, oder der Berechnungen in der Teilchenphysik.

Im öffentlichen Vortrag sprach Christian Hesse über *Schach und Mathematik*, und die Cantormedaille wurde vergeben an Martin Grötschel für seine Arbeiten zur den Anwendungen der diskreten Optimierung. Pandemiebedingt konnte die Veranstaltung nur als Online-Konferenz durchgeführt werden, so dass den Teilnehmern die Schönheit der Stadt Passau und ihrer Universität leider vorenthalten blieb.

Martin Kreuzer

---

## Hinweise auf Konferenzen

---

### **Combinatorial Algebra meets Algebraic Combinatorics (CAAC) 2022**

Toronto, Kanada, 21.01. – 23.01.2022

[garsia.math.yorku.ca/CAAC\\_2022](http://garsia.math.yorku.ca/CAAC_2022)

This workshop is the 19th in a series of weekend workshops that bring together the mathematical community sharing interests in algebraic combinatorics, commutative algebra and combinatorial algebraic geometry.

Additional information can be found at the conference webpage.

### **Computeralgebra-Tagung der Fachgruppe**

München, 10.03. – 12.03.2022

[www.fachgruppe-computeralgebra.de](http://www.fachgruppe-computeralgebra.de)

Die Jahrestagung der Fachgruppe Computeralgebra sollte turnusgemäß im Frühjahr 2021 stattfinden. Aufgrund der COVID-19-Pandemie wurde sie auf das Jahr 2022 verschoben. Nach den vielen erfolgreichen Tagungen 2003, 2005, 2009, 2012, 2014, 2017, 2019 in Kassel und 2007 in Kaiserslautern wird die Tagung im Jahr 2022 in München stattfinden. Eine ausführliche Ankündigung finden Sie auf Seite 6.

Das Ziel dieser Tagungsreihe ist es, ein Forum zu bieten, das es erstens Nachwuchswissenschaftlern ermöglicht, ihre Ergebnisse vorzustellen, andererseits aber auch einige Hauptvortragende zu gewinnen, die Übersichtsvorträge über wichtige Gebiete der Computeralgebra und über Computeralgebra-Software geben sollen.

### **CoCoA - School and Conference on Computational Commutative Algebra**

Hue, Vietnam, März 2022

[cocoa.dhsphue.edu.vn](http://cocoa.dhsphue.edu.vn)

Die internationale Doktorandenschule und Tagung COCOA 2020 ist aufgrund der Pandemie nochmals verschoben worden und findet jetzt im März 2022 in Hue (Vietnam) statt.

### **MEGA 2022**

Krakau, Polen, Juni 2022

[mega.sciencesconf.org](http://mega.sciencesconf.org)

MEGA is the acronym for Effective Methods in Algebraic Geometry (and its equivalent in Italian, French, Spanish, German, Russian, etc.). This series of biennial international conferences, with the tradition dating back to 1990, is devoted to computational and application aspects of Algebraic Geometry and related topics, over any characteristics.

MEGA 2022 will take place in Krakow, Poland, in June 2022. Mark your calendars!

### **CCAAGS-22**

Seattle, USA, 27.06. – 01.07.2022

[sites.google.com/view/ccaaggs-22/home](https://sites.google.com/view/ccaaggs-22/home)

CCAAGS-22 aims to bring together researchers working in the creative mixture of combinatorial and computational ideas in applied algebraic geometry. As part of the event we will celebrate Bernd Sturmfels and his contributions to the field.

Details of the conference will be posted at the website as they become available.

### **ISSAC 2022**

Lille, Frankreich, 04.07. – 07.07.2022

[www.issac-conference.org/2022](http://www.issac-conference.org/2022)

The International Symposium on Symbolic and Algebraic Computation (ISSAC) is the premier conference for research in symbolic computation and computer algebra. ISSAC 2022 will be the 47th meeting in the series, which started in 1966 and has been held annually since 1981. The conference presents a range of invited speakers, tutorials, poster sessions, software demonstrations and vendor exhibits with a center-piece of contributed research papers.

---

## Fachgruppenleitung Computeralgebra 2020–2023

---

**Sprecherin:**

Prof. Dr. Anne Fruehbs-Krueger  
Carl-von Ossietzky Universität Oldenburg  
Institut für Mathematik  
Carl-von-Ossietzky-Straße 11, 26129 Oldenburg  
0441 798-3233  
[anne.fruehbs-krueger@uni-oldenburg.de](mailto:anne.fruehbs-krueger@uni-oldenburg.de)  
<https://uol.de/anne-fruehbs-krueger>

**Stellvertretender Sprecher:**

Prof. Dr. Gregor Kemper  
Zentrum Mathematik – M11  
Technische Universität München  
Boltzmannstr. 3, 85748 Garching  
089 289-17454, -17457 (Fax)  
[kemper@ma.tum.de](mailto:kemper@ma.tum.de)  
<http://www.groups.ma.tum.de/algebra/kemper>

**Vertreterin der GI:**

Prof. Dr. Erika Abraham  
Fachgruppe Informatik  
RWTH Aachen University  
Ahornstr. 55, 52056 Aachen  
0241 80-21242, -22243 (Fax)  
[abraham@cs.rwth-aachen.de](mailto:abraham@cs.rwth-aachen.de)  
<https://ths.rwth-aachen.de/people/erika-abraham/>

**Fachreferentin Industrie:**

Xenia Bogomolec  
Coding Services Hannover  
Engelbosteler Damm 15, 30167 Hannover  
0173 3031816  
[indigomind@protonmail.ch](mailto:indigomind@protonmail.ch)  
<https://quant-x-sec.com>

**Fachreferent CA an der Hochschule:**

Prof. Dr. Michael Cuntz  
Leibniz Universität Hannover  
Institut für Algebra, Zahlentheorie und Diskrete Math.  
Welfengarten 1, 30167 Hannover  
0511 762-4252  
[cuntz@math.uni-hannover.de](mailto:cuntz@math.uni-hannover.de)  
<http://www.iazd.uni-hannover.de/~cuntz>

**Fachreferent CA-Systeme und -Bibliotheken:**

Prof. Dr. Claus Fieker  
Fachbereich Mathematik  
Technische Universität Kaiserslautern  
Gottlieb-Daimler-Straße, 67663 Kaiserslautern  
0631 205-2392, -4427 (Fax)  
[fieker@mathematik.uni-kl.de](mailto:fieker@mathematik.uni-kl.de)  
<http://www.mathematik.uni-kl.de/~fieker>

**Fachexperte Physik:**

Dr. Thomas Hahn  
Max-Planck-Institut für Physik  
Föhringer Ring 6, 80805 München  
089 32354-300, -304 (Fax)  
[hahn@feynarts.de](mailto:hahn@feynarts.de)  
<http://wwwth.mpp.mpg.de/members/hahn>

**Vertreter der DMV:**

Prof. Dr. Florian Hess  
Carl-von Ossietzky Universität Oldenburg  
Institut für Mathematik, 26111 Oldenburg  
0441 798-2906, -3004 (Fax)  
[florian.hess@uni-oldenburg.de](mailto:florian.hess@uni-oldenburg.de)  
<https://uol.de/florian-hess>

**Fachreferent CA-Systeme und -Bibliotheken:**

Prof. Dr. Max Horn  
Fachbereich Mathematik  
Technische Universität Kaiserslautern  
Gottlieb-Daimler-Straße, 67663 Kaiserslautern  
0631 205-2730, -4427 (Fax)  
[horn@mathematik.uni-kl.de](mailto:horn@mathematik.uni-kl.de)  
<https://www.quendi.de/de/mathe>

**Fachreferent Themen, Anwendungen und Publikationen:**

Prof. Dr. Jürgen Klüners  
Mathematisches Institut der Universität Paderborn  
Warburger Str. 100, 33098 Paderborn  
05251 60-2646, -3516 (Fax)  
[klueners@math.uni-paderborn.de](mailto:klueners@math.uni-paderborn.de)  
<https://math.uni-paderborn.de/ag/klueners/>

**Fachreferent Themen, Anwendungen und Publikationen:**

Prof. Dr. Martin Kreuzer  
Fakultät für Informatik und Mathematik  
Universität Passau  
Innstr. 33, 94030 Passau  
0851 509-3120, -3122 (Fax)  
[martin.kreuzer@uni-passau.de](mailto:martin.kreuzer@uni-passau.de)  
<http://www.fim.uni-passau.de/~kreuzer>

**Fachreferent Redaktion Rundbrief:**

Dr. Fabian Reimers  
Zentrum Mathematik – M11  
Technische Universität München  
Boltzmannstr. 3, 85748 Garching  
089 289-17474  
[reimers@ma.tum.de](mailto:reimers@ma.tum.de)  
<http://www.groups.ma.tum.de/algebra/reimers>

**Vertreterin der GAMM:**

Prof. Dr. Eva Zerz  
Lehrstuhl D für Mathematik  
RWTH Aachen  
Pontdriesch 14/16, 52062 Aachen  
0241 80-94544, -92108 (Fax)  
[eva.zerz@math.rwth-aachen.de](mailto:eva.zerz@math.rwth-aachen.de)  
<http://www.math.rwth-aachen.de/~Eva.Zerz/>



## Ihr Fundus an spannenden Ideen für den Unterricht

Auf der TI Materialdatenbank finden Sie praxisorientierte Aufgabenbeispiele für Ihren MINT-Unterricht, meist zum kostenlosen Download. Videos, Tutorials und Verweise auf die Lehrwerke der Schulbuchverlage gibts noch dazu.

**Stöbern, herunterladen – und teilen!**  
**[www.ti-unterrichtsmaterialien.net](http://www.ti-unterrichtsmaterialien.net)**

