EACA 2016

XV ENCUENTRO
ÁLGEBRA COMPUTACIONAL Y APLICACIONES

Edited by
Jónathan Heras
and Ana Romero

UNIVERSIDAD DE LA RIOJA

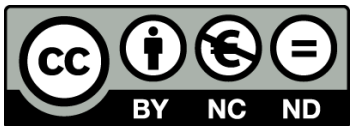
XV Encuentro de Álgebra Computacional y Aplicaciones, EACA 2016

Edited by
Jónathan Heras and Ana Romero

Universidad de La Rioja
2016

EACA (15º. 2016. Logroño)
XV Encuentro de Álgebra Computacional y Aplicaciones : EACA 2016 /
edited by Jónathan Heras and Ana Romero. - Logroño : Universidad de
La Rioja, 2016.
156 p. : il. ; 29 cm.
ISBN 978-84-608-9024-9
1. Álgebra. 2. Congresos y asambleas. I. Heras, Jónathan. II. Romero,
Ana. III. Universidad de La Rioja. IV. Título.

512 (063)
PBF -- IBIC 1.1



http://www.unirioja.es/EACA2016
E-mail: publicaciones@unirioja.es
ISBN 978-84-608-9024-9
Edita: Universidad de La Rioja

Contents

Foreword

The "Encuentros de Álgebra Computacional y Aplicaciones" (Meetings on Computer Algebra and Applications) are organized by the Spanish "Red Temática de Cálculo Simbólico, Álgebra Computacional y Aplicaciones" to provide a meeting frame for researchers in the fields of Computer Algebra and Symbolic Computation, and for those who use these techniques in their research. We emphasize and specially favor the participation of young researchers.

This XV Meeting (biennial since 2002) is the natural continuation of those organized in Santander (1995), Sevilla (1996), Granada (1997), Sigüenza (1998), Tenerife (1999), Barcelona (2000), Ezcaray (2001), Valladolid (2002), Santander (2004), Sevilla (2006), Granada (2008), Santiago de Compostela (2010), Alcalá de Henares (2012) and Barcelona (2014). During these years, the conference has achieved a remarkable relevance and prestige within the Symbolic Computation community. The main subjects of interest of the meetings are:

- Effective Methods in Algebra, Analysis, Geometry and Topology.
- Algorithmic Complexity.
- Scientific Computation by means of Symbolic-Numerical Methods.
- Symbolic-Numeric Software Development.
- Analysis, Specification, Design and Implementation of Symbolic Computation Systems.
- Applications to Science and Technology.

EACA 2016 will take place in Logroño, at the Centro Científico y Tecnológico (Universidad de La Rioja), from June 22nd to 24th, 2016, together with the satellite event AICA ("Aplicaciones Industriales del Álgebra Computacional"). More information can be found at the websites `www.unirioja.es/EACA2016` and `www.unirioja.es/AICA2016`.

This book contains the extended abstracts of the accepted contributions and the plenary talks. EACA2016 has 34 contributions, accepted after a standard referee process, and 5 plenary talks. The plenary speakers are:

- Carlos D'Andrea, Universitat de Barcelona, Spain.
- Enrique Artal, Universidad de Zaragoza, Spain.
- Mohamed Barakat, University of Siegen, Germany.
- Lawrence C. Paulson, University of Cambridge, England.
- Andrea Solotar, Universidad de Buenos Aires, Argentina.

We would like to express our sincere gratitude to all organizers, specially to the members of the Scientific Committee: María Emilia Alonso (U. Complutense de Madrid), Isabel Bermejo (U. La Laguna), Francisco J. Castro (U. Sevilla, *Chair*), Carlos D'Andrea (U. Barcelona), Joan Elías (U. Barcelona), Philippe Giménez (U. Valladolid), José Gómez-Torrecillas (U. Granada), Laureano González-Vega (U. Cantabria), Manuel Ladra (U. Santiago de Compostela), Antonio Montes (U. Politècnica de Catalunya), Tomás Recio (U. Cantabria), Ana Romero (U. La Rioja) and J. Rafael Sendra (U. Alcalá de Henares) and of the Local Committee: Jesús María Aransay, Jose Divasón, César Domínguez, Francisco J. García, Jónathan Heras (Webmaster), Arturo Jaime, Laureano Lambán, Gadea Mata (Webmaster), Eloy Mata, Juan José Olarte, M. Vico Pascual, Beatriz Pérez, Ana Romero (Chair), Ángel Luis Rubio, Carlos Sáenz and Eduardo Sáenz de Cabezón.

Specially thanks are due to Julio Rubio, in charge of the organization of EACA16 until April 2016. The success of the conference is possible because of his effort and his great work.

Finally, we would like to thank to the following institutions for their financial support:

- Universidad de La Rioja.
- Departamento de Matemáticas y Computación.
- Ministerio de Economía y Competitividad.
- Real Sociedad Española de Matemáticas.
- Agrupación Empresarial Innovadora del sector TIC de La Rioja, AERTIC.
- Foundation Compositio Mathematica.
- MI-NET: Mathematics for Industry Network.

We finish wishing to all participants a successful conference and a very pleasant stay in the city of Logroño.

Jónathan Heras and Ana Romero
Logroño, June 2016

Plenary talks

ELIMINATION THEORY IN POSITIVE CHARACTERISTIC

CARLOS D'ANDREA

ABSTRACT. Several problems of elimination theory involving arithmetic over the integers (like resultants, the Nullstellensatz, etc.) have as an outcome a number which if it is not zero modulo a prime p, implies that classical results over the complex number (dimension, number of zeroes, etc.) "descend" to the residual field. In this talk we will show some of these properties, applications, as well as some attempts to understand what happens when these invariants do vanish modulo p.

Universitat de Barcelona, Spain.
E-mail address: cdandrea@ub.edu

COMPUTATIONAL METHODS IN THE TOPOLOGY OF ALGEBRAIC VARIETIES

ENRIQUE ARTAL

Abstract. The topological study of algebraic varieties may involve plenty of computational techniques. In the late 20's, Zariski (and van Kampen) gave a method to compute the fundamental group of the complement of an algebraic curve. Though this method had important theoretical consequences its actual use for the application to curves of degree not so big needs a lot of computation, which involves both standard techniques in commutative algebra (e.g., Groebner basis) and numerical techniques. Even once these groups are computed, we encounter the problem of getting properties for finite presentations of groups. Alexander-type invariants need to be used combining group theory, commutative algebra and algebraic geometry from the computational point of view. In this talk, we present different ways for the application of these techniques, both as empirical testing and a way of proving theorems.

Universidad de Zaragoza, Spain.
E-mail address: artal@unizar.es

HOW TO IMPLEMENT A CATEGORY ON THE COMPUTER AND WHY?

MOHAMED BARAKAT

Abstract. What does it mean to program a category? Can one program localisations of categories including the derived category formalism and what is this good for? I will try to answer these questions by showing our applications, how far we got, and where we are heading.

University of Siegen, Germany
E-mail address: mohamed.barakat@uni-siegen.de

THE FUTURE OF FORMALISED MATHEMATICS

LAWRENCE C. PAULSON

Abstract. Recent years have witnessed tremendous achievements in formalised mathematics, including the completion of the Flyspeck project (a machine-checked proof of the Kepler Conjecture) and the formalisation of the odd order theorem, the central limit theorem and Gödel's second incompleteness theorem. Formalised mathematics has started to attract the attention of mainstream mathematicians such as Harvey Friedman, Tim Gowers and Tom Hales. Nevertheless, there is much disagreement on the details of formalisms (constructive or classical, typed or typeless), proof languages (linear or structured) and automation (minimal, heuristic or algorithmic). The recent translation of the HOL Light multivariate analysis library to Isabelle highlights some of these differences. The speaker will address these issues, referencing recent developments in the formalisation of real algebraic geometry.

University of Cambridge, England
E-mail address: lp15@cam.ac.uk

REWRITING PROCESSES, PROJECTIVE RESOLUTIONS AND AMBIGUITIES

ANDREA SOLOTAR

Abstract. The aim of this talk is to explain our method [1] to construct bimodule resolutions of associative algebras using rewriting processes, and generalizing Bardzell's well-known resolution of monomial algebras. This method leads to concrete computations, providing a useful tool for computing invariants associated to algebras presented by generators and relations. In this talk I will illustrate how to use it by giving some examples. I will also discuss conditions on the rewriting system that guarantee that the obtained resolution is minimal.

References

[1] Sergio Chouhy and Andrea Solotar. *Projective resolutions of associative algebras and ambiguities.* To appear in Journal of Algebra. doi:10.1016/j.jalgebra.2015.02.019, arXiv:1406.2300.

Universidad de Buenos Aires, Argentina.
E-mail address: asolotar@dm.uba.ar

Contributed talks

DESCUBRIMIENTO AUTOMÁTICO EN GEOGEBRA. PRIMEROS PASOS

MIGUEL A. ABÁNADES, FRANCISCO BOTANA Y TOMAS RECIO

Abstract. A prototype for automatic discovery of geometric theorems, based on the merging of the dynamic geometry computer program GeoGebra and the computer algebra system Singular (via Sage) is presented.

Introducción

En los últimos años parte de nuestro trabajo se ha centrado en dotar de capacidades simbólicas a la geometría dinámica en tareas tales como la obtención de lugares geométricos [1], envolventes [2] y demostración automática de teoremas [3]. Una estrecha colaboración con desarrolladores de software ha permitido incorporar estos resultados al programa de Geometría Dinámica GeoGebra[1]. Este programa, usado por más de veinte millones de estudiantes en el mundo, posibilita que ideas y resultados, hasta hace poco limitados a ser meros prototipos académicos, puedan ser testados en condiciones *reales* de uso en procesos de enseñanza y aprendizaje de las matemáticas.

En esta comunicación describimos los primeros pasos en torno a la implementación de la capacidad de *descubrimiento automático en geometría*, específicamente en el programa GeoGebra. Téngase en cuenta que por *descubrimiento* nos referimos a la obtención de condiciones necesarias para la satisfacción de alguna propiedad que enunciemos sobre algunos objetos de una construcción geométrica, y por *automático* a que el proceso para la generación de esas condiciones sea puramente mecánico, es decir, sin intervención humana.

En la Sección 1 damos una breve descripción y referencias del protocolo utilizado por nosotros para el *descubrimiento*. La Sección 2 ilustra, sobre un resultado bien conocido, cómo un usuario/a inexperto/a realizaría –usando nuestro prototipo– un descubrimiento, y, finalmente, cómo se validaría dicho descubrimiento usando la opción –ya implementada en GeoGebra, desde la versión 5– de demostración automática.

1. Descubrimiento automático en geometría

Por demostración automática nos referimos a la verificación mecánica de enunciados geométricos. Por contra, el descubrimiento automático trata de la obtención de hipótesis complementarias para que enunciados cualesquiera sean verdaderos. En otros términos, el objetivo es encontrar hipótesis ocultas para que una conclusión dada se siga de un conjunto incompleto de hipótesis.

El método que ahora seguiremos para el descubrimiento fue propuesto en [5] y consiste, *grosso modo*, en la eliminación de unas ciertas variables en la traducción algebraica del

[1]GeoGebra. Dynamic mathematics for learning and teaching, http://www.geogebra.org

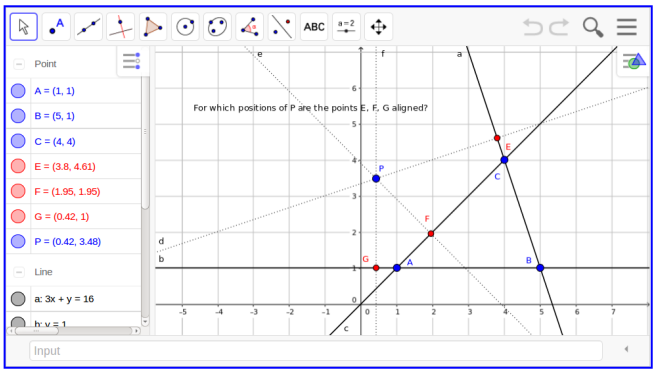


FIGURA 1. ¿Para qué puntos P están sus proyecciones E, F y G alineadas?

problema geométrico. Dada una colección de hipótesis $H = \{h_1 = 0, \ldots, h_p = 0\}$ y una tesis T, asumimos que T no se sigue de H, $H \not\Rightarrow T$. Es decir, si $\{x_1, \ldots, x_d, \ldots, x_n\}$ son las variables de hipótesis y tesis, siendo las d primeras geométricamente independientes, y t una variable muda para el truco de Rabinowitsch, se satisface que

$$(0) = (h_1, \ldots, h_p, tT - 1)K[x_1, \ldots, x_n, t] \cap K[x_1, \ldots, x_d].$$

La idea consiste en añadir la tesis al conjunto de hipótesis. Puesto que $H \wedge T \Rightarrow T$, es claro que, independientemente de T, estamos ante una proposición verdadera. Geométricamente, la intersección de la variedad de las hipótesis y la de la tesis está contenida en esta. La clave está en reformular la conjunción de hipótesis y tesis en términos de variables geométricamente significativas, es decir, de las variables independientes. En resumen, el método se reduce a la eliminación de las variables dependientes en el ideal (hipotesis, tesis). La anulación de cualquier polinomio h' en el ideal de eliminación

$$(\text{hipótesis}, \text{tesis}) \cap K[\text{variables independientes}]$$

es, pues, una condición necesaria para la satisfacción de $H \Rightarrow T$. Véase [5] para una descripción pormenorizada del método de descubrimiento.

2. UN SENCILLO EJEMPLO DE DESCUBRIMIENTO

Siguiendo nuestra práctica habitual hemos construido un prototipo[2] de descubrimiento basado en el método expuesto que usa un applet de GeoGebra y Singular, a través de Sage, para la eliminación. Previsiblemente, tal prototipo será incorporado a GeoGebra en un futuro próximo.

Aquí ilustramos su uso mediante una propiedad elemental de la geometría del triángulo, el teorema de Simson-Wallace. La situación inicial puede verse en la figura 1. Dado un triángulo ABC y un punto P en su plano, se trazan las proyecciones ortogonales E, F, G sobre los lados. Se trata de estudiar cuándo tales proyecciones están alineadas.

Independientemente de que tal pregunta haya sido sugerida al usuario o se le ocurra a este mismo, resulta natural codificarla como `Descubrir(P, Alineados(E,F,G))`. La aplicación

[2] http://193.146.36.205:8080/GgbSageDirect/DiscoverGGB/

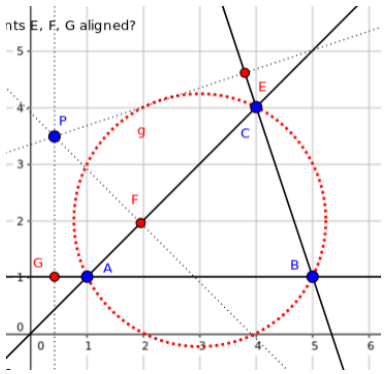


FIGURA 2. E, F y G están alineados si P está en el circuncírculo de ABC.

traduce la construcción a ecuaciones polinomiales y asigna variables a cada punto. En la versión actual, las coordenadas de los puntos básicos de la construcción son numéricas, pero no las del punto P sobre el que se propone el descubrimiento, que son puramente simbólicas. Aunque esta decisión resta generalidad al descubrimiento, téngase en cuenta que se evitan situaciones de degeneración que implicarían un estudio más detallado del resultado. Se descubre sobre un triángulo concreto, una instancia específica de la construcción. Cualquier propiedad encontrada será una condición necesaria solo para ese triángulo, pero la generalización es posible usando demostración automática. La introducción de datos se completa especificando el punto sobre el que se quiere descubrir, P, y la(s) condición(es) impuesta(s).

Para este descubrimiento, las únicas variables independientes son las de $P(x_1, x_2)$ y las variables a eliminar son las de las proyecciones, E, F, G. La eliminación devuelve el polinomio $x_1^2 + x_2^2 - 6x_1 - 4x_2 + 8$, que describe el circuncírculo del triángulo dado. Es decir, una condición necesaria para la alineación de las proyecciones es que P se mueva sobre este círculo. En nuestro prototipo tal respuesta se muestra dibujando esa circunferencia (figura 2), y también en lenguaje natural merced a la consulta a un diccionario álgebro-geométrico construido para la instancia considerada.

La suficiencia de la condición encontrada puede comprobarse acudiendo a las facilidades de demostración automática que ofrece GeoGebra. Escribiendo "Prove[AreCollinear[E,F,G]]" obtenemos la verdad de la afirmación, estableciendo así el teorema de Simson-Wallace, salvo en caso de degeneración, que GeoGebra resume en la coincidencia de dos de los tres vértices usando "ProveDetails[AreCollinear[E,F,G]]".

Desde un punto de vista más técnico, el proceso usa el servicio de SageMathCell de Sage[3], donde la descripción XML de la construcción de GeoGebra es enviada en primer lugar. A continuación, en el servidor de Sage, la construcción sigue un proceso de "algebrización", según la especificación de código desarrollado por los autores. Con el sistema polinomial obtenido se lleva a cabo un proceso de eliminación de las variables distintas de las de los puntos sobre los que se descubre, mediante algoritmos basados en bases de Gröbner en Singular. El resultado de esta eliminación es analizado y procesado para su incorporación en el applet. La comunicación GeoGebra-Sage se basa en JavaScript.

[3]https://sagecell.sagemath.org/

Conclusiones

Creemos que la generalización y popularización de la capacidad de demostración y descubrimiento automático en un programa como GeoGebra exigirá una reflexión sobre su impacto en la enseñanza de las matemáticas, al igual que ya afecta ahora mismo, en el ámbito académico, a la forma en la que se investiga en geometría elemental. Evidentemente, la capacidad de descubrimiento automático en un programa de Geometría Dinámica convierte al mismo en una especie de tutor inteligente para guiar al alumno en las tareas de experimentación y de desarrollo del pensamiento inductivo que forman parte, cada vez más, de la enseñanza actual de la Geometría, frente a la aproximación tradicional, que primaba el carácter formal y deductivo. Como señala [4], se trata de usar la demostración automática para ayudar

> "... exploring and modeling the more creative human-like thought processes of inductively exploring and manipulating diagrams to discover new insights about geometry".

En todo caso planteamos, como continuación natural de este trabajo, su mejora y testeo, de cara a la incorporación del prototipo a una próxima versión de GeoGebra.

Agradecimientos

Los resultados obtenidos forman parte del Proyecto de Investigación "Construcciones Algebra-geométricas: fundamentos, algoritmos y aplicaciones" (MTM2014-54141-P).

Referencias

[1] Abánades, M.A., Botana, F., Montes, A., Recio, T.: An algebraic taxonomy for locus computation in dynamic geometry, *Computer-Aided Design* 56 (2014) 22–33

[2] Botana, F., Recio, T.: Some issues on the automatic computation of plane envelopes in interactive environments, *Mathematics and Computers in Simulation* 125 (2016) 115–125

[3] Botana, F., Hohenwarter, M., Janičić, P., Kovács, Z., Petrović, I., Recio, T., Weitzhofer, S.: Automated theorem proving in GeoGebra: Current achievements, *Journal of Automated Reasoning* 55 (2015) 39–59

[4] Johnson, L. E.: Automated Elementary Geometry Theorem Discovery via Inductive Diagram Manipulation. Master Thesis. Master of Engineering in Electrical Engineering and Computer Science, Massachusetts Institute of Technology. (2015).

[5] Recio, T., Vélez, M.P.: Automatic discovery of theorems in elementary geometry, *Journal of Automated Reasoning* 23 (1999) 63–82

Depto. de Economía Financiera y Contabilidad e Idioma Moderno, Universidad Rey Juan Carlos, Madrid

E-mail address: miguelangel.abanades@urjc.es

Depto. de Matemática Aplicada I, EE Forestal, Campus A Xunqueira, Universidade de Vigo, Pontevedra

E-mail address: fbotana@uvigo.es

Depto. de Matemáticas, Estadística y Computación, Universidad de Cantabria, Santander

E-mail address: tomas.recio@unican.es

COMPUTING THE MEDIAL AXIS FOR CLOSED PLANAR DOMAINS BOUNDED BY FINITELY MANY SEGMENTS AND CONIC ARCS

IBRAHIM ADAMOU, MARIO FIORAVANTI, AND LAUREANO GONZALEZ–VEGA

ABSTRACT. A new approach is presented for computing the medial axis of a planar, closed and bounded domain whose boundary consists of finitely many segments and conic arcs. The new method is topologically correct (no components are missed) and geometrically exact (each component is represented exactly).

INTRODUCTION

Let $\mathcal{D}$ be bounded domain in $\mathbb{R}^2$ with boundary $\mathcal{C}$ consisting of a finite number of curve segments. The medial axis of $\mathcal{D}$, denoted $\mathcal{M}(\mathcal{D})$, can be geometrically defined as the closed locus of the centers of all maximal circles inside $\mathcal{D}$ which are tangent at least at two different points in the boundary of $\mathcal{D}$, i.e:

$$\mathcal{M}(\mathcal{D}) = \{P \in \mathcal{D}\text{: there exists } P_1, P_2 \in \mathcal{C} \text{ such that } P_1 \neq P_2,\ d(P, P_1) = d(P, P_2)\}.$$

If $\mathcal{C}$ is given by a parametrization $\mathcal{C}(u)$ ($u \in [a, b]$, $\mathcal{C}(a) = \mathcal{C}(b)$ and $\mathcal{C}$ continuous and differentiable except in a finite number of points), $\mathcal{M}(\mathcal{D})$ can be defined by

$$\mathcal{M}(\mathcal{D}) = \{P \in \mathcal{D}\text{: there exists } u_1, u_2 \in [a, b] \text{ such that } u_1 \neq u_2,\ d(P, \mathcal{C}(u_1)) = d(P, \mathcal{C}(u_2))\}.$$

The following equations give the conditions for a point $P \in \mathcal{D}$ to belong to the medial axis $\mathcal{M}(\mathcal{D})$: $P \in \mathcal{M}(\mathcal{D})$ if there exists parameter values $u_1, u_2 \in [a, b]$ such that

- P is at normals of $\mathcal{C}$ from $C_1 = \mathcal{C}(u_1)$ and $C_2 = \mathcal{C}(u_2)$:

$$\langle P - \mathcal{C}(u_1), \mathcal{C}'(u_1)\rangle = 0 \text{ and } \langle P - \mathcal{C}(u_2), \mathcal{C}'(u_2)\rangle = 0 \tag{1}$$

- P is at equal distance from $C_1 = \mathcal{C}(u_1)$ and $C_2 = \mathcal{C}(u_2)$:

$$\langle P, 2(\mathcal{C}(u_2) - \mathcal{C}(u_1))\rangle + \|\mathcal{C}(u_1)\|^2 - \|\mathcal{C}(u_2)\|^2 = 0 \tag{2}$$

- The points $\mathcal{C}(u_1)$ and $\mathcal{C}(u_2)$ are not equal: $\mathcal{C}(u_2) \neq \mathcal{C}(u_1)$.

$\mathcal{M}(\mathcal{D})$ is a collection of finitely many curve segments coming from the bisectors of any two curve segments in the boundary $\mathcal{C}$ of $\mathcal{D}$.

We introduce here a new approach determining the medial axis of $\mathcal{D}$ which is topologically correct (no components are missed) and geometrically exact (each component is represented exactly), the medial axis of $\mathcal{D}$ when its boundary $\mathcal{C}$ is a finite number of segments and conic arcs. This is achieved, first, by determining all possible "topologies" and exact representations for the bisector of two parametric curves which are either lines or conics, second, by analyzing what happens when previous computations are applied to segments and (bounded)

Partially supported by the Spanish Ministerio de Economía y Competitividad and by the European Regional Development Fund (ERDF), under the project MTM2014-54141-P.

conic arcs and, finally, by analyzing the arrangement of all those bisectors to derive the medial axis of $\mathcal{D}$ by keeping only those curves fulfilling the conditions presented previously in (1) and (2). Compared with [1], we only deal with segments and (bounded) conic arcs and all curves required to determine the medial axis are presented exactly and not approximated but we follow the same strategy in order to get the final result by trimming and composing the involved biectors.

1. Curve–curve and point–curve bisectors for lines and conics

Next two tables show the characteristics of the curve–curve and point–curve bisectors for lines and conics. For each couple of possibilities the table shows the degree of the implicit equation of the considered bisector, D, together with its character (rational or not): RP means that it is rational, NRP means that can be parametrized using square roots and NP means that the only available representation of the bisector is its implicit equation.

Case	Parametrization	D
line–line	RP	2
line–circle	RP	4
line-ellipse	NRP	8
line–hyperbola	NRP	8
line–parabola	RP	6
circle–circle	RP	4
circle–ellipse	NRP	12
circle–hyperbola	NRP	12
circle–parabola	RP	10
ellipse–ellipse	NP	28
ellipse–hyperbola	NP	28
ellipse–parabola	NP	22
hyperbola–hyperbola	NP	28
hyperbola–parabola	NP	22
parabola–parabola	NP	17

Table 1. Curve–curve bisector for lines and conics.

Case	Parametrization	D
point–line	RP	2
point–circle	RP	2
point–ellipse	RP	6
point–parabola	RP	5
point–hyperbola	RP	6

Table 2. Point–curve bisector for lines and conics.

These are the curves to use when constructing the bisector of two bounded parametric curves which are either segments or conic arcs: such a bisector is a union of finitely may arcs extracted from the rows in these two tables (see next section). But, apart of the algebraic representation for these arcs, focusing the attention on lines and conics it is possible to characterize all possible bisector topologies. Next, two examples are presented one for each case: curve–curve and point–curve.

The the bisector of a line and an ellipse $e(t) = (a(t), b(t))$ is given by a parametrization involving square roots, $(a'(t)^2 + b'(t)^2)^{1/2}$, its implicit equation has degree 8 and it is irreducible. Depending on the relative position of the line and the ellipse, the three unique possibilities for the bisector topology can be found in Figure 1. In general, the situation can

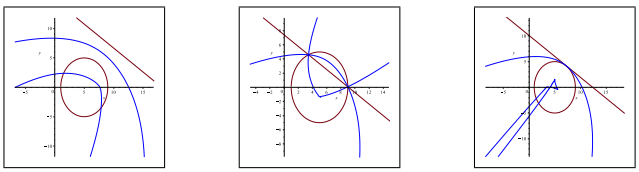


FIGURE 1. All possible configurations for the line–ellipse bisector (in blue).

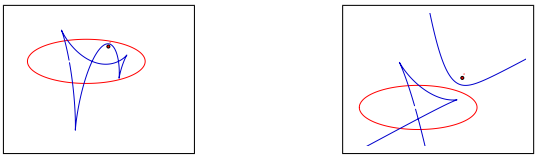


FIGURE 2. All possible configuration for the point–ellipse bisector (in blue).

be much more complicated: for the circle–circle case there are 5 possible topologies and this number becomes 9 for the circle–ellipse case.

The bisector of a point and an ellipse is given by a rational parametrization and its implicit equation has degree 6. Depending on the relative position of the point with respect to the ellipse, the two possibilities for the bisector topology can be found in Figure 2.

In all cases the implicit equation and/or the parametrization for the bisector have been computed by using the techniques in [2, 3, 4, 5].

2. Curve–curve bisector for segments and conic arcs

Since we are dealing with the problem of computing the medial axis, we restrict here our attention to the case where the two considered arcs whose bisector is requiered are either disjoint or they intersect in the endpoints. Let $s_1(u)$, $(u \in [a_1, b_1])$ and $s_2(t)$, $(t \in [a_2, b_2])$ be the parametric two curves whose bisector is to be computed. Using the equations (1) we obtain for P a description $B(u,t)$ that, after replacement in the equation 2, produces the following relation for the values of u and t when they generate, as footpoints, a point in the bisector:

$$h(u,t) = \langle B(u,t), 2(s_1(u) - s_2(t))\rangle + \|s_2(t)\|^2 - \|s_1(u)\|^2 = 0. \tag{3}$$

The intersection of $h(u,t) = 0$ with the boundary of $[a_1, b_1] \times [a_2, b_2]$ together with some of the non bounded branches of the bisectors produces the searched bisector (see [3, 6]). Figure 3 shows how the initial curves need to be partitioned in order to determine their bisector.

3. Medial axis computation

Let the boundary $\mathcal{C}$ of $\mathcal{D}$ consists of finitely many bounded segments and conic arcs $\mathcal{C}_i$, $i \in \{1, 2, \ldots, n\}$. Analyzing the arrangement of the bisectors $\mathcal{S}_{i,j}$ for $\mathcal{C}_i$ and $\mathcal{C}_j$ with $i \neq j$ inside $\mathcal{D}$ produces the medial axis (see [1, 6, 7]). Figure 4 shows one example: checking all possible arcs in the arrangement produces the medial axis after keeping only those verifying the conditions in medial axis definition (it is enough to check one point in each arc in order to select it or to discard it).

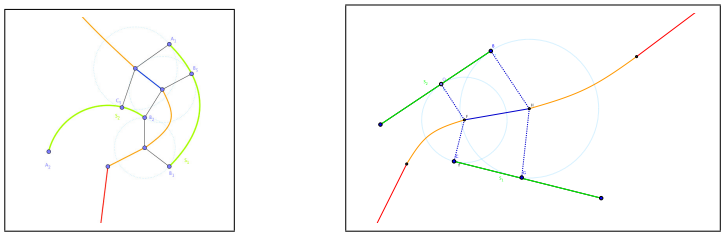


FIGURE 3. The bisector of two curve segments sharing no points: blue part comes from the curve–curve bisector and from the analysis of $h(u,t)=0$, red part comes from the bisector of two endpoints and orange part is generated from the point–curve bisector of one endpoint and the other curve.

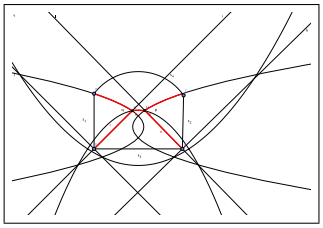


FIGURE 4. The domain $\mathcal{D}$ with all bisectors $\mathcal{S}_{i,j}$ $(1 \leq i \neq j \leq 4)$. The medial axis of $\mathcal{D}$, $\mathcal{M}(\mathcal{D})$, consist of the red arcs (2 segments and 3 parabolic arcs).

REFERENCES

[1] O. Aichholzer, W. Aigner, F. Aurenhammer, T. Hackl, B. Jüttler, M. Rabl: *Medial axis computation for planar freeÐform shapes*. CAD 41, 339Ð349,2009.

[2] I. Adamou: *Curvas y Superficies Bisectrices y Diagrama de Voronoi de una familia finita de semirrectas paralelas en* $\mathbb{R}^3$. PhD Thesis, Universidad de Cantabria, 2013.

[3] G. Elber, M.–S. Kim: *Bisector curves of planar rational curves*. CAD 30, 1089–1096, 1998.

[4] R. T. Farouki, J. K. Johnstone: *The bisector of a point and a plane parametric curve*. CAGD 11, 117–151, 1994.

[5] R. T. Farouki, J. K. Johnstone: *Computing point/curve and curve/curve bisectors*. Proceedings of the 5th IMA Conference on the Mathematics of Surfaces, 327–354, Clarendon Press, 1994.

[6] R. T. Farouki, R. Ramamurthy: *Specified–precision computation of curve/curve bisectors*. International Journal of Computational Geometry & Applications 8, 599–617, 1998.

[7] R. T. Farouki, R. Ramamurthy: *Degenerate point/curve and curve/curve bisectors arising in medial axis computations for planar domains with curved boundaries*. CAGD 15, 615–635, 1998.

Université de Maradi, Niger
E-mail address: adamouhakime14@gmail.com

Universidad de Cantabria, Spain
E-mail address: mario.fioravanti@unican.es

Universidad de Cantabria, Spain
E-mail address: laureano.gonzalez@unican.es

A CONSTRUCTIVE APPROACH TO THE REAL RANK OF A BINARY FORM

M. ANSOLA, A. DÍAZ-CANO, AND M^{a} A. ZURRO

Abstract. We will show two points associated with the real Waring Problem. First, we are going to exhibit an algorithm to obtain a Waring decompositon of a real binary form. Afterwards, we will assay some non trivial examples following the Sylvester Theorem for canonical forms of degree $d = 5$.

Introduction

The aim of this work is to study the decomposition of real symmetric tensors of dimension 2 and order d. In fact, we are going to study the decomposition of homogeneous polynomials of degree d in two variables as sum of r dth-powers of real linear forms.

Symmetric tensor decomposition (also called the Waring Problem) arise in signal and image processing, automatic control and many problems in Electrical Engineering. This problem is normally studied over $\mathbb{C}$ and it looks for the least integer number r so that it is possible to write a multivariate form of degree d as a sum of r dth-powers of (complex) linear forms. This number r is called the Waring rank or symmetric tensor rank of the form. Many authors have studied this problem over the complex numbers or over algebraically closed fields ([4],[5]) and some of them have tackled it for real forms, in particular, real binary forms ([1],[6]).

We are going to present a construction for a Waring decomposition of a real binary form. We are also going to classify the canonical forms of degree 5 given in [6] by means of their real ranks. The notion of *generic form* in [6] is coarser than our approach to study typical ranks.

1. An algorithm for constructing a real Waring decomposition

Let us consider any real binary form $p(x, y)$ of degree d in the variables x, y. Let it be $p(x, y) = p_{\underline{c}}(x, y) = \sum_{i=0}^{d} \binom{d}{i} c_i x^i y^{d-i}$, where $\underline{c} = (c_0, \ldots, c_d) \in \mathbb{R}^{d+1}$, except for all of them zero at the same time.

A Theorem due to Sylvester (see Theorem 2.1 in [4] and the references therein) stands that p can be written like a finite sum of dth-powers of complex linear forms,

$$p(x, y) = \sum_{j=1}^{r} \lambda_j (\alpha_j x + \beta_j y)^d,$$

if and only if

Second author partially supported by Spanish MTM2014-55565 and Grupo UCM 910444.
Third author partially supported by Grupo UCM 910444.

(1) There exists a vector $\underline{q} = (q_0, \cdots, q_r)$ such that

$$\begin{pmatrix} c_0 & c_1 & \cdots & c_r \\ c_1 & c_2 & \cdots & c_{r+1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{d-r} & c_{d-r+1} & \cdots & c_d \end{pmatrix} \begin{pmatrix} q_0 \\ q_1 \\ \vdots \\ q_r \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

(2) And the form $q(x,y) = \sum_{l=0}^{r} q_l x^l y^{r-l}$ factors as a product of r distinct linear forms. In fact, $q(x,y) = \prod_{j=1}^{r}(\beta_j x - \alpha_j y)$.

The real case.

Case d odd.

Let $\ell = (d-1)/2$. For $i \in \{1, \cdots, \ell\}$, let us consider real variables S_i and the matrix

$$V_{d+1} = \begin{pmatrix} X_0 & 1 & 1 & \cdots & 1 & 1 & c_d \\ X_1 & S_1 & -S_1 & \cdots & S_\ell & -S_\ell & c_{d-1} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ X_d & S_1^d & (-1)^d S_1^d & \cdots & S_\ell^d & (-1)^d S_\ell^d & c_0 \end{pmatrix}.$$

We expand its determinant as

$$h(X_0, \ldots, X_d) = \Delta_0 X_0 + \Delta_1 X_1 + \cdots + \Delta_d X_d \in \mathbb{R}[S_1, \cdots, S_\ell, X_0, \cdots, X_d]$$

with $\Delta_i \in \mathbb{R}[S_1, \cdots, S_\ell]$. Let $R = -\Delta_{d-1}/\Delta_d$. Take the real algebraic set

$$\mathrm{C} = \bigcup_{i=1}^{\ell} \{S_i = 0\} \cup \{\Delta_d = 0\} \cup \left\{ \bigcup_{i=1}^{\ell} \{\Delta_{d-1} \pm S_i \Delta_d = 0\} \right\} \cup \left\{ \bigcup_{i<j} \{S_i \pm S_j = 0\} \right\}$$

and consider the open set $\mathcal{G}_\ell = \mathbb{R}^\ell \backslash \mathrm{C}$. Now, for $(s_1, \cdots, s_\ell) \in \mathcal{G}_\ell$, the real system

$$(\star) \qquad \begin{pmatrix} 1 & 1 & \cdots & 1 & 1 & 1 \\ s_1 & -s_1 & \cdots & s_\ell & -s_\ell & R \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ s_1^d & (-1)^d s_1^d & \cdots & s_\ell^d & (-1)^d s_\ell^d & R^d \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_d \end{pmatrix} = \begin{pmatrix} c_d \\ c_{d-1} \\ \vdots \\ c_0 \end{pmatrix}$$

has a solution[1]. And then $p(x,y) = \sum_{j=1}^{d} \lambda_j L_j^d(x,y)$, with $L_j(x,y) = x + (-1)^{j+1} s_{\lfloor \frac{j+1}{2} \rfloor}\, y$ for $j \in \{1, \cdots, d-1\}$ and $L_d(x,y) = x + Ry$.

Case d even. In this situation, we are going to change the system $(\star)$ by

$$(\star\star) \qquad \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & s_1 & -s_1 & \cdots & s_{\ell-1} & -s_{\ell-1} & R \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & s_1^d & (-1)^d s_1^d & \cdots & s_{\ell-1}^d & (-1)^d s_{\ell-1}^d & R^d \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_d \end{pmatrix} = \begin{pmatrix} c_d \\ c_{d-1} \\ \vdots \\ c_0 \end{pmatrix}$$

and we proceed as in the previous case.

[1]Since $h(1, R, R^2, \cdots, R^d) = 0$.

Hence we have a Waring decomposition for our real form p with at most d summands. This statement can be seen with a different approach in [6], Proposition 2.1.

2. The degree $d = 5$

2.1. **The bezoutian matrix.** Let $u(t) = \sum_{i=0}^{n} u_i t^i$ and $v(t) = \sum_{i=0}^{n} v_i t^i$, two real polynomials in a variable t. The ***Hankel's Bezoutian*** or, simply, ***Bezoutian*** of u and v is the $n \times n$ matrix $B_n(u, v) = (b_{ij})_{1 \leq i,j \leq n}$, where the b_{ij} are given by the formula:

$$Bez_H(u, v) = \frac{u(t)v(s) - u(s)v(t)}{t - s} = \sum_{i,j=1}^{n} b_{ij}\, t^{i-1} s^{j-1} \tag{1}$$

We will use the following result (see [3] and also [7])

Theorem 2.1 (Borchardt-Jacobi Theorem)**.** *The number of distinct real roots of a real polynomial q of degree r is equal to the signature of the matrix $B_r(q, q')$, where q' stands for the usual derivate of q.*

2.2. In this section we are going to present the calculus of the real rank of the canonical forms proposed by Common and Ottaviani in [6] for the degree $d = 5$.

(1) **Type 1**. $p\,(x, y) = x(x^2 + y^2)(x^2 + 2axy + by^2)$.

By direct calculus, we obtain

a	0	0	$a^2 = b - b^2$	0	0	0	$\mathbb{R}\backslash\{0\}$
b	$9 \pm 4\sqrt{5}$	$\mathbb{R}\backslash\{-1, 0, 9 \pm 4\sqrt{5}\}$	$(\frac{1}{5}, 1)$	0	-1	$(-1, 0)$	0
$rk_{\mathbb{R}}$	2	3	3	4	4	4	4

Now, if $a \neq 0, a^2 \neq b - b^2$, we will consider the Hankel matrix

$$H_3 = \begin{pmatrix} 0 & \frac{b}{5} & \frac{a}{5} & \frac{b+1}{10} \\ \frac{b}{5} & \frac{a}{5} & \frac{b+1}{10} & \frac{2a}{5} \\ \frac{a}{5} & \frac{b+1}{10} & \frac{2a}{5} & 1 \end{pmatrix}$$

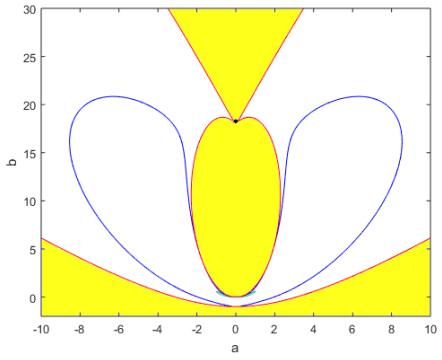
Since $\operatorname{Ker} H_3$ is a one-dimensional space, we have a generator, q, only depending on a and b. And then, the semialgebraic region E where $B_3(q, q')$ is positive definite gives us the canonical forms of type 1 with real rank 3. This region is described by three inequations: $f_1(a, b) > 0, f_2(a, b) > 0, f_3(a, b) > 0$, where $f_3(a, b) = 16 I_{12}(a, b)(16a^2b + 24a^2 + b^3 - 17b^2 - 17b + 1)^2$, where $I_{12}(a, b)$ is the algebraic curve of degree 12 given by Common and Ottaviani in [6].

In the figure [1] we can observe the region E.

The curve defining the border of E has an interesting singularity near the origin. This part is not studied in [6]. We include a zoom picture (figure [2]) of that part of E. The remaining cases are of rank 4 since in this kind of canonical forms we never get rank 5.

(2) **Type 2**. $p(x, y) = x(x^2 - y^2)(x^2 + 2axy + by^2)$.

In this case we proceed analogously to the type 1. We just point out that the polynomials defining the semialgebraic sets are also of degree 12 and that Bezoutians are also used to determine them.



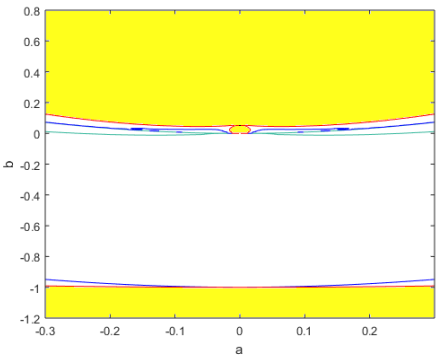


FIGURE 1. Region associated with rank 3 for a canonical form of type 1.
FIGURE 2: Detail of the region near the origin

Acknowledgement. We thank Prof. L. González Vega who for the first time showed us that the picture of the curve defined by the polynomial $I_{12} = 0$ in [6] could be improved. He used some topological facts to study this.

REFERENCES

[1] Ballico, E., 2014. On the typical rank of real bivariate polynomials. Linear Algebra Appl., vol. 452, pp. 263-269.
[2] Blekherman, G., 2015. Typical Real Ranks of Binary Forms. Found. Comput. Math., June 2015, Vol. 15, Issue 3, pp. 793-798.
[3] Borchardt, C.W., 1847. Développements sur l'équation à l'aide de laquelle on détermine les inégalités séculaires du mouvemeut des planètes. J. Math. Pures Appl. Série 1, tome 12, pp. 50-67. `http://portail.mathdoc.fr/JMPA/afficher_notice.php?id=JMPA_1847_1_12_A3_0`.
[4] Brachat, J. Comon, P., Mourrain, B., Tsigaridas, E., 2010. Symmetric tensor decomposition. Linear Algebra Appl., 433, 1851-1872.
[5] Carlini, E., Catalisano, M.V., Geramita, A. V., 2012. The solution to the Waring problem for monomials and the sum of coprime monomials. J. Algebra, 370, 5-14.
[6] Comon, P., Ottaviani, G., 2012. On the typical rank of real binary forms. Linear Multilinear A., vol. 60, (6), 657-667.
[7] Fuhrmann, P.A., 2010. A Polynomial Approach to Linear Algebra. Universitext. Springer.
[8] Reznick, B., 2013. On the length of binary forms, Quadratic and Higher Degree Forms. (K. Alladi, M. Bhargava, D. Savitt, P. Tiep, eds.), Developments in Math. 31 (2013), Springer, New York, pp. 207-232.

Universidad Complutense de Madrid
E-mail address: `mansola@ucm.es`

Universidad Complutense de Madrid. Facultad de Matemáticas. IMI and Dpto. de Álgebra
E-mail address: `adiazcan@ucm.es`

Universidad Autónoma de Madrid
E-mail address: `mangeles.zurro@uam.es`

VERIFIED COMPUTER LINEAR ALGEBRA

JESÚS ARANSAY AND JOSE DIVASÓN

Abstract. We present the execution tests and benchmarks of some Linear Algebra programs generated from their verified formalisation in Isabelle/HOL; more concretely, the Gauss-Jordan algorithm and the QR decomposition, together with the techniques used to improve the performance of the extracted code, are described.

Introduction

Computer Algebra systems are commonly seen as *black boxes* in which one has to trust, but they are no error-free [6]. Theorem provers are designed to prove the correctness of algorithms and mathematical results, but this task is far from trivial and it has a significant cost in terms of performance. One of the most accepted techniques to verify a program is to describe the algorithm within the language of the proof checker, then extract code and run it independently. Following this strategy, we have formalised some well-known Linear Algebra algorithms in Isabelle/HOL and then code is extracted to SML and Haskell. In this paper, we briefly present the techniques that we followed to improve the performance of the generated code, as well as some execution tests. This code cannot compete with Computer Algebra systems in terms of efficiency, but it pays off in feasibility and the results also show the code to be useful for matrices of considerable size.

1. Verified Computing

Isabelle is a generic interactive theorem prover, in the sense that different logics can be implemented on top of it. The most widespread logic is HOL (Higher-Order Logic, whose Isabelle implementation is referred to as Isabelle/HOL), which includes interesting features such as code generation. The HOL Multivariate Analysis (or *HMA* for short) library is a set of Isabelle/HOL theories which contains theoretical results in mathematical fields such as Analysis and Linear Algebra. It is based on the work by Harrison in HOL Light and one of the fundamentals of the library is the representation of n-dimensional vectors (type `vec`) by means of functions from a finite type [7]. We have formalised several algorithms and their applications (Gauss-Jordan algorithm, QR decomposition...) based on the *HMA* library [4, 5], all of them are defined making use of the representation based on `vec` (functions over finite domains). Following the *data refinement* strategy, we have refined the algorithms to the more efficient representation `iarray`, which is later code-generated to its corresponding implementation in SML (*Vector.vector*) and Haskell (*IArray.array*). This representation defines polymorphic vectors, immutable sequences with constant-time access. Furthermore, *serialisations* are a process to map Isabelle types and operations to the corresponding ones in the target languages. They are common practice to avoid Isabelle generating from scratch. We focus our work on $\mathbb{Z}_2$, $\mathbb{Q}$, and $\mathbb{R}$ (types `bit`, `rat`, and `real` in Isabelle). We serialised them

to their corresponding structures in SML and Haskell (see [3] for further details). Let us note that `real` can be serialised to both fractions of integers (obtaining arbitrary precision) and floating-point numbers in the target languages. In the latter case, although the original algorithm is formalised, the computations cannot be trusted.

2. Experimental Outcomes

Let us present the performance tests that we have carried out to our verified programs obtained from the formalisation of the Gauss-Jordan algorithm [4] and the QR decomposition [5]. The times presented throughout the tables are expressed in seconds. The benchmarks have been carried out in a laptop with an Intel Core i5-3360M processor, 4 GB of RAM, Poly/ML 5.6, Ubuntu 14.04, GHCi 7.6.3, and Isabelle2016. We have noticed that Poly/ML, which is an interpreter, performs as well as an optimiser compiler as MLton when executing our generated code (times are similar, so we just present here the Poly/ML ones).

2.1. **The Gauss-Jordan algorithm.** We have formalised a version of the well-known Gauss-Jordan algorithm to compute the *reduced row echelon form* (from here on, rref) of a matrix in Isabelle/HOL, as well as its applications such as the computation of ranks and determinants. The algorithm has been formalised over an arbitrary field. Some preliminary experiments had been already carried out in Poly/ML [8], but developers of the compiler suggested us improvements that eliminated the *processing* time of the input matrices (which showed to be the real bottleneck, see the figures in [2]). More concretely, in our first experiments, the input matrices were directly introduced in the system by means of an explicit binder as static data. The Poly/ML maintainer also modified the system behaviour in the SVN version of the tool to improve the processing capabilities of big inputs. From our side, we changed our methodology to input matrices from external files by means of an ad-hoc parser. Processing input matrices this way proved to be no time consuming. In addition, we have serialised the `bit` type in Isabelle to the booleans in SML and Haskell, whereas in the results presented in [2] the `bit` type was serialised to integers. The experimental tests presented here shows that this change provides a significant improvement of the computation times (the computation of the *rref* of a 800×800 $\mathbb{Z}_2$ matrix needed 43.9s in Poly/ML, now only 15.96s). Let us show a fragment of the experiments carried out with the new methodology. The input matrices can be downloaded from [1]. Table 1 presents the times of computing the *rref* of randomly generated $\mathbb{Z}_2$ matrices. The same randomly generated matrices have been used across the different systems. It is well-known that the Gauss-Jordan algorithm has arithmetic complexity of $\mathcal{O}(n^3)$. The computing times of our programs also grow cubically (in both SML and Haskell) with respect to the number of elements in the input matrices (let us note that the underlying field notably affects the performance and can even affect the complexity bounds).

Size (n)	**Poly/ML**	**Haskell**
100	0.04	0.36
200	0.25	2.25
400	2.01	17.17
800	15.96	131.73
1200	62.33	453.57
1600	139.70	1097.41
2000	284.28	2295.30

Table 1. Time to compute the *rref* of randomly generated $\mathbb{Z}_2$ matrices.

An interesting case appears when working with matrices over $\mathbb{Q}$. Following the standard code generator setup to SML (serialising `rat` to fractions of *IntInf.int*), we detected

that the greatest amount of time was spent in reducing fractions (operations `gcd` and `divmod`). Serialising the Isabelle operation `gcd` to the corresponding built-in Poly/ML function (which is not part of the SML Standard Library, but particular to the compiler), decreased by a factor of 20 the computing times. In addition, the natural serialisation for the Isabelle operation `divmod` would be *IntInf.divmod* in SML, which returns the pair (*i IntInf.div j*, *i IntInf.mod j*) where the result of *div* is truncated toward negative infinity (for example, `divmod (-10, 6)` returns `(-2, 2)`). However, in SML (and Haskell) there also exists a more efficient operation *IntInf.quotrem*, which returns the pair (*i IntInf.quot j*, *i IntInf.rem j*) where *quot* is integer division truncated toward zero (for instance, `quotrem (-10, 6)` returns `(-1, -4)`). Since in the case of $\mathbb{Q}$ matrices we only divide when normalising fractions in SML (and thus, we only divide by divisors) the remainder is always 0, so we can serialise to the more efficient operation *IntInf.quotrem*. Table 2 presents the performance tests to compute determinants of randomly generated matrices over $\mathbb{Q}$. Apparently, Haskell takes advantage of its native *Rational* type to match the results of Poly/ML. Table 3 presents the times used to compute the *rref* of matrices over $\mathbb{R}$ represented as floating-point numbers. In this case, numerical stability problems arise (the *rref* of matrices contains small nonzero entries), as also happens in Computer Algebra systems. Once again, Poly/ML outperforms Haskell and the performance is cubic as well.

Size (n)	Poly/ML	Haskell
10	0.01	0.01
20	0.02	0.03
40	0.21	0.24
80	3.77	3.53

TABLE 2. Time to compute the *rref* of randomly generated $\mathbb{Q}$ matrices.

2.2. **The QR Decomposition.** We have also formalised the QR decomposition in Isabelle/HOL and its application to compute the *least squares approximation* to an unsolvable system of linear equations. The QR decomposition decomposes a real matrix A into the product of two different matrices Q and R (the first one containing an *orthonormal* collection of vectors, the second one being upper triangular). The QR decomposition is important, among other things, since it significantly reduces round-off errors when computing the least squares approximation (which can also be solved by means of the Gauss-Jordan algorithm). Since the Isabelle type `real` is used and square roots are involved in the algorithm (they are necessary to normalise the vectors), the use of a representation of `real` based on fractions of integers (*IntInf.int*) in SML is not possible (square roots are not computable in such a setting). A development by Thiemann was published in the Archive of Formal Proofs [9]. This development provides a refinement for real numbers of the form $p + q\sqrt{b}$ (with $p, q \in \mathbb{Q}$, $b \in \mathbb{N}$). We make use of this development to get exact symbolic computations. The performance obtained by means of this refinement depends much on the size of the entries. For example the computation of the QR decomposition of a 10×10 matrix requires several minutes.

Size (n)	Poly/ML	Haskell
100	0.03	0.38
200	0.25	2.62
400	1.85	19.51
800	13.99	148.20

TABLE 3. Time to compute the *rref* of randomly generated $\mathbb{R}$ matrices.

The other possibility is the use of the serialisation to floating-point numbers, which is specially interesting (despite the computations cannot be trusted) when comparing the precision obtained with the one of the Gauss-Jordan development. We present an

experiment involving the Hilbert matrix (which is known to be very ill-conditioned) in dimension 6, H_6. We have computed the least squares solution to the system $H_6\, x = (1\,0\,0\,0\,0\,0\,5)^T$ using both the QR decomposition and the Gauss-Jordan algorithm. The exact solution of the least squares approximation can be obtained symbolically: `"["-13824","415170","-2907240","7754040","-8724240","3489948"]"` . If we now use the refinement from Isabelle `real` to SML *floats*, and both algorithms to solve the least squares problem, the following solutions are obtained:

- QR solution using floats:
  `[-13824.0,415170.0001,-2907240.0,7754040.001,-8724240.001,3489948.0]`
- Gauss-Jordan solution using floats:
  `[-13808.64215,414731.7866,-2904277.468,7746340.301,-8715747.432,3486603.907]`

As it can be noticed, the QR decomposition is much more precise than the Gauss-Jordan algorithm. Table 4 shows the performance obtained with this serialisation.

3. Conclusions and Further Work

We have presented the results obtained in the execution of verified Linear Algebra programs generated from their formalisation in Isabelle/HOL, as well as some serialisations and data refinement devoted to improve the performance. The verified code cannot compete with Computer Algebra systems, but it is usable with matrices of remarkable dimensions. This is an attempt to try to reduce the existing gap between software verification and working software. As a future work, the study of the performance of our verified algorithms to compute the echelon form and the Hermite normal form of a matrix would be desirable. For the moment, the Hermite normal form of a 25×25 integer matrix can be completed in seconds, but we run out of memory in higher dimensions.

Size (n)	**Poly/ML**
100	0.748
160	4.621
220	18.941
280	42.100
340	97.360
400	183.754

Table 4. Time to compute the QR decomposition of Hilbert matrices over $\mathbb{R}$.

References

[1] `http://www.unirioja.es/cu/jodivaso/Isabelle/Gauss-Jordan-2013-2-Generalized/`.

[2] J. Aransay and J. Divasón. Performance Analysis of a Verified Linear Algebra Program in SML. In Fredlund and Castro, editors, *TPF 2013*, pages 28 – 35, 2013.

[3] J. Aransay and J. Divasón. Formalisation in higher-order logic and code generation to functional languages of the Gauss-Jordan algorithm. *J. of Func. Programming*, 25, 2015.

[4] J. Divasón and J. Aransay. Gauss-Jordan Algorithm and Its Applications. *Archive of Formal Proofs*, September 2014. `http://afp.sf.net/entries/Gauss_Jordan.shtml`, Formal proof development.

[5] J. Divasón and J. Aransay. QR Decomposition. *Archive of Formal Proofs*, 2015. `http://afp.sf.net/entries/QR_Decomposition.shtml`, Formal proof development.

[6] A. J. Durán, M. Pérez, and J. L. Varona. Misfortunes of a mathematicians' trio using Computer Algebra Systems: Can we trust? *Notices of the AMS*, 61(10):1249 – 1252, 2014.

[7] J. Harrison. The HOL Light Theory of Euclidean Space. *J. of Autom. Reasoning*, 50(2):173 – 190, 2013.

[8] The Poly/ML website. `http://www.polyml.org/`, 2015.

[9] R. Thiemann. Implementing field extensions of the form $\mathbb{Q}[\sqrt{b}]$. *Archive of Formal Proofs*, February 2014. `http://afp.sf.net/entries/Real_Impl.shtml`, Formal proof development.

Universidad de La Rioja
E-mail address: `{jesus-maria.aransay,jose.divasonm}@unirioja.es`

COMPUTING JUMPING NUMBERS IN HIGHER DIMENSIONS

HANS BAUMERS AND FERRAN DACHS-CADEFAU

Abstract. We design an algorithm that computes a small set containing the jumping numbers of an ideal in a regular local ring of arbitrary dimension. We also provide some criteria to decide whether these numbers are jumping numbers.

Introduction

The jumping numbers of an ideal sheaf on a smooth algebraic variety are very interesting geometric invariants, that were studied in [5], but also appeared earlier in [10], [12], and [15]. They indicate in some sense how bad a singularity is, and are determined by the exceptional divisors in a resolution of the ideal.

Several algorithms have been developed to compute jumping numbers in specific settings. Tucker [14] designed an algorithm to compute jumping numbers on surfaces with rational singularities. Alberich-Carramiñana, Àlvarez-Montaner and the second author [1] introduced another algorithm in that setting. Berkesch and Leykin [3] and Shibuta [13] developed algorithms using Bernstein-Sato polynomials.

We present a generalization of the algorithm in [1], that can be used for computing jumping numbers on higher-dimensional varieties. The idea is to compute a small subset of the candidate jumping numbers, containing all the jumping numbers, and then, in many cases, one can check whether these numbers are jumping numbers.

1. Multiplier ideals and jumping numbers

In this first section, we introduce some basic notions, such as multiplier ideals, jumping numbers and contribution. Let R be a regular local ring over $\mathbb{C}$ such that $X = \operatorname{Spec} R$ is the germ of a smooth algebraic variety. We will be particularly interested in the case where $d := \dim X \geqslant 3$, since the two-dimensional case has been worked out completely in [1].

Let $\mathfrak{a} \subset \mathcal{O}_X$ be a sheaf of ideals on X, and take a log resolution $\pi : Y \to X$ of $\mathfrak{a}$. Denote by K_π the relative canonical divisor of π, and by F the normal crossings divisor on Y satisfying $\mathfrak{a} \cdot \mathcal{O}_Y = \mathcal{O}_Y(-F)$. The *multiplier ideal* of $(X, \mathfrak{a})$ with coefficient $c \in \mathbb{Q}_{>0}$ is defined as

$$\mathcal{J}(X, \mathfrak{a}^c) = \pi_* \mathcal{O}_Y(K_\pi - \lfloor cF \rfloor).$$

There exists a sequence of numbers $0 < \lambda_1 < \lambda_2 < \dots$ such that, for all i, we have that $\mathcal{J}(X, \mathfrak{a}^{\lambda_i}) \supsetneq \mathcal{J}(X, \mathfrak{a}^{\lambda_{i+1}})$, and $\mathcal{J}(X, \mathfrak{a}^c)$ is constant for $c \in [\lambda_i, \lambda_{i+1})$. These numbers are called the *jumping numbers* of the pair $(X, \mathfrak{a})$. The multiplier ideals and jumping numbers do not depend on the chosen resolution.

The first author is supported by a PhD fellowship of the Research Foundation - Flanders (FWO). The second author was partially supported by Generalitat de Catalunya SGR2014-634 project, Spanish Ministerio de Economía y Competitividad MTM2015-69135-P and by the KU Leuven grant OT/11/069.

Note that if we write $K_\pi = \sum_{i\in I} k_i E_i$ and $F = \sum_{i\in I} e_i E_i$, where the E_i are the irreducible components of F, then the set $\left\{ \frac{k_i+n}{e_i} \middle| i \in I, n \in \mathbb{Z}_{>0} \right\}$ contains the jumping numbers. The numbers in this set are called *candidate jumping numbers*. The smallest candidate is always a jumping number, and is called the *log canonical threshold*. We say λ is a *candidate for* $G = E_1 + \cdots + E_r$ if λ can be expressed as $\frac{k_i+n_i}{e_i}$ for $i = 1, \ldots, r$, where $n_i \in \mathbb{Z}_{>0}$.

If λ is a jumping number, and $G = E_1 + \cdots + E_r$ is a divisor such that λ is a candidate for G, we say G *contributes* λ if $\mathcal{J}(X, \mathfrak{a}^\lambda) \subsetneq \pi_*\mathcal{O}_Y(K_\pi - \lfloor \lambda F \rfloor + G)$. This happens if and only if $H^0(G, \mathcal{O}_Y(K_\pi - \lfloor \lambda F \rfloor + G)|_G) \neq 0$.

2. π-antieffective divisors

Definition 2.1. Generalizing the notion of antinef divisors, we say that a divisor D on Y is π-*antieffective* if $H^0(E, \mathcal{O}_Y(-D)|_E) \neq 0$ for every π-exceptional prime divisor E. This is equivalent with saying that $-D|_E$ defines a class in $\operatorname{Pic} E$ that contains an effective divisor.

Given a divisor D, one can compute its π-*antieffective closure* by the *unloading procedure*, i.e., if $H^0(E, \mathcal{O}_Y(-D)|_E) = 0$ for some E, replace D by $D + E$, and continue until the obtained divisor $\tilde{D}$ is π-antieffective. This is a generalization of the unloading procedure for divisors on surfaces, described in [4], [6] or [8]. The π-antieffective closure satisfies $\pi_*\mathcal{O}_Y(-\tilde{D}) = \pi_*\mathcal{O}_Y(-D)$.

3. An algorithm to compute jumping numbers

The set of *supercandidates* is constructed as follows.

Algorithm 3.1 (Computing supercandidates). `Input:` An ideal $\mathfrak{a}$ and a resolution of $\mathfrak{a}$. `Output:` The set of supercandidates with their minimal jumping divisors.

- The first supercandidate is the log canonical threshold.
- If λ is a supercandidate, then the next supercandidate is $\lambda' = \min\left\{ \frac{k_i+1+e_i^\lambda}{e_i} \middle| i \in I \right\}$, where $D_\lambda := \sum_{i\in I} e_i^\lambda E_i$ is the π-antieffective closure of $\lfloor \lambda F \rfloor - K_\pi$. The *minimal jumping divisor* of λ' is the reduced divisor $G_{\lambda'}$ supported on those E_i where this minimum is achieved.

Theorem 3.2. *The set of supercandidates contains all the jumping numbers.*

Proof. This follows from the fact that

$$\mathcal{J}(X, \mathfrak{a}^\lambda) = \pi_*\mathcal{O}_Y(K_\pi - \lfloor \lambda F \rfloor) = \pi_*\mathcal{O}_Y(-D_\lambda),$$

so there can be no jumping numbers between two consecutive supercandidates. □

If $\dim X = 2$, the converse also holds. This is a consequence of Lipman's result in [11, Section 18], stating that there is a one-on-one relation between integrally closed ideals and π-antieffective divisors. In higher dimensions, different π-antieffective divisors might determine the same ideal. Therefore, we might have supercandidates that are not jumping numbers. However, in several cases, we can check that a supercandidate is a jumping number.

Proposition 3.3. *If λ is a supercandidate such that G_λ has an irreducible connected component, then λ is a jumping number.*

This is a very important case, since in many situations, a significant number of supercandidates seem to have an irreducible jumping divisor.

Proposition 3.4. *If λ is a jumping number, it is contributed by G_λ, and hence there is a minimal contributing divisor $G \leqslant G_\lambda$.*

So if we want to check whether a supercandidate is a jumping number, we only need to check contribution by divisors $G \leqslant G_\lambda$. This seems to be hard in general when G_λ is reducible, but the following result can help.

Proposition 3.5. *If λ is a candidate for $G = E_1 + E_2$, and if $\mathcal{O}_Y(K_\pi - \lfloor \lambda F \rfloor + G)|_{E_i} \cong \mathcal{O}_{E_i}$ for $i \in \{1, 2\}$, then λ is a jumping number contributed by $E_1 + E_2$.*

All together, we get the following algorithm.

Algorithm 3.6 (Computing jumping numbers)**.** `Input:` An ideal $\mathfrak{a}$ and a resolution of $\mathfrak{a}$. `Output:` The set of jumping numbers of $\mathfrak{a}$.

- Compute the supercandidates λ, along with their minimal jumping divisors G_λ.
- If G_λ has an irreducible connected component, λ is a jumping number.
- Otherwise, check whether λ is a jumping number.

By Skoda's theorem [9, Theorem 9.6.21], it suffices to compute the supercandidates in $(0, d]$, or even in $(0, n]$, where n is the number of generators of $\mathfrak{a}$.

Remark 3.7. It can be hard in general to determine whether a linear equivalence class contains an effective divisor, or to decide about the existence of a global section on reducible divisors. This might complicate the unloading procedure and make it hard to check whether a supercandidate is a jumping number when Proposition 3.5 does not apply. However, in many examples, the provided results suffice to determine all the jumping numbers.

Remark 3.8. Apart from the obstructions mentioned in Remark 3.7, the algorithm can be implemented as follows. For the computation of a log resolution, one could use the algorithm of [7], implemented in the packages "resolve.lib" and "reszeta.lib" in Singular. If we are able to describe the effective cones of the exceptional divisors in the resolution, the computation of supercandidates and their minimal jumping divisors is easy to implement.

Example 3.9. Let X be the germ of affine threespace around the origin, and $\mathfrak{a}$ the ideal generated by $f = x(yz - x^4)(x^4 + y^2 - 2yz) + yz^4 - y^5$. After six point blow-ups, we obtain a resolution $\pi : Y \to X$. We have $K_\pi = 2E_1 + 4E_2 + 8E_3 + 14E_4 + 6E_5 + 6E_6$ and $F = F_{aff} + 5E_1 + 9E_2 + 16E_3 + 27E_4 + 11E_5 + 11E_6$, where the E_i are the exceptional divisors, numbered in order of creation, and F_{aff} is the strict transform of $\{f = 0\}$. The supercandidates in $(0, 1]$ are $\frac{5}{9}$, $\frac{2}{3}$, $\frac{20}{27}$, $\frac{7}{9}$, $\frac{23}{27}$, $\frac{8}{9}$, $\frac{25}{27}$, $\frac{26}{27}$ and 1, and G_λ equals E_4 for $\frac{20}{27}$, $\frac{23}{27}$, $\frac{25}{27}$ and $\frac{26}{27}$, and $E_2 + E_4$ for $\frac{5}{9}$, $\frac{2}{3}$, $\frac{7}{9}$ and $\frac{8}{9}$. Since the ideal is principal, 1 is a jumping number. By Proposition 3.3, $\frac{20}{27}$, $\frac{23}{27}$, $\frac{25}{27}$ and $\frac{26}{27}$ are jumping numbers contributed by E_4. The log canonical threshold $\frac{5}{9}$ is a jumping number, and using Proposition 3.5, one can see that $\frac{2}{3}$ is a jumping number contributed by $E_2 + E_4$. Finally, one can check that $\frac{7}{9}$ and $\frac{8}{9}$ are jumping numbers contributed by E_2. By Skoda's theorem, we know all the jumping numbers.

In this example, our method is clearly faster than naively checking for all candidate jumping numbers whether they are jumping numbers. The algorithm of [3], that is implemented in Macaulay2, did not give a result after several days of computing.

Example 3.10. Let X be as in the previous example, and let $\mathfrak{a}$ be the ideal generated by $f = (x^d + y^d + z^d)^2 + g(x, y, z)$, with $d \geqslant 3$ and $g(x, y, z)$ a homogeneous polynomial of degree $2d + 1$. To compute a resolution, we first blow up at the origin, and then at the $k = d(2d + 1)$ singular points on the strict transform of D. The exceptional divisors are denoted E_1 and E_i^p for $i = 1, \ldots, k$, respectively. After two more blow-ups, centered at a curve of genus $g = \frac{1}{2}(d-1)(d-2)$, with exceptional divisors E_2 and E_3, we have a resolution. We have $F = F_{aff} + 2dE_1 + (2d+2)\sum_{i=1}^k E_i^p + (2d+1)E_2 + (4d+2)E_3$ and $K_\pi = 2E_1 + 4\sum_{i=1}^k E_i^p + 3E_2 + 6E_3$. Since E_2 and E_3 are ruled surfaces over a curve of higher genus, it is not obvious to determine the classes in their Picard groups containing effective divisors. However, we can deduce enough information to run our algorithm. We find that the set of supercandidates in $(0, 1]$ is

$$\left\{ \frac{n}{2d} \,\middle|\, 3 \leqslant n < d \right\} \cup \left\{ \frac{2n+1}{4d+2} \,\middle|\, d \leqslant n \leqslant 2d \right\} \cup \left\{ \frac{n}{2d} \,\middle|\, d+3 \leqslant n \leqslant 2d \right\},$$

and all of them are jumping numbers contributed by E_1 or E_3.

References

[1] M. Alberich-Carramiñana, J. Alvarez Montaner, and F. Dachs-Cadefau. Multiplier ideals in two-dimensional local rings with rational singularities. *ArXiv:1412.3605, to appear in Mich. Math. J.*, 2014.

[2] H. Baumers and F. Dachs-Cadefau. Computing jumping numbers in higher dimensions. *ArXiv e-prints, arXiv:1603.00787*, 2016.

[3] C. Berkesch and A. Leykin. Algorithms for Bernstein-Sato polynomials and multiplier ideals. In *ISSAC 2010—Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, pages 99–106. ACM, New York, 2010.

[4] E. Casas-Alvero. *Singularities of plane curves*, volume 276 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2000.

[5] L. Ein, R. Lazarsfeld, K. E. Smith, and D. Varolin. Jumping coefficients of multiplier ideals. *Duke Math. J.*, 123(3):469–506, 2004.

[6] F. Enriques and O. Chisini. *Lezioni sulla teoria geometrica delle equazioni e delle funzioni algebriche*. N. Zanichelli, 1915.

[7] A. Frühbis-Krüger and G. Pfister. Algorithmic resolution of singularities. In *Singularities and computer algebra*, volume 324 of *London Math. Soc. Lecture Note Ser.*, pages 157–183. Cambridge Univ. Press, Cambridge, 2006.

[8] H. B. Laufer. On rational singularities. *Amer. J. Math.*, 94:597–608, 1972.

[9] R. Lazarsfeld. *Positivity in algebraic geometry. II*, volume 49 of *Ergebnisse der Mathematik und ihrer Grenzgebiete*. Springer-Verlag, Berlin, 2004.

[10] A. S. Libgober. Alexander invariants of plane algebraic curves. In *Singularities, Part 2 (Arcata, Calif., 1981)*, volume 40 of *Proc. Sympos. Pure Math.*, pages 135–143. Amer. Math. Soc., Providence, RI, 1983.

[11] J. Lipman. Rational singularities, with applications to algebraic surfaces and unique factorization. *Inst. Hautes Études Sci. Publ. Math.*, (36):195–279, 1969.

[12] F. Loeser and M. Vaquié. Le polynôme d'Alexander d'une courbe plane projective. *Topology*, 29(2):163–173, 1990.

[13] T. Shibuta. Algorithms for computing multiplier ideals. *J. Pure Appl. Algebra*, 215(12):2829–2842, 2011.

[14] K. Tucker. Jumping numbers on algebraic surfaces with rational singularities. *Trans. Amer. Math. Soc.*, 362(6):3223–3241, 2010.

[15] M. Vaquié. Irrégularité des revêtements cycliques. In *Singularities (Lille, 1991)*, volume 201 of *London Math. Soc. Lecture Note Ser.*, pages 383–419. Cambridge Univ. Press, Cambridge, 1994.

KU Leuven, Dept. of Mathematics, Celestijnenlaan 200B box 2400, 3001 Leuven, Belgium
E-mail address: Hans.Baumers@wis.kuleuven.be, Ferran.DachsCadefau@wis.kuleuven.be

COMPUTING IN LIE ALGEBRAS WITH SMALL NUMBER OF IDEALS

PILAR BENITO AND IVÁN PÉREZ-ARADROS

ABSTRACT. In this paper we focus on Lie algebras L in which the number of ideals, $n_I(L)$, is finite. Using simple classical Lie algebras of rank ≤ 2, we develop a computer algorithm which let us to construct nonsolvable Lie algebras such that $n_I(L) \leq 5$. The algorithm is based on preliminary structure results of this class of Lie algebras and on classical results on representation theory of simple Lie algebras.

INTRODUCTION

Let us denote as $n_I(L) \in \mathbb{Z}^+$ the number of ideals of a Lie algebra L. In the present work we are interested in the construction of Lie algebras in which $n_I(L)$ is finite by using computational algorithms. In [1] this problem is treated from a theoretical point of view for small values of $n_I(L)$. The results therein lead to the classification of the complex and real solvable Lie algebras such that $n_I(L) \leq 5$. The solvable Lie algebras satisfying the latter condition are of small dimension; they are up to dimension 7 in the case of real Lie algebras and up to dimension 4 in the case of complex Lie algebras. Hovewer, according to [2], there exist nonsolvable Lie algebras of arbitrary dimension such that $n_I(L) \leq 5$. This fact makes the classification of nonsolvable Lie algebras with a few number of ideals a problem hard to deal with.

In the structure of any Lie algebra, the representation theory of the simple ones plays an important role due to Levi's theorem:

Levi's Theorem. (E.E. Levi, 1905) *Any finite-dimensional Lie algebra L of characteristic zero decomposes as a sum of its solvable radical R, and a semisimple subalgebra S.*

A subalgebra S satisfying these conditions is called a *Levi factor of* L and the corresponding decomposition, $L = S \oplus R$, a *Levi decomposition* of L. The radical R is an ideal that can be viewed as an S-module by means of the restricted adjoint representation of S on R.

Any semisimple Lie algebra is expressible as a direct sum of ideals that are simple Lie algebras. The split 3-dimensional algebra $\mathfrak{sl}_2(\mathbb{C})$ and the 8 and 10-dimensional Lie algebras $\mathfrak{sl}_3(\mathbb{C})$ and $\mathfrak{sp}_4(\mathbb{C})$ are the classical simple Lie algebras of *rank* ≤ 2 over the complex field $\mathbb{C}$. The root space decomposition of any simple Lie algebra is controlled by $\mathfrak{sl}_2(\mathbb{C})$ which turns $\mathfrak{sl}_2(\mathbb{C})$ into a essential algebra.

Starting with $\mathfrak{sl}_2(k)$ and using some suitable $\mathfrak{sl}_2(k)$-invariant bilinear products $V_n \otimes V_m \to V_{m+n-2k}$ that appeared in [4], we propose computer algorithms for constructing Lie algebras with a small number of ideals. The algorithms are inspired in [3] and their specifications are presented in the final section 3. Previous sections 1 and 2 are devoted to set the basic facts on structure and representation theory of Lie algebras and the results on existence and constructions of Lie algebras with few ideals that are the theoretical environment of the algorithms. The basic results on Lie algebras follows from [6] and [5].

1. Preliminaries on Lie algebras

A Lie algebra L is a vector space over a field k endowed with a binary skew-symmetric ($\frac{1}{2} \in k$) bilinear product $[x, y]$ satisfying the Jacobi identity:

$$J(x, y, z) = [[x, y], z] + [[z, x], y] + [[y, z], x] = 0 \quad \forall_{x,y,z \in L}. \tag{1}$$

In case $[x, y] = 0\ \forall_{x,y,z \in L}$, the Lie algebra L is called *abelian*. For a given vector space V, the set $End\,(V)$ of linear endomorphisms of V, becomes a Lie algebra under the product $[f, g] = fg - gf$. This algebra is named the *general linear Lie algebra on* V and denoted as $\mathfrak{gl}(V)$. From Ado's Theorem and Ivasawa's Theorem, [6, Chapters III and VI], any finite-dimensional Lie algebra can be imbedded into a general Lie algebra.

The Lie bracket of two vector subspaces U, V of L is the whole linear span

$$[U, V] = span < [u, v] : u \in U, v \in V > . \tag{2}$$

An ideal I of L is a subspace such that $[I, L] \subseteq I$. A *simple* (*semisimple*) Lie algebra is a non abelian Lie algebra without proper ideals (that decomposes as a direct sum as ideals of simple Lie algebras). The derived series of L is defined recursively as $L^{(1)} = L$ and $L^{(n)} = [L^{(n-1)}, L^{(n-1)}],\ \forall n > 1$. The lower central series (l.c.s.) is defined as $L^1 = L$ and $L^n = [L, L^{n-1}],\ \forall n > 1$. If the derived (l.c.s.) series vanishes, L is called *solvable* (*nilpotent*). The *solvable radical* (*nilpotent radical*) of L, denoted as $R(L)$ (N(L)) is the biggest solvable (nilpotent) ideal of L. We also will denote these ideals as R or N.

For a vector space V, a representation of a Lie algebra L is an homomorphism of Lie algebras $\rho : L \to \mathfrak{gl}(V)$. The vector space V under the action $x \cdot v = \rho(x)(v)$ is called L-module. The module V is irreducible if it is nontrivial and does not contain proper submodules; in case the kernel of ρ is trivial, V is named a *faithful L-module*. New modules can be obtained from old ones: as *a quotient of a module by a submodule* in the natural way or as a tensor product $V \otimes W$ of modules by declaring: $x \cdot (v \otimes w) = (x \cdot v) \otimes w + v \otimes (x \cdot w)$.

2. Theoretical results and tools

Our algorithms are based on the Jacobi identity, the representation theory of $\mathfrak{sl}_2(k)$, and the structure results [2, Theorem 2.2] and [8, Theorem 2].

Theorem 2.1. *(Benito, 1994) Let L be a nonsolvable Lie algebra. Then, the ideals of L are in chain if and only if L is a simple Lie algebra or a direct sum of a nonzero nilpotent ideal N and a simple algebra S such that N/N^2 is a faithful S-module and N^j/N^{j+1} are irreducible S-modules for $j \geq 1$. In that case, if $N^t \neq 0$ and $N^{t+1} = 0$ (a such t is called the nilindex of N), the ideals of L are the $(t+1)$-element chain $0 \subset N^t \subset \cdots \subset N^i \subset \cdots \subset N \subset L$.*

Theorem 2.2. *(Snobl, 2010) Let L be Lie algebra with product $[x, y]$, nilpotent radical N of $(t+1)$-nilindex and nontrivial Levi decomposition $L = S \oplus N$ for some semisimple Lie algebra S. Then, there exists a decomposition of N into a direct sum of S-modules*

$$N = m_1 \oplus m_2 \oplus \cdots \oplus m_t$$

where $N^j = m_2 \oplus N^{j+1}$, $m_j \subseteq [m_{j-1}, m_j]$ such that m_1 is a faithful S-module and for $2 \leq j \leq t$, m_j decomposes into a sum of some subset of irreducible components of the tensor representation $m_1 \otimes m_{j-1}$.

Basic representation theory of $\mathfrak{sl}_2(k)$: Let $k[x,y]$ be the ring of polynomials in the variables x e y. For every $d \geq 0$, we denote as $V_d = span < x^d, x^{d-1}y, ..., xy^{d-1}, y^d >$ the set of homogeneous polynomials of degree d. Then, V_d is a vector spaces, $V_0 = k \cdot 1$ is of dimension 1 and V_d is of dimension $d+1$ $\forall_{d\geq 1}$. The set V_d can be viewed as a $\mathfrak{sl}_2(k)$-module in a natural way once $\mathfrak{sl}_2(k)$ is identified, into the Lie algebra $\mathfrak{gl}(k[x,y])$, as the Lie subalgebra of partial derivations $span < e = x\frac{\partial}{\partial y}, f = y\frac{\partial}{\partial x}, h = x\frac{\partial}{\partial x} - y\frac{\partial}{\partial y} >$. This action turns V_d into an irreducible representation. Even more, any finite-dimensional irreducible representations of $\mathfrak{sl}_2(k)$ can be viewed in this way. (V_0 is the trivial representation.)

The *Clebsch-Gordan's formula* gives the following decomposition of the tensor product of two $\mathfrak{sl}_2(k)$-irreducible representations:

$$V_n \otimes_k V_m \cong V_{n+m} \oplus V_{n+m-2} \oplus V_{n+m-4} \ldots \oplus V_{n-m}, \quad n \geq m. \tag{3}$$

Now, for $0 \leq k \leq m$, let's consider the bilinear *transvection map* $(\cdot,\cdot)_k : V_n \times V_m \to V_{n+m-2k}$ introduced in [4] as:

$$(f,g)_k = \frac{(m-k)!}{m!}\frac{(n-k)!}{n!}\sum_{i=0}^{k}(-1)^i \binom{k}{i} \frac{\partial^k f}{\partial x^{k-i}\partial y^i}\frac{\partial^k g}{\partial x^i \partial y^{k-i}} \tag{4}$$

The map $(f,g) \mapsto (f,g)_k$ provides a $\mathfrak{sl}_2(k)$-invariant product as it is explained in [4]. From Schur's lemma and Clebs-Gordan's formula, it is easy to prove the following result:

Lemma 2.3. *Any bilinear $\mathfrak{sl}_2(k)$-invariant products $P_{n,m,p} : V_n \otimes V_m \to V_p$, satisfies:*

- *$P_{n,m,p} = 0$ in case $p \neq n+m-2k$.*
- *$P_{n,m,p} = \alpha \cdot (f,g)_k$ for some $\alpha \in k$ for $p = n+m-2k$ and $0 \leq k \leq \min\{n,m\}$.*

Moreover, $P_{n,m,p}(x,y) = (-1)^k P_{n,m,p}(y,x)$, i.e. the product $P_{n,m,p}$ is either symmetric or skewsymmetric.

The previous lemma plays an important role in the construction of Lie algebras in which their Levi factor is, up to isomorphism, $\mathfrak{sl}_2(k)$.

3. Computational Algorithms

The based field is $k = \mathbb{C}$ along this section.

Algorithm 3.1. 3-step algorithm for the construction of Lie algebras with 5 ideals and Levi factor $\mathfrak{sl}_2$. We use transvector maps given in (4) for defining the products and Jacobi Identity for checking a Lie algebra is being generated.

Input: Three integers will be provided: n, m, p. Each of these integers represents respectively the $\mathfrak{sl}_2$-modules V_n, V_m, V_p. The integer m should verify that $m = 2n - 2k$ where k is an odd number $0 < k \leq n$. Also p should verify that $p = 3n - 2k - 2r$ with $0 \leq r \leq \min\{n, 2(n-k)\}$. In that way the input observe Clebsch-Gordan's formula (3) that controls the selection of the irreducible $\mathfrak{sl}_2$-modules V_m and V_p.

Output: True in the case that the Jacobi identity is met for all the 3-tuples of elements considered. False in the opposite case. At the same time it will be written in a text file the product (to keep it defined) of each of the following pairs of elements: two elements of V_n, two elements of V_m or one element of V_n and the other one of V_m.

1. Given the integers, the program generates the monomials bases. For example, if the integer is n, the base will be $\{v_0 := X^n, v_1 := X^{n-1}Y, \ldots, v_n := Y^n\}$.

2. Check if the following equations are verified:

$J(a,b,c) = [[a,b],c]+[[c,a],b]+[[a,b],c] = 0$ with $a,b,c \in \{v_0,\dots,v_n,w_0,\dots,w_m,u_0,\dots,u_p\}$.

3. $[a,b]$ will be defined through the transvection map $(,)_s : V_n \otimes V_\alpha \to V_\beta$ for the tuples $(s,\alpha,\beta) = (k,n,2(n-k)),(r,2(n-k),3n-2(k+r))$. The definition of these products will be saved in the text file.

4. In order to check the whole set of identities in 2, consider $i = 0$.

A) Set $a = v_i$. Check the identity for all pairs (b,c) of elements as in item 2.

B) If is true $i := i+1$ and goes to a). Else RETURN FALSE.

C) If is true for $0 < i < n+1$. Set $i = 0$, $a = w_i$, A) and B) with $0 < i < m+1$. Else RETURN FALSE.

D) If is true for $0 < i < m+1$. Set $i = 0$, $a = u_i$. A) and B) with $0 < i < p+1$. Else RETURN FALSE.

5. If the program is in this point is because the Jacobi identity is verified by all the 3-tuples considered. RETURN TRUE.

Algorithm 3.2. 3-step algorithm for the construction of Lie algebras with 5-ideals and Levi factor $\mathfrak{sl}_3(k)$ or $\mathfrak{sp}_4(k)$. Following Theorems 2.1 and 2.2, we start with a fundamental module $m_1 = V(\lambda_i)$, $i = 1,2$ and using LiE software [7], a computer algebra package for Lie computations, we check the tensor product decompositions and we select the suitable irreducible modules. Then we check Jacobi identity in a similar vein to that included in Algorithm 3.1 for determining the structure constants that provides a Lie algebra structure.

Remark 3.3. Let L be an indecomposable and nonsolvable Lie algebra satisfying $n_I(L) = 3$ or 4. Then, L can be imbedded into a *split extension* $\mathcal{L}(S,N) = S \oplus N$ where S is a simple Lie algebra and N is either an S-irreducible and faithful module in the case $n_I(L) = 3$ or the free nilpotent 2-step Lie algebra $N = V \oplus \Lambda^2 V$ where V is an S-irreducible and faithful module.

References

[1] P. Benito: Lie algebras with a small number of ideals, Linear Algebra and its Applications, 177: 233-249 (1992).

[2] P. Benito: On constructions of non-solvable Lie algebras whose ideals are in chain, Non-associative algebras and its Applications, Math. Appl. 303 (1994).

[3] P. Benito, D. de-la-Concepción: Sage computations of $\mathfrak{sl}_2(k)$-Levi extensions, Tbilisi Math. Journal, 5 (2) (2012), 3-17.

[4] J. Dixmier: Certaines algèbres non associatives simples définies par la transvection des formes binaires. J. Reine Angew Math. 346 (1984), 110-128.

[5] J.E. Humpreys: Introduction to Lie Algebras and Representation Theory, Springer Verlag NY, Inc. New York, 1972.

[6] N. Jacobson: Lie algebras, Dover Publications, Inc. New York, 1962.

[7] LiE program web page: wwwmathlabo.univ-poitiers.fr/∼maavl/LiE/

[8] L. Snobl: On the structure of maximal solvable extensions and on Levi extensions of nilpotent Lie algebras, J. Phys. A: Math Theo. 43 (2010), 505202 (17 pp)

University of La Rioja
E-mail address: pilar.benito@unirioja.es, arnedoivan5@hotmail.com

ALGEBRAIC INVARIANTS OF PROJECTIVE MONOMIAL CURVES ASSOCIATED TO GENERALIZED ARITHMETIC SEQUENCES

I. BERMEJO, E. GARCÍA-LLORENTE, AND I. GARCÍA-MARCO

ABSTRACT. Let K be an infinite field and let $m_1 < \cdots < m_n$ be a generalized arithmetic sequence of positive integers, i.e., there exist $h, d, m_1 \in \mathbb{Z}^+$ such that $m_i = hm_1 + (i-1)d$ for all $i \in \{2, \ldots, n\}$. We consider the projective monomial curve $\mathcal{C} \subset \mathbb{P}^n_K$ parametrically defined by

$$x_1 = s^{m_1}t^{m_n-m_1}, \ldots, x_{n-1} = s^{m_{n-1}}t^{m_n-m_{n-1}}, x_n = s^{m_n}, x_{n+1} = t^{m_n}.$$

In this work, we characterize the Cohen-Macaulay and Koszul properties of the homogeneous coordinate ring $K[\mathcal{C}]$ of $\mathcal{C}$. Whenever $K[\mathcal{C}]$ is Cohen-Macaulay we also obtain a formula for its Cohen-Macaulay type. Moreover, when h divides d, we obtain a minimal Gröbner basis $\mathcal{G}$ of the vanishing ideal of $\mathcal{C}$ with respect to the degree reverse lexicographic order. From $\mathcal{G}$ we derive formulas for the Castelnuovo-Mumford regularity, the Hilbert series and the Hilbert function of $K[\mathcal{C}]$ in terms of the sequence.

INTRODUCTION

Let $R := K[x_1, \ldots, x_{n+1}]$ be the polynomial ring in $n+1$ variables over an infinite field K, where $n \geq 2$. Let $m_1 < \cdots < m_n$ be positive integers and consider the projective monomial curve $\mathcal{C} \subset \mathbb{P}^n_K$ parametrically defined by

$$x_1 = s^{m_1}t^{m_n-m_1}, \ldots, x_{n-1} = s^{m_{n-1}}t^{m_n-m_{n-1}}, x_n = s^{m_n}, x_{n+1} = t^{m_n}.$$

Consider the K-algebra homomorphism $\varphi : R \to K[s,t]$ induced by $\varphi(x_i) = s^{m_i}t^{m_n-m_i}$ for $i \in \{1, \ldots, n-1\}$, $\varphi(x_n) = s^{m_n}$ and $\varphi(x_{n+1}) = t^{m_n}$. Since K is infinite, the prime ideal $ker(\varphi) \subset R$ is the vanishing ideal $I(\mathcal{C})$ of $\mathcal{C}$. Thus, $R/I(\mathcal{C})$ is the homogenous coordinate ring $K[\mathcal{C}]$ of $\mathcal{C}$. Moreover, $I(\mathcal{C})$ is a homogeneous binomial ideal.

The *Castelnuovo-Mumford regularity* , or *regularity*, of $K[\mathcal{C}]$ is defined as follows: if

$$0 \to \bigoplus_{j=1}^{\beta_p} R(-e_{pj}) \to \cdots \to \bigoplus_{j=1}^{\beta_1} R(-e_{1j}) \to R \to K[\mathcal{C}] \to 0$$

is a minimal graded free resolution of $K[\mathcal{C}]$, then $\mathrm{reg}(K[\mathcal{C}]) := \max\{e_{ij} - i;\ 1 \leq i \leq p,\ 1 \leq j \leq \beta_i\}$. Whenever $K[\mathcal{C}]$ is Cohen-Macaulay, the integer β_p is the *Cohen-Macaulay type* of $K[\mathcal{C}]$; in addition, if $\beta_p = 1$, the K-algebra $K[\mathcal{C}]$ is *Gorenstein*. The corresponding ideal, $I(\mathcal{C})$, is a *complete intersection* if it can be generated by a set of $n-1$ polynomials. $K[\mathcal{C}]$ is said to be a *Koszul algebra* if its residue class field K has a linear free resolution as $K[\mathcal{C}]$-module.

For all $s \in \mathbb{N}$, we denote by R_s the K-vector space spanned by all homogeneous polynomials of degree s. The *Hilbert function* and *Hilbert series* of $K[\mathcal{C}]$ are $HF_{K[\mathcal{C}]}(s) := \dim_K(R_s/I(\mathcal{C}) \cap R_s)$ for all $s \in \mathbb{N}$, and $\mathcal{H}_{K[\mathcal{C}]}(t) := \sum_{s\in\mathbb{N}} HF_{K[\mathcal{C}]}(s)\, t^s$, respectively.

In this work we study the already mentioned invariants and properties when $m_1 < \cdots < m_n$ is a generalized arithmetic sequence of relatively prime integers, i.e., there exist $h, d, m_1 \in \mathbb{Z}^+$ such that $m_i = hm_1 + (i-1)d$ for each $i \in \{2, \ldots, n\}$ and $\gcd\{m_1, d\} = 1$. In this context, we prove that $K[\mathcal{C}]$ is Cohen-Macaulay if and only if $m_1 < \cdots < m_n$ is an arithmetic sequence. Additionally, in the Cohen-Macaulay case, we provide a formula for its Cohen-Macaulay type yielding a characterization of the Gorenstein property. Moreover, we prove that $K[\mathcal{C}]$ is Koszul if and only if $m_1, \ldots, m_n$ are consecutive numbers and $n > m_1$.

In addition, we study in detail when $m_1 < \cdots < m_n$ is a generalized arithmetic sequence and h divides d. In this setting we provide a formula for $\operatorname{reg}(K[\mathcal{C}])$. To this end, we follow the ideas of [4]. In that paper the authors prove that for any projective monomial curve $\mathcal{C}$, the regularity of $K[\mathcal{C}]$ is equal to the regularity of $R/\operatorname{in}(I(\mathcal{C}))$, where $\operatorname{in}(I(\mathcal{C}))$ denotes the initial ideal of $I(\mathcal{C})$ with respect to the degree reverse lexicographic order with $x_1 > \cdots > x_{n+1}$. Moreover, they prove that the regularity of $R/\operatorname{in}(I(\mathcal{C}))$ is the maximum of the regularities of the irreducible components of $\operatorname{in}(I(\mathcal{C}))$. Since the regularity of an irreducible monomial ideal is easy to compute, our strategy consists of obtaining a minimal Gröbner basis $\mathcal{G}$ of $I(\mathcal{C})$ with respect to the degree reverse lexicographic order with $x_1 > \cdots > x_{n+1}$. From $\mathcal{G}$, we get the irreducible components of $\operatorname{in}(I(\mathcal{C}))$, which allow us to deduce the value of $\operatorname{reg}(K[\mathcal{C}])$ in terms of the sequence $m_1 < \cdots < m_n$. For obtaining this result we separate two cases: the Cohen-Macaulay case, i.e., when $m_1 < \cdots < m_n$ is an arithmetic sequence (Section 1); and the non Cohen-Macaulay one, i.e., when $h \geq 2$ and h divides d (Section 2). We also exploit the knowledge of $\mathcal{G}$ to describe both the Hilbert series and the Hilbert function of $K[\mathcal{C}]$.

The results of this work are included in [1].

1. Arithmetic sequences

This section concerns the study of $K[\mathcal{C}]$ when $m_1 < \cdots < m_n$ an arithmetic sequence of relatively prime integers and $n \geq 2$. The key point of this section is to prove Theorem 1.1, where we describe a minimal Gröbner basis $\mathcal{G}$ of $I(\mathcal{C})$ with respect to the degree reverse lexicographic order with $x_1 > \cdots > x_{n+1}$.

We begin by associating to the sequence $m_1 < \cdots < m_n$ a pair $(\alpha, k) \in \mathbb{N}^2$ in the following way. Take $q \in \mathbb{N}$ and $r \in \{1, \ldots, n-1\}$ such that $m_1 = q(n-1) + r$ and set $\alpha := q + d$ and $k := n - r$. With this notation, we can state the main result of this section.

Theorem 1.1.

$$\mathcal{G} = \{x_i x_j - x_{i-1} x_{j+1} \mid 2 \leq i \leq j \leq n-1\} \cup \{x_1^{\alpha} x_i - x_{n+1-i} x_n^q x_{n+1}^d \mid 1 \leq i \leq k\}$$

is a minimal Gröbner basis of $I(\mathcal{C})$ with respect to the degree reverse lexicographic order. Moreover $\mathcal{G}$ is a minimal set of generators of $I(\mathcal{C})$.

We observe that the variables x_n and x_{n+1} do not appear in the minimal set of generators of $\operatorname{in}(I(\mathcal{C}))$. Hence, by [3, Proposition 2.1], we deduce that $K[\mathcal{C}]$ is Cohen-Macaulay. The following result provides the Cohen-Macaulay type and characterizes the Gorenstein property for $K[\mathcal{C}]$.

Corollary 1.2. *$K[\mathcal{C}]$ is Cohen-Macaulay. Moreover, if we take $\tau \in \{1, \ldots, n-1\}$ such that $m_1 - 1 \equiv \tau \pmod{n-1}$; then, the Cohen-Macaulay type of $K[\mathcal{C}]$ is τ. In particular, $K[\mathcal{C}]$ is Gorenstein $\iff m_1 \equiv 2 \pmod{n-1}$.*

As we claimed in the Introduction, $\operatorname{reg}(K[\mathcal{C}]) = \operatorname{reg}(R/\operatorname{in}(I(\mathcal{C})))$ and the regularity of $R/\operatorname{in}(I(\mathcal{C}))$ is the maximum of the regularities of the irreducible components of $\operatorname{in}(I(\mathcal{C}))$. Theorem 1.1 provides a set of generators of $\operatorname{in}(I(\mathcal{C}))$, hence, by computing the irreducible components of $\operatorname{in}(I(\mathcal{C}))$ and their regularities, we deduce the following.

Theorem 1.3. *Let $m_1 < \cdots < m_n$ be an arithmetic sequence with $\gcd\{m_1, d\} = 1$. Then,*

$$\operatorname{reg}(K[\mathcal{C}]) = \left\lceil \frac{m_n - 1}{n-1} \right\rceil.$$

Now we exploit Theorem 1.1 to provide an explicit description of the Hilbert series and the Hilbert function of $K[\mathcal{C}]$.

Theorem 1.4. *The Hilbert series and Hilbert function of $K[\mathcal{C}]$ are*

$$\mathcal{H}_{K[\mathcal{C}]}(t) = (1 + (n-1)(t + \cdots + t^{\alpha}) + (n-1-k)t^{\alpha+1})/(1-t)^2,$$

$$HF_{K[\mathcal{C}]}(s) = \begin{cases} \binom{s+2}{2} + (n-2)\binom{s+1}{2} & \text{if } 0 \leq s < \alpha, \\ m_n\, s + \alpha(2-n+k) - \binom{\alpha+1}{2} - (n-2)\binom{\alpha}{2} + 1 & \text{if } s \geq \alpha. \end{cases}$$

Corollary 1.2 and Theorem 1.4 were already obtained in [5] by other methods in the particular setting where $\{m_1, \ldots, m_n\}$ form a minimal set of generators of the semigroup they generate.

2. Generalized arithmetic sequences

This section concerns the study of $K[\mathcal{C}]$ when $m_1 < \cdots < m_n$ is a generalized arithmetic sequence and $n \geq 3$. The first result of this section is a characterization of the Cohen-Macaulay property in this setting.

Corollary 2.1. *$K[\mathcal{C}]$ is Cohen-Macaulay if and only if $m_1 < \cdots < m_n$ is an arithmetic sequence.*

Moreover, from Corollary 2.1 and Theorem 1.1, we recover [2, Theorem 6.1].

Corollary 2.2. *Let $m_1 < \cdots < m_n$ be a generalized arithmetic sequence of relatively prime integers. Then, $I(\mathcal{C})$ is a complete intersection if and only if $n = 3$, $h = 1$, and m_1 is even.*

We also characterize the Koszul property in this setting.

Theorem 2.3. *Let $m_1 < \cdots < m_n$ be a generalized arithmetic sequence of relatively prime integers. Then, $K[\mathcal{C}]$ is a Koszul algebra if and only if $m_1 < \cdots < m_n$ are consecutive numbers and $n > m_1$.*

In the rest of this section we assume that $m_1 < \cdots < m_n$ is a generalized arithmetic sequence of relatively prime integers with $n \geq 3$, $h > 1$ and h divides d. It turns out that under this hypotheses $I(\mathcal{C})$ is closely related to $I(\mathcal{C}')$, where $\mathcal{C}'$ is the projective monomial curve associated to the arithmetic sequence $m_2 < \cdots < m_n$. For a sake of convenience we assume that $I(\mathcal{C}') \subset K[x_2, \ldots, x_{n+1}]$. The following result relates the initial ideals of $I(\mathcal{C})$ and $I(\mathcal{C}')$ with respect to the degree reverse lexicographic order with $x_1 > \cdots > x_{n+1}$.

Proposition 2.4. $\operatorname{in}(I(\mathcal{C}')) = \operatorname{in}(I(\mathcal{C})) \cap K[x_2, \ldots, x_{n+1}]$.

In order to describe a minimal Gröbner basis of $I(\mathcal{C})$ we introduce the following notation. Take $p \in \mathbb{N}$ and $s \in \{1, \dots, n-1\}$ such that $m_1 = p(n-1)+s$ and set $\delta := ph+d+h$ and $\delta' := \delta/h$. For all $j \in \{0 \dots, \delta'-1\}$, we set

- $\beta_{\delta'-j} := j + \lfloor (j+s-2)/(n-2) \rfloor$,
- $\sigma_{\delta'-j}$ is the only integer in $\{3, \dots, n\}$ such that $\sigma_{\delta'-j} \equiv s+1+j \pmod{n-2}$, and
- $\lambda_{\delta'-j} := p + \lfloor (j+s-2)/(n-2) \rfloor$.

Theorem 2.5. *Let $\mathcal{G}_1 \subset K[x_2, \dots, x_n, x_{n+1}]$ be a minimal Gröbner basis with respect to the degree reverse lexicographic order of $I(\mathcal{C}')$. Then,*

$$\begin{array}{rcl} \mathcal{G} & := & \mathcal{G}_1 \cup \{x_1^h x_i - x_2 x_{i-1} x_{n+1}^{h-1} \mid 3 \le i \le n\} \\ & & \cup \{x_1^{jh} x_2^{\beta_j} - x_{\sigma_j} x_n^{\lambda_j} x_{n+1}^{j(h-1)+(d/h)} \mid 1 \le j \le \delta/h\}, \end{array}$$

is a minimal Gröbner basis of $I(\mathcal{C})$ with respect to the degree reverse lexicographic order. Moreover, $\mathcal{G}$ is a minimal set of generators of $I(\mathcal{C})$.

Following the same strategy as in the previous section, we obtain the value of the regularity.

Theorem 2.6. *Let $m_1 < \cdots < m_n$ be a generalized arithmetic sequence with $\gcd\{m_1, d\} = 1$, $h > 1$ and h divides d. Then,*

$$\operatorname{reg}(K[\mathcal{C}]) = \begin{cases} \delta - 1 & \text{if } n-1 \text{ does not divide } m_1, \\ \delta & \text{if } n-1 \text{ divides } m_1. \end{cases}$$

Finally, we provide formulas for the Hilbert series and the Hilbert function of $K[\mathcal{C}]$ in this setting.

Theorem 2.7. *The Hilbert series and Hilbert function of $K[\mathcal{C}]$ are*

$$\mathcal{H}_{K[\mathcal{C}]}(t) = (1 + \cdots + t^{h-1})\mathcal{H}_{K[\mathcal{C}']}(t) + \frac{\sum_{j=h}^{\delta-1} t^j - \left(\sum_{j=0}^{h-1} t^j\right)\left(\sum_{i=1}^{(\delta/h)-1} t^{ih+\beta_i}\right)}{(1-t)^2},$$

$$HF_{K[\mathcal{C}]}(s) = \textstyle\sum_{i=0}^{h-1} HF_{K[\mathcal{C}']}(s-i) + (n-3)\Delta_s + \Delta_{s+1},$$

where $\Delta_s := \#\{(a,b) \in \mathbb{N}^2 \mid a+b < s \text{ and } b < \beta_j, \text{ with } j := \lfloor a/h \rfloor \ge 1\}$.

References

[1] I. Bermejo, E. García-Llorente, I. García-Marco, Algebraic invariants of projective monomial curves associated to generalized arithmetic sequences. arXiv:1512.00617 [math.AC].

[2] I. Bermejo, I. García-Marco, Complete intersections in certain affine and projective monomial curves, Bull Braz Math Soc, New Series **45(4)**, 2014, 1-26.

[3] I. Bermejo, P. Gimenez, Computing the Castelnuovo-Mumford regularity of some subeschemes of $\mathbb{P}^n_K$ using quotients of monomial ideals. *J. Pure Appl. Algebra* **164** (2001), 23–33.

[4] I. Bermejo, P. Gimenez, Saturation and Castelnuovo-Mumford regularity. *J. Algebra* **303** (2006), 592–617.

[5] S. Molinelli, G. Tamone, On the Hilbert function of certain rings of monomial curves. *J. Pure Appl. Algebra* **101** (1995), no. 2, 191–206

Facultad de Ciencias, Sección de Matemáticas, Universidad de La Laguna, La Laguna, Spain
E-mail address: ibermejo@ull.es, evgarcia@ull.es

Laboratoire de l'Informatique du Parallélisme (LIP), Ecole Normale Supérieure de Lyon, France
E-mail address: ignacio.garcia-marco@ens-lyon.fr, iggarcia@ull.es

NOETHER RESOLUTIONS IN DIMENSION 2

I. BERMEJO, E. GARCÍA-LLORENTE, I. GARCÍA-MARCO, AND M. MORALES

Abstract. Let $R := K[x_1, \dots, x_n]$ be a polynomial ring over an infinite field K, and let $I \subset R$ be a homogeneous ideal with respect to a weight vector $\omega = (\omega_1, \dots, \omega_n) \in (\mathbb{Z}^+)^n$ such that $\dim(R/I) = d$. In this work we study the minimal graded free resolution of R/I as A-module, which we call the Noether resolution of R/I, whenever $A := K[x_{n-d+1}, \dots, x_n]$ is a Noether normalization of R/I. When $d = 2$ and I is saturated, we give an algorithm for obtaining this resolution that involves the computation of a minimal Gröbner basis of I with respect to the weighted degree reverse lexicographic order. In the particular case when R/I is a 2-dimensional semigroup ring, we also describe the multigraded version of this resolution in terms of the underlying semigroup. Whenever we have the Noether resolution of R/I or its multigraded version, we obtain formulas for the corresponding Hilbert series of R/I, and when I is homogeneous, we obtain a formula for the Castelnuovo-Mumford regularity of R/I.

Introduction

Let $R := K[x_1, \dots, x_n]$ be a polynomial ring over an infinite field K, and let $I \subset R$ be a weighted homogeneous ideal with respect to the vector $\omega = (\omega_1, \dots, \omega_n) \in (\mathbb{Z}^+)^n$, i.e., I is homogeneous for the grading $\deg_\omega(x_i) = \omega_i$. We denote by d the Krull dimension of R/I and we assume that $d \geq 1$. Suppose $A := K[x_{n-d+1}, \dots, x_n]$ is a Noether normalization of R/I, i.e., $A \hookrightarrow R/I$ is an integral ring extension. Under this assumption R/I is a finitely generated A-module, so to study the minimal graded free resolution of R/I as A-module is an interesting problem. Set

$$\mathcal{F} : 0 \longrightarrow \oplus_{v \in \mathcal{B}_p} A(-s_{p,v}) \xrightarrow{\psi_p} \cdots \xrightarrow{\psi_1} \oplus_{v \in \mathcal{B}_0} A(-s_{0,v}) \xrightarrow{\psi_0} R/I \longrightarrow 0 \tag{1}$$

this resolution, where for all $i \in \{0, \dots, p\}$ $\mathcal{B}_i$ denotes some finite set, and $s_{i,v} \in \mathbb{N}$. This work concerns the study of this resolution of R/I, which will be called the *Noether resolution of* R/I. More precisely, we aim at determining the sets $\mathcal{B}_i$, the shifts $s_{i,v}$ and the morphisms ψ_i. When $d = 2$ and I is saturated we get the whole Noether resolution of R/I and, as a consequence, we obtain the Hilbert series of R/I. Moreover, whenever I is a homogeneous ideal, i.e., homogeneous for the weight vector $\omega = (1, \dots, 1)$, we derive a formula for the Castelnuovo-Mumford regularity of R/I.

For the second part of this work, we consider that R/I is a simplicial semigroup ring, i.e., I is a toric ideal and $A = K[x_{n-d+1}, \dots, x_n]$ is a Noether normalization of R/I. We recall that I is a toric ideal if $I = I_{\mathcal{A}}$ with $\mathcal{A} = \{a_1, \dots, a_n\} \subset \mathbb{N}^d$ and $a_i = (a_{i1}, \dots, a_{id}) \in \mathbb{N}^d$; where $I_{\mathcal{A}}$ denotes the kernel of the homomorphism of K-algebras $\varphi : R \to K[t_1, \dots, t_d]$; $x_i \mapsto t^{a_i} = t_1^{a_{i1}} \cdots t_d^{a_{id}}$ for all $i \in \{1, \dots, n\}$. If we denote by $\mathcal{S} \subset \mathbb{N}^d$ the semigroup generated by $a_1, \dots, a_n$, then the image of φ is $K[\mathcal{S}] := K[t^s \mid s \in \mathcal{S}] \simeq R/I_{\mathcal{A}}$. By [4, Corollary 4.3], $I_{\mathcal{A}}$ is multigraded with respect to the grading induced by $\deg_{\mathcal{S}}(x_i) = a_i$ for all $i \in \{1, \dots, n\}$. In

this setting we may consider the minimal multigraded free resolution of $K[\mathcal{S}]$ as A-module, which will be called the *multigraded Noether resolution of* $K[\mathcal{S}]$:

$$0 \longrightarrow \oplus_{s\in\mathcal{S}_p} A\cdot s \xrightarrow{\psi_p} \cdots \xrightarrow{\psi_1} \oplus_{s\in\mathcal{S}_0} A\cdot s \xrightarrow{\psi_0} K[\mathcal{S}] \longrightarrow 0, \tag{2}$$

where $\mathcal{S}_i \subset \mathcal{S}$ are finite sets for all $i \in \{0,\dots,p\}$ and $A\cdot s$ denotes the shifting of A by $s \in \mathcal{S}$.

The problem we tackle here is to describe combinatorially the multigraded Noether resolution of $K[\mathcal{S}]$ in terms of the semigroup $\mathcal{S}$. In this work we completely determine the Noether resolution of $K[\mathcal{S}]$ by means of $\mathcal{S}$ when $d = 2$. As a consequence, we obtain the multigraded Hilbert series of $K[\mathcal{S}]$. Moreover we get a formula for the Castelnuovo-Mumford regularity of $K[\mathcal{S}]$ in terms of $\mathcal{S}$ when I is a homogeneous ideal. The results regarding $K[\mathcal{S}]$ are obtained as a consequence of those we get in the first part of the work.

All the results of this work are included in [1].

1. Noether resolution. General case

Let $I \subset R$ be a ω-homogeneous ideal such that $A := K[x_{n-d+1},\dots,x_n]$ is a Noether normalization of R/I. We consider the minimal graded free resolution of R/I as A-module:

$$\mathcal{F} : 0 \longrightarrow \oplus_{v\in\mathcal{B}_p} A(-s_{p,v}) \xrightarrow{\psi_p} \cdots \xrightarrow{\psi_1} \oplus_{v\in\mathcal{B}_0} A(-s_{0,v}) \xrightarrow{\psi_0} R/I \longrightarrow 0, \tag{3}$$

where for all $i \in \{0,\dots,p\}$ $\mathcal{B}_i$ is a finite set, and $s_{i,v}$ are nonnegative integers.

We start by describing the first step of the Noether resolution of R/I. Consider the weighted degree reverse lexicographic order $>_\omega$. We recall that $x^\alpha >_\omega x^\beta$ if and only if

- $\deg_\omega(x^\alpha) > \deg_\omega(x^\beta)$, or
- $\deg_\omega(x^\alpha) = \deg_\omega(x^\beta)$ and the last nonzero entry of $\alpha-\beta \in \mathbb{Z}^n$ is negative.

For every ideal $J \subset R$, we denote by $\mathrm{in}(J)$ its initial ideal with respect to $>_\omega$.

Proposition 1.1. *Let $\mathcal{B}_0$ be the set of monomials not belonging to $\mathrm{in}(I+(x_{n-d+1},\dots,x_n))$. Then, $\{x^\alpha + I \mid x^\alpha \in \mathcal{B}_0\}$ is a minimal set of generators of R/I as A-module and the shifts of the first step of the Noether resolution (3) are given by $\deg_\omega(x^\alpha)$ with $x^\alpha \in \mathcal{B}_0$.*

By Auslander-Buchsbaum formula, the depth of R/I as A-module equals $d-p$, where p is given in (3). Hence, R/I is Cohen-Macaulay if and only if $p = 0$ or, equivalently, if R/I is a free A-module. This observation together with Proposition 1.1 lead to the following result. This result generalizes [3, Proposition 2.1], which applies for I a homogeneous ideal.

Proposition 1.2. *R/I is Cohen-Macaulay if and only if $x_{n-d+1},\dots,x_n$ do not divide any minimal generator of $\mathrm{in}(I)$.*

The rest of this section concerns I a saturated ideal such that R/I is 2-dimensional and it is not Cohen-Macaulay. We assume that $A = K[x_{n-1},x_n]$ is a Noether normalization of R/I. Since K is an infinite field, I is a saturated ideal and A is a Noether normalization of R/I, one has that $x_n + \tau x_{n-1}$ is a nonzero divisor on R/I for all $\tau \in K$ but a finite set. Thus, by performing a mild change of coordinates if necessary, we may assume that x_n is a nonzero divisor on R/I. Our aim is to describe the whole Noether resolution of R/I in this setting. For this purpose we consider $\chi : R \longrightarrow R$ the evaluation morphism induced by $x_i \mapsto x_i$ for $i \in \{1,\dots,n-2\}$, $x_i \mapsto 1$ for $i \in \{n-1,n\}$.

Proposition 1.3. *Let J be the ideal $\chi(\mathrm{in}(I)) \cdot R$. Then, $\mathcal{B}_1 = \mathcal{B}_0 \cap J$ and the shifts of the second step of the Noether resolution are given by $\deg_\omega(ux_{n-1}^{\delta_u})$, where $u \in \mathcal{B}_1$ and $\delta_u := \min\{\delta \mid ux_{n-1}^{\delta} \in \mathrm{in}(I)\}$.*

From propositions 1.1 and 1.3 and their proofs, we obtain the Noether resolution $\mathcal{F}$ of R/I by means of a Gröbner basis of I with respect to $>_\omega$. We also observe that for obtaining the shifts of the resolution it suffices to know a set of generators of $\mathrm{in}(I)$.

Theorem 1.4. *Let $\mathcal{G}$ be a Gröbner basis of I with respect to $>_\omega$. If $\delta_u := \min\{\delta \mid ux_{n-1}^{\delta} \in \mathrm{in}(I)\}$ for all $u \in \mathcal{B}_1$, then*

$$\mathcal{F} : 0 \longrightarrow \oplus_{u\in\mathcal{B}_1} A(-\deg_\omega(u) - \delta_u\omega_{n-1}) \xrightarrow{\psi_1} \oplus_{v\in\mathcal{B}_0} A(-\deg_\omega(v)) \xrightarrow{\psi_0} R/I \longrightarrow 0,$$

is the Noether resolution of R/I, where

$$\begin{array}{rccc} \psi_0 : & \oplus_{v\in\mathcal{B}_0} A(-\deg_\omega(v)) & \to & R/I, \\ & e_v & \mapsto & v + I \end{array}$$

and

$$\begin{array}{rccc} \psi_1 : & \oplus_{u\in\mathcal{B}_1} A(-\deg_\omega(u) - \delta_u\omega_{n-1}) & \longrightarrow & \oplus_{v\in\mathcal{B}_0} A(-\deg_\omega(v)), \\ & e_u & \mapsto & x_{n-1}^{\delta_u} e_u - \sum_{v\in\mathcal{B}_0} f_{u_v} e_v \end{array}$$

where $\sum_{v\in\mathcal{B}_0} f_{u_v} v$ with $f_{u_v} \in A$ is the remainder of the division of $ux_{n-1}^{\delta_u}$ by $\mathcal{G}$.

As a consequence of Theorem 1.4, we obtain the Hilbert series of R/I.

Corollary 1.5.

$$HS_{R/I}(t) = \frac{\sum_{v\in\mathcal{B}_0} t^{\deg_\omega(v)} - \sum_{u\in\mathcal{B}_1} t^{\deg_\omega(u)+\delta_u w_{n-1}}}{(1-t^{\omega_{n-1}})(1-t^{\omega_n})}$$

In the particular case where I is standard graded homogeneous, i.e., $\omega = (1, \ldots, 1)$, we obtain a formula for the Castelnuovo-Mumford regularity of R/I in terms of $\mathcal{B}_0$ and $\mathcal{B}_1$. This formula is equivalent to the one provided in [2, Theorem 2.7] when x_n is a nonzero divisor of R/I.

Corollary 1.6.

$$\mathrm{reg}(R/I) = \max\{\deg(v), \deg(u) + \delta_u - 1 \mid v \in \mathcal{B}_0, u \in \mathcal{B}_1\}$$

2. Noether resolution. Simplicial semigroup rings

Let R/I be a simplicial semigroup ring. This section concerns the study of the multigraded Noether resolution of $K[\mathcal{S}]$

$$\mathcal{F} : 0 \longrightarrow \oplus_{s\in\mathcal{S}_p} A \cdot s \xrightarrow{\psi_p} \cdots \xrightarrow{\psi_1} \oplus_{s\in\mathcal{S}_0} A \cdot s \xrightarrow{\psi_0} K[\mathcal{S}] \longrightarrow 0,$$

where $\mathcal{S}_i \subset \mathcal{S}$ for all $i \in \{0, \ldots, p\}$.

For any value of $d \geq 1$, the first step of the resolution corresponds to a minimal set of generators of $K[\mathcal{S}]$ as A-module and is given by the following well known result.

Proposition 2.1. *$\mathcal{S}_0 = \{s \in \mathcal{S} \mid s - a_i \notin \mathcal{S}$ for all $i \in \{n-d+1, \ldots, n\}\}$. Moreover $\psi_0 : \oplus_{s\in\mathcal{S}_0} A \cdot s \longrightarrow K[\mathcal{S}]$ is the homomorphism of A-modules induced by $e_s \mapsto t^s$, where $\{e_s \mid s \in \mathcal{S}_0\}$ is the canonical basis of $\oplus_{s\in\mathcal{S}_0} A \cdot s$.*

Proposition 2.1 gives the whole Noether resolution of $K[\mathcal{S}]$ when $K[\mathcal{S}]$ is Cohen-Macaulay. Moreover, the following result characterizes when $K[\mathcal{S}]$ is Cohen-Macaulay.

Proposition 2.2. *Let $\mathcal{S}$ be a simplicial semigroup as above. Set $D := \left(\prod_{i=1}^{d} \omega_{n-d+i}\right)/[\mathbb{Z}^d : \mathbb{Z}\mathcal{S}]$, where $[\mathbb{Z}^d : \mathbb{Z}\mathcal{S}]$ denotes the index of the group generated by $\mathcal{S}$ in $\mathbb{Z}^d$. Then, $K[\mathcal{S}]$ is Cohen-Macaulay $\iff |\mathcal{S}_0| = D$.*

As we did in the previous section, from now on we restrict to 2-dimensional and non Cohen-Macaulay simplicial semigroup rings. In this setting we get the second step of the multigraded Noether resolution and, hence, we have described the whole resolution.

Theorem 2.3.

$$\mathcal{S}_1 = \{s \in \mathcal{S} \mid s - a_{n-1}, s - a_n \in \mathcal{S} \text{ and } s - a_n - a_{n-1} \notin \mathcal{S}\}.$$

As a byproduct, we obtain the multigraded Hilbert series of $K[\mathcal{S}]$.

Corollary 2.4. *The multigraded Hilbert series of $K[\mathcal{S}]$ is*

$$HS_{K[\mathcal{S}]}(t) = \frac{\sum_{s\in\mathcal{S}_0} t^s - \sum_{s\in\mathcal{S}_1} t^s}{(1-t_1^{\omega_{n-1}})(1-t_2^{\omega_n})}$$

Moreover, whenever I is homogeneous, we have the following result.

Corollary 2.5.

$$\operatorname{reg}(K[\mathcal{S}]) = \max\left(\left\{\frac{b_1+b_2}{\omega_1} \mid (b_1,b_2) \in \mathcal{S}_0\right\} \cup \left\{\frac{b_1+b_2}{\omega_1} - 1 \mid (b_1,b_2) \in \mathcal{S}_1\right\}\right).$$

References

[1] I. Bermejo, E. García-Llorente, I. García-Marco, M. Morales, Noether resolutions in dimension 2. (2016)

[2] I. Bermejo, Ph. Gimenez, On Castelnuovo-Mumford regularity of projective curves. Proc. Amer. Math. Soc. **128** (2000), no. 5, 1293–1299.

[3] I. Bermejo, Ph. Gimenez, Computing the Castelnuovo-Mumford regularity of some subschemes of $\mathbb{P}^n_K$ using quotients of monomial ideals, Effective methods in algebraic geometry (Bath, 2000). J. Pure Appl. Algebra bf 164 (2001), no. 1-2, 23–33.

[4] B. Sturmfels, *Gröbner Bases and Convex Polytopes*. University Lecture Series **8**, American Mathematical Society, Providence, RI, 1996.

[5] R. H. Villarreal, Monomial Algebras, Second Edition, Monographs and Research Notes in Mathematics. *Chapman and Hall/CRC*, 2015.

Facultad de Ciencias, Sección de Matemáticas, Universidad de La Laguna, La Laguna, 38071, Spain

E-mail address: ibermejo@ull.es, evgarcia@ull.es

LIP, ENS Lyon - CNRS - UCBL - INRIA, Université de Lyon UMR 5668, Lyon, France

E-mail address: ignacio.garcia-marco@ens-lyon.fr, iggarcia@ull.es

Université de Grenoble I, Institut Fourier, UMR 5582, B.P.74, 38402 Saint-Martin D'Heres Cedex, Grenoble and ESPE de Lyon, Université de Lyon 1, Lyon, France

E-mail address: morales@ujf-grenoble.fr

AN ALGORITHM FOR CONSTRUCTING CERTAIN DIFFERENTIAL OPERATORS IN POSITIVE CHARACTERISTIC

ALBERTO F. BOIX, ALESSANDRO DE STEFANI, AND DAVIDE VANZO

ABSTRACT. Given a non-zero polynomial f in a polynomial ring R with coefficients in a finite field of prime characteristic p, we present an algorithm to compute a differential operator δ which raises $1/f$ to its pth power. In particular, we obtain a characterization of supersingular elliptic curves in terms of the level of the associated differential operator, i.e., the least integer e such that δ is R^{p^e}-linear.

INTRODUCTION

Let $R = k[x_1, \ldots, x_d]$ be the polynomial ring over a field k, and let $\mathcal{D}_R$ be the ring of k-linear differential operators on R. For every non-zero $f \in R$, the natural action of $\mathcal{D}_R$ on R extends uniquely to an action on R_f. In characteristic 0, it has been proven by Bernstein in the polynomial ring case that R_f has finite length as a $\mathcal{D}_R$-module. The minimal m such that $R_f = \mathcal{D}_R \cdot \frac{1}{f^m}$ is related to Bernstein-Sato polynomials (cf. [4, Theorem 23.7, Definition 23.8, and Corollary 23.9]), and there are examples in which $m > 1$ (e.g., [4, Example 23.13]). Remarkably, in positive characteristic, not only R_f is finitely generated as a $\mathcal{D}_R$-module, but it is generated by $\frac{1}{f}$ (cf. [1, Theorem 3.7 and Corollary 3.8]). This is shown by proving the existence of a differential operator $\delta \in \mathcal{D}_R$ such that $\delta(1/f) = 1/f^p$, i.e., a differential operator that acts as the Frobenius homomorphism on $1/f$. The main result of this report exhibits an algorithm that, given $f \in R$, produces a differential operator $\delta \in \mathcal{D}_R$ such that $\delta(1/f) = 1/f^p$ (see Section 2). We will call such a δ a *differential operator associated with* f; moreover, this procedure has been implemented using the computer algebra system Macaulay2.

Assume that $\mathrm{char}(k) = p > 0$ and that $[k : k^p] < \infty$. For $e \geqslant 1$ let R^{p^e} be the subring of R consisting of all p^e-th powers of elements in R, which can also be viewed as the image of the e-th iteration of the Frobenius endomorphism $F : R \to R$. We set $R^{p^0} := R$. It is shown in [5, 1.4.9] that $\mathcal{D}_R$ is equal to the increasing union $\bigcup_{e \geqslant 0} \mathrm{End}_{R^{p^e}}(R)$. Therefore, given $\delta \in \mathcal{D}_R$, there exists $e \geqslant 0$ such that $\delta \in \mathrm{End}_{R^{p^e}}(R)$ but $\delta \notin \mathrm{End}_{R^{p^{e'}}}(R)$ for any $e' < e$. Given a non-zero polynomial $f \in R$, we have seen above that there exists $\delta \in \mathcal{D}_R$ that is associated with f. We say that f has level e if such δ is R^{p^e}-linear, and there is no $R^{p^{e'}}$-linear differential operator δ', with $e' < e$, that is associated with f.

2010 *Mathematics Subject Classification.* Primary 13A35; Secondary 13N10, 14B05.

Key words and phrases. Algorithm, Differential operator, Frobenius map, Prime characteristic.

The first named author is partially supported by MTM2013-40775-P and by the CASB fellowship program.

The second named author is partially supported by NSF Grant DMS-1259142.

In Section 3 we focus on Elliptic Curves $\mathcal{C} \subseteq \mathbb{P}^2_{\mathbb{F}_p}$, where $\mathbb{F}_p$ is the finite field with p elements; our main result is the following:

Theorem. Let $p \in \mathbb{Z}$ be a prime number and let $\mathcal{C} \subseteq \mathbb{P}^2_{\mathbb{F}_p}$ be an elliptic curve defined by a cubic $f(x, y, z) \in \mathbb{F}_p[x, y, z]$. Then $\mathcal{C}$ is supersingular if and only if f has level two.

The content of this report is based on [2], where the reader can find all the details; in particular, our implementation of the algorithm is described in [2, Section 8].

1. The level of a differential operator

The goal of this section is to review the definitions, notations and results that we use throughout this report. Unless otherwise specified, k will denote a perfect field of prime characteristic p. Under this assumption, it is known (see [3, IV, Théorème 16.11.2]) that the ring of k-linear differential operators over $R = k[x_1, \dots, x_d]$ can be expressed in the following way:

$$\mathcal{D}_R := R\langle D_{x_i,t} \mid i = 1, \dots, d \text{ and } t \geqslant 1\rangle, \quad \text{where } D_{x_i,t} := \frac{1}{t!}\frac{\partial^t}{\partial x_i^t}.$$

This allows us to regard $\mathcal{D}_R$ as a filtered ring. Indeed, one has that

$$\mathcal{D}_R = \bigcup_{e\geqslant 0} \mathcal{D}_R^{(e)}, \text{ where } \mathcal{D}_R^{(e)} := R\langle D_{x_i,t} \mid i = 1, \dots, d \text{ and } 1 \leqslant t \leqslant p^e - 1\rangle.$$

Moreover, it is shown by A. Yekutieli (see [5, 1.4.9]) that $\mathcal{D}_R^{(e)} = \mathrm{End}_{R^{p^e}}(R)$, hence the previous filtration does not depend on the choice of coordinates.

Now, we fix additional notation; given an $\alpha = (a_1, \dots, a_d) \in \mathbb{N}^d$ we shall use the following multi-index notation: $\mathbf{x}^\alpha := x_1^{a_1} \cdots x_d^{a_d}$. With this notation, we set $||\alpha|| := \max\{a_1, \dots, a_d\}$; moreover, we also define $\deg(g)$ as the total degree of g. Finally, for any ideal $J \subseteq R$, $J^{[p^e]}$ will denote the ideal generated by all the p^e-th powers of elements in J, or equivalently the ideal generated by the p^e-th powers of any set of generators of J.

1.1. The ideal of p^e-th roots.

Due to the central role which the ideal of p^e-th roots plays throughout this article, we review some well-known definitions and facts (cf. [1, page 465]).

Definition 1.1. Given $g \in R$, and write $g = \sum_{0\leqslant||\alpha||\leqslant p^e-1} g_\alpha^{p^e} \mathbf{x}^\alpha$, then we set $I_e(g)$ to be the ideal of R generated by elements g_α.

We have the following easy properties (see [1, Lemma 3.2 and Lemma 3.4] for details).

Proposition 1.2. *Given $f \in R$ a non-zero polynomial, and given $e \geqslant 0$, one has that $I_e(f) = I_{e+1}(f^p)$, and $I_e(f^{p^e-1}) \supseteq I_{e+1}(f^{p^{e+1}-1})$.*

Note that Proposition 1.2 produces the following decreasing chain of ideals:

$$R = I_0(f^{p^0-1}) \supseteq I_1(f^{p-1}) \supseteq I_2(f^{p^2-1}) \supseteq I_3(f^{p^3-1}) \supseteq \dots \tag{1}$$

It is shown in [1] that under our assumptions this chain stabilizes. The smallest integer $e \in \mathbb{N}$ where the chain stabilizes plays a central role in this paper. We summarize the facts that we will need in the following theorem. See [1, Proposition 3.5, and Theorem 3.7] for details and proofs.

Theorem 1.3. *Let k be a perfect field of prime characteristic p. Let $R = k[x_1, \dots, x_d]$, and let $f \in R \setminus \{0\}$. Define $e := \inf\left\{ s \geqslant 1 \mid \ I_{s-1}\left(f^{p^{s-1}-1}\right) = I_s\left(f^{p^s-1}\right)\right\}$. Then, the following assertions hold.*

(i) The chain (1) stabilizes rigidly, that is $e < \infty$ and $I_{e-1}\left(f^{p^{e-1}-1}\right) = I_{e+s}\left(f^{p^{e+s}-1}\right)$ for any $s \geqslant 0$.

(ii) One has $e = \min\left\{ s \geqslant 1 \mid \ f^{p^s-p} \in I_s\left(f^{p^s-1}\right)^{[p^s]}\right\}$, and $e \leqslant \deg(f)$.

(iii) There is $\delta \in \mathcal{D}_R^{(e)}$ such that $\delta(f^{p^e-1}) = f^{p^e-p}$, or equivalently such that $\delta(1/f) = 1/f^p$.

(iv) There is no $\delta' \in \mathcal{D}_R^{(e')}$, with $e' < e$, such that $\delta'(1/f) = 1/f^p$.

Motivated by Theorem 1.3, we make the following definition.

Definition 1.4. For a non-zero polynomial $f \in R$, we call the integer e defined in Theorem 1.3 the *level of* f. Also, we will say that $\delta \in \mathcal{D}_R^{(e)}$ such that $\delta(f^{p^e-1}) = f^{p^e-p}$, or equivalently such that $\delta(1/f) = 1/f^p$, is a differential operator *associated with* f.

2. The algorithm

Let k be a computable perfect field of prime characteristic p (e.g., k is finite). Let $R = k[x_1, \dots, x_d]$, and let $f \in R$ be a non-zero polynomial. We now describe the algorithm that computes a differential operator $\delta \in \mathcal{D}_R$ associated with f.

- **Step 1.** Find the smallest integer $e \in \mathbb{N}$ with the property that $I_e(f^{p^e-p}) = I_{e-1}(f^{p^{e-1}-1}) = I_e(f^{p^e-1})$.
- **Step 2.** For $e \in \mathbb{N}$ as in **Step 1** write $f^{p^e-1} = \sum_{i=1}^n c_i^{p^e} \mu_i$, where $\{\mu_1, \dots, \mu_n\}$ is the basis of R as an R^{p^e}-module consisting of all the monomials $x_1^{a_1} \cdots x_d^{a_d}$, with $a_i \leqslant p^e - 1$ for all $i = 1, \dots, d$. In this situation, one can see that, for all $i = 1, \dots, n$, there exists $\delta_i \in \mathcal{D}_R^{(e)}$ such that $\delta_i(\mu_j) = 1$ if $i = j$ and $\delta_i(\mu_j) = 0$ if $i \neq j$.
- **Step 3.** Since $1 \in \mathcal{D}_R^{(e)}$, for $e \in \mathbb{N}$ as in **Step 1** we have

  $$f^{p^e-p} \in \mathcal{D}_R^{(e)}(f^{p^e-p}) = I_e(f^{p^e-p})^{[p^e]} = I_e(f^{p^e-1})^{[p^e]} = (c_1, \dots, c_n)^{[p^e]}.$$

  In particular there exist $\alpha_1, \dots, \alpha_n \in R$ such that $f^{p^e-p} = \sum_{i=1}^n \alpha_i c_i^{p^e}$. Consider $\delta_i \in \mathcal{D}_R^{(e)}$ as in **Step 2**, so that $\delta_i(f^{p^e-1}) = c_i^{p^e}$, and set $\delta := \sum_{i=1}^n \alpha_i \delta_i \in \mathcal{D}_R^{(e)}$. With this choice we have

  $$\delta(f^{p^e-1}) = \delta\left(\sum_{j=1}^n c_j^{p^e} \mu_j\right) = \sum_{i,j=1}^n c_j^{p^e} \alpha_i \delta_i(\mu_j) = \sum_{i=1}^n \alpha_i c_i^{p^e} = f^{p^e-p},$$

  and using that $\delta \in \mathcal{D}_R^{(e)}$ we finally get

  $$\delta\left(\frac{1}{f}\right) = \delta\left(\frac{f^{p^e-1}}{f^{p^e}}\right) = \frac{1}{f^{p^e}} \delta\left(f^{p^e-1}\right) = \frac{f^{p^e-p}}{f^{p^e}} = \frac{1}{f^p}.$$

We finish this section with an example where we develop the algorithm step by step for the reader's benefit.

Example 2.1. Let $f = x^2y^3z^5 \in R = \mathbb{Z}/2\mathbb{Z}[x,y,z]$. In this case, $p = 2$ and one can check that chain (1) stabilizes at $I_4(f^{2^4-1}) = (xy^2z^4)$. Notice that this equality of ideals follows from the following identity: $f^{15} = x^{30}y^{45}z^{75} = (xy^2z^4)^{16} \cdot (x^{14}y^{13}z^{11})$, so $e = 4$; this finishes Step 1. Now, in Step 2 we only need to find $\delta_1 \in \mathcal{D}_R^{(4)}$ such that $\delta_1(x^{14}y^{13}z^{11}) = 1$ and $\delta_1(x^iy^jz^k) = 0$ for any $0 \leq i,j,k \leq 15 = 2^4 - 1$ different from $i = 14, j = 13, k = 11$. This δ_1 turns out to be $(D_{x,15}D_{y,15}D_{z,15}) \cdot (xy^2z^4)$; this concludes Step 2. Finally, we move to Step 3; indeed, write $f^{p^4-p} = f^{2^4-2} = (x^{12}y^{10}z^6) \cdot (x^{16}y^{32}z^{64}) \in I_4(f^{15})^{[16]}$. Hence, in this case, $\alpha = x^{12}y^{10}z^6$, and therefore the differential operator we produce is

$$\delta = (x^{12}y^{10}z^6) \cdot (D_{x,15}D_{y,15}D_{z,15}) \cdot (xy^2z^4).$$

3. The case of elliptic curves

Let $p \in \mathbb{Z}$ be a prime and let $\mathcal{C} \subseteq \mathbb{P}^2_{\mathbb{F}_p}$ be an elliptic curve defined by an homogeneous cubic $f(x,y,z) \in \mathbb{F}_p[x,y,z]$; we want to review here the following notion.

Definition 3.1. $\mathcal{C}$ is said to be ordinary if the monomial $(xyz)^{p-1}$ appears in the expansion of f^{p-1} with non-zero coefficient. Otherwise, $\mathcal{C}$ is said to be supersingular.

It is known that $\mathcal{C}$ is ordinary if and only if f has level one; the main result of this section is the following:

Theorem 3.2. *$\mathcal{C}$ is supersingular if and only if f has level two.*

References

[1] J. Àlvarez Montaner, M. Blickle, and G. Lyubeznik. Generators of D-modules in positive characteristic. *Math. Res. Lett.*, 12(4):459–473, 2005.

[2] Alberto F. Boix, Alessandro De Stefani, and Davide Vanzo. An algorithm for constructing certain differential operators in positive characteristic. *Matematiche (Catania)*, 70(1):239–271, 2015.

[3] A. Grothendieck. Éléments de géométrie algébrique. IV. Étude locale des schémas et des morphismes de schémas IV. *Inst. Hautes Études Sci. Publ. Math.*, (32), 1967.

[4] S. B. Iyengar, G. J. Leuschke, A. Leykin, C. Miller, E. Miller, A. K. Singh, and U. Walther. *Twenty-four hours of local cohomology*, volume 87 of *Grad. Stud. Math.* American Mathematical Society, Providence, RI, 2007.

[5] A. Yekutieli. An explicit construction of the Grothendieck residue complex. *Astérisque*, (208):127, 1992. With an appendix by Pramathanath Sastry.

Department of Economics and Business, Universitat Pompeu Fabra, Jaume I Building, Ramon Trias Fargas 25-27, 08005 Barcelona, Spain.

E-mail address: alberto.fernandezb@upf.edu

Department of Mathematics,University of Virginia, 141 Cabell Drive, Kerchof Hall Charlottesville, VA 22903, USA.

E-mail address: ad9fa@viginia.edu

Departimento di Matematica e Informatica, Universitá di Firenze, Viale Morgagni, 67/a - 50134 Firenze, Italy.

E-mail address: davide.vanzo@unifi.it

CANONICAL REPRESENTATION OF CONSTRUCTIBLE SETS

JOSEP M. BRUNAT AND ANTONIO MONTES

Abstract. Constructible sets are needed in many algorithms of Computer Algebra, particularly in the Gröbner Cover and other algorithms for parametric polynomial systems. In this extended abstract we summarize the canonical form of constructible sets. The full paper has recently been published in [2].

Introduction

The existence of the canonical level sequence of locally closed sets of a constructible set was already proved by [1], in the context of general topology, and more recently studied by [3] in the context of Zariski topology. The object of this work is, taken this last description as starting point, to give formulas and algorithms for computing it effectively.

In [4] we already gave the canonical representations of locally closed sets as well as the algorithms for computing them. Here we generalize them to constructible sets. In Section 1, we remember the two canonical representations of locally closed sets and an algorithm PCrep for computing them, that is central for our purposes[1]. In Section 2, we recall the canonical structure of constructible sets given in [3], complementing it with dimension characteristics and an effective formula. This formula allows us to give an algorithm for building the canonical representation of constructible sets.

Some remarks about notation. All along this extended abstract we use the notations $\subseteq$ and $\subset$ to represent inclusion and strict inclusion, respectively. If $r \geq 1$ is an integer the symbol $[r]$ means the set $[r] = \{i \in \mathbb{N} : 1 \leq i \leq r\}$. For a set $S \subseteq \mathbb{C}^n$, the complementary set $\mathbb{C}^n \setminus S$ of S is denoted S^c. Finally $A \uplus B$ means disjoint reunion, that is, $A \cup B$ with the additional information that $A \cap B = \emptyset$.

1. Canonical representations of locally closed sets

A set $S \subseteq \mathbb{C}^n$ is *locally closed* if it is the intersection of an open and a closed set. We consider the $\mathbb{Q}$-Zarisky topology of $\mathbb{C}^m$, where the closed sets are varieties defined by polynomials in $\mathbb{Q}[\mathbf{x}]$ taking values in $\mathbb{C}^m$. In this context, a locally closed set is the set of points of $\mathbb{C}^m$ defined by a difference of varieties $S = \mathbf{V}(E) \setminus \mathbf{V}(N)$, where E, N are ideals.

We summarize here the two canonical representations of a locally closed set whose details can be consulted in [4].

Let S be a locally closed set, and $\overline{S}$ be its closure. For a locally closed set, $\overline{S}$ and $\overline{S} \setminus S$ are closed. So, there exist radical ideals $\mathfrak{a}$ and $\mathfrak{b}$ such that

$$\overline{S} = \mathbf{V}(\mathfrak{a}) \qquad \text{and} \qquad \overline{S} \setminus S = \mathbf{V}(\mathfrak{b}).$$

[1] Algorithms are not detailed in this extended abstract.

These ideals satify

$$S = \overline{S} \setminus (\overline{S} \setminus S) = \mathbf{V}(\mathfrak{a}) \setminus \mathbf{V}(\mathfrak{b}). \tag{1}$$

Taking into account the one-to-one correspondence between radical ideals and varieties, the ideals $\mathfrak{a} = \mathbf{I}(\overline{S})$ and $\mathfrak{b} = \mathbf{I}\left(\overline{S} \setminus S\right)$ are uniquely determined by S. The pair $\textsc{Crep}(S) = [\mathfrak{a}, \mathfrak{b}]$ is called the C-*canonical representation* of the locally closed set S. It is canonical in the sense that it does not depend on how the locally closed set S is given: it depends only on S. The set $\mathbf{V}(\mathfrak{a})$ (or $\mathfrak{a}$) is called the *top* of S, whereas $\mathbf{V}(\mathfrak{b})$ (or $\mathfrak{b}$) is called the *hole* of S.

We can further decompose $\textsc{Crep}(S) = [\mathfrak{a}, \mathfrak{b}]$ and obtain another representation of S. Let $\{\mathfrak{p}_i : i \in [r]\}$ be the prime decomposition of $\mathfrak{a}$ and for $i \in [r]$ let $\{\mathfrak{p}_{ij} : j \in [r_i]\}$ be the prime decomposition of $\mathfrak{p}_i + \mathfrak{b}$. The set

$$\textsc{Prep}(S) = \{[\mathfrak{p}_i, \{\mathfrak{p}_{ij} : j \in [r_i]\}] : i \in [r]\} \tag{2}$$

is called the P-representation of S. Note that it only depends on S. Each $[\mathfrak{p}_i, \{\mathfrak{p}_{ij} : j \in [r_i]\}]$ is called a *component* of S, from which $\mathbf{V}(\mathfrak{p}_i)$ (or $\mathfrak{p}_i$) is the *top* and $\mathbf{V}(\mathfrak{p}_{ij})$ (or $\mathfrak{p}_{ij}$) with $j \in [r_i]$ its *holes*, and we have

$$S = \bigcup_{i=1}^{r} \left(\mathbf{V}(\mathfrak{p}_i) \setminus \left(\bigcup_{j=1}^{r_i} \mathbf{V}(\mathfrak{p}_{ij} \right) \right)$$

Proposition 1.1. *Let S be a non empty locally closed set with*

$$\textsc{Crep}(S) = [\mathfrak{a}, \mathfrak{b}] \qquad \textit{and} \qquad \textsc{Prep}(S) = \{[\mathfrak{p}_i, \{\mathfrak{p}_{ij} : j \in [r_i]\}] : i \in [r]\}.$$

Then

(i) $\dim\ \mathbf{V}(\mathfrak{p}_{ij}) < \dim\ \mathbf{V}(\mathfrak{p}_i)$ *for all* $i \in [r]$ *and* $j \in [r_i]$*;*
(ii) $\dim \mathbf{V}(\mathfrak{b}) < \dim \mathbf{V}(\mathfrak{a})$.

Algorithms for computing Crep ans Prep are given in [4].

2. Canonical representation of constructible sets

A set $S \subseteq \mathbb{C}^n$ is *constructible* if it is a finite union of locally closed sets. In particular, locally closed sets are constructible. Constructible sets appear naturally in solving parametric polynomial systems of equations. Many authors give special representations for constructible sets adequate for its goals. Our goal is developing the invariant sequence of a constructible set described in [3] setting the outlook on its effective computation, to generalize the Crep of a locally closed set.

Lets denote $\mathcal{L}$ the family of locally closed sets and $\mathcal{C}$ the family of constructible sets.

Remark 2.1. If S_1 and S_2 are constructible, then $S_1 \cup S_2$, $S_1 \cap S_2$ and S_1^c are constructible sets, too. Thus $\mathcal{C}$ is a Boolean algebra of subsets of $\mathbb{C}^n$ containing $\mathcal{L}$. On the other hand, if a Boolean algebra $\mathcal{A}$ contains $\mathcal{L}$ then it must contain the finite union of locally closed sets, that is, $\mathcal{C} \subseteq \mathcal{A}$. We conclude that $\mathcal{C}$ is the Boolean algebra generated by $\mathcal{L}$. Let $\mathcal{T}$ be the union of the family of open sets and the family of closed sets. The boolean algebra generated by $\mathcal{T}$ contains $\mathcal{L}$, so $\mathcal{C}$ is also the boolean algebra generated by $\mathcal{T}$.

The first step of the construction of the canonical structure of the constructible set S given as a union of locally closed sets is to separate $\overline{S}$ into two disjoint sets: $\overline{S} = S \uplus C$ where C is the complement of S with respect to $\overline{S}$. Having this in mind we define the maps:

$$\begin{array}{rcl} \mathbf{C}\colon \mathcal{C} & \to & \mathcal{C} \\ S & \mapsto & \mathbf{C}(S) = \overline{S} \setminus S \end{array} \qquad \begin{array}{rcl} \mathbf{L}\colon \mathcal{C} & \to & \mathcal{L} \\ S & \mapsto & \mathbf{L}(S) = \overline{S} \setminus \overline{C} \end{array}$$

and we have $\mathbf{L}(S) \subseteq S$.

For a constructible set S, the set $\mathbf{L}(S)$ can be characterized as the largest locally closed set included in S. We give a Proposition that determines an explicit expression of C as a union of locally closed sets in terms of the input expression of S.

Proposition 2.2. *Let $S = S_1 \cup \cdots \cup S_r$ be a constructible set with each S_i locally closed. For $i \in [r]$ let $\textsc{Crep}(S_i) = [\mathfrak{a}_i, \mathfrak{b}_i]$, $V_i = \mathbf{V}(\mathfrak{a}_i)$ and $W_i = \mathbf{V}(\mathfrak{b}_i)$. Then,*

$$C = \overline{S} \setminus S = \bigcup_{T \subset [r]} \left(\left(\bigcap_{j \in T} V_j^c \right) \cap \left(\bigcap_{j \notin T} W_j \right) \right) = \bigcup_{T \subset [r]} \left(\left(\bigcap_{j \notin T} W_j \right) \setminus \left(\bigcup_{j \in T} V_j \right) \right).$$

Proposition 2.2 provides an explicit formula of $C = \mathbf{C}(S) = \overline{S} \setminus S$, as a union of locally closed sets. We can compute the $\textsc{Crep}$ of each one of these subsets of C and obtain an expression that allows us to handle $C \subset \overline{S}$ in the same way as we have done with S. This provides an iterative method to build the canonical representation of S. Next Proposition summarizes the basic properties of the first step in the recursive construction.

Proposition 2.3. *Let $S \neq \emptyset$ be a constructible set, $C = \mathbf{C}(S)$, $L = \mathbf{L}(S)$, $\mathfrak{a} = \mathbf{I}(S)$ and $\mathfrak{b} = \mathbf{I}(C)$. Then,*

(i) $C \subset \overline{S}$;
(ii) $\overline{C} \subset \overline{S}$;
(iii) $\overline{S} = \overline{L}$;
(iv) $[\mathfrak{a}, \mathfrak{b}] = [\mathbf{I}(S), \mathbf{I}(C)] = [\mathbf{I}(\overline{S})], \mathbf{I}(\overline{C})]$ *is the C-representation of L.*
(v) $\dim C < \dim S$.

We proceed now to describe the method for obtaining the canonical representation. Let S be a constructible set. Define the sequence (A_i) by

$$A_1 = S, \quad A_{i+1} = \mathbf{C}(A_i).$$

By Proposition 2.3 (ii) and (v), if $A_i \neq \emptyset$, we have $\overline{A_i} \supset \overline{A_{i+1}}$ and $\dim \overline{A_i} > \dim \overline{A_{i+1}}$. Therefore, there exists an integer $k \geq 1$ such that $A_{k+1} = \emptyset$ and A_k is closed. Consider the finite sequences

$$\begin{aligned} & S = A_1, A_2, \ldots, A_k, A_{k+1} = \emptyset \\ & \overline{S} = \overline{A_1} \supset \overline{A_2} \supset \cdots \supset \overline{A_{k+1}} = \emptyset, \\ & \dim(S) = \dim(A_1) > \dim(A_2) > \ \ldots \ > \dim(A_{k+1}) = -1. \end{aligned} \tag{3}$$

By construction $A_2 = \mathbf{C}(A_1) = \overline{S} \setminus S$ is disjoint with $S = A_1$. But $A_3 = \overline{A_2} \setminus A_2$ is disjoint with A_2 and a subset of S. Thus, have two decreasing and disjoint subsequences

$$\begin{aligned} S = & A_1 \supset A_3 \supset \cdots \supset A_{2\ell \pm 1}, \\ C = & A_2 \supset A_4 \supset \cdots \supset A_{2\ell}. \end{aligned}$$

Applying $\mathbf{L}$ to sequence (3), i.e. $L_i = \overline{A_i} \setminus \overline{A_{i+1}}$, we get a new sequence of disjoint sets that fill the whole $\overline{S}$,

$$L_1 = \overline{A_1} \setminus \overline{A_2}, \quad L_2 = \overline{A_2} \setminus \overline{A_3}, \ldots, L_k = \overline{A_k} \setminus \overline{A_{k+1}} = \overline{A_k}$$

so that

$$\overline{S} = \overline{A_1} = \overline{A_1} \setminus \overline{A_{k+1}} = L_1 \uplus L_2 \uplus \cdots \uplus L_k.$$

As the L_i belong alternatively to S and to C the previous sequence is divided into

$$S = L_1 \uplus L_3 \uplus \cdots \uplus L_{2\ell \pm 1}, \tag{4}$$

$$C = L_2 \uplus L_4 \uplus \cdots \uplus L_{2\ell}. \tag{5}$$

The odd disjoint locally closed subsets $L_1, L_3 \ldots L_{2\ell \pm 1}$ in which S is decomposed by the above procedure form the *canonical structure of the constructible set* S and is independent of the initially given locally closed sets defining S. We also obtain the canonical structure of the complement $C = \overline{S} \setminus S$ as the union of the even locally closed subsets $L_2 \uplus L_4 \uplus \cdots \uplus L_{2\ell}$. From them it is obvious how to obtain the *canonical representation* of S and C whose levels are already given by their CREP's.

For $i \in [k]$, define the ideals $\mathfrak{a}_i = \mathbf{I}(\overline{A_i})$. By using Proposition 2.3 (iv) and (v) it results

$$\begin{aligned}
&L_i = \mathbf{V}(\mathfrak{a}_i) \setminus \mathbf{V}(\mathfrak{a}_{i+1}),\\
&\text{CREP}(L_i) = [\mathfrak{a}_i, \mathfrak{a}_{i+1}],\\
&\dim \mathbf{V}(\mathfrak{a}_i) > \dim \mathbf{V}(\mathfrak{a}_{i+1}),\\
&\mathbf{I}(S) = \mathfrak{a}_1 \subset \mathfrak{a}_2 \subset \cdots \subset \mathfrak{a}_{k+1} = \langle 1 \rangle,\\
&\overline{S} = \mathbf{V}(\mathfrak{a}_1) \supset \mathbf{V}(\mathfrak{a}_2) \supset \mathbf{V}(\mathfrak{a}_3) \supset \cdots \supset \mathbf{V}(\mathfrak{a}_{k+1}) = \emptyset
\end{aligned}$$

We give the algorithms[2] FIRSTLEVEL and CONSLEVELS for computing the canonical representation of constructible sets, that have been included in the last version of the [5] GROBCOV library. We also give examples.

REFERENCES

[1] J.P.. Allouche, Note on the constructible sets of a topological space, in: Papers on general topology and applications (Gorham, ME, 1995), 1–10, Ann. New York Acad. Sci., 806, 1996.

[2] J.M. Brunat, A. Montes, Computing the Canonical Representation of Constructible Sets, Math. Comput. Sci. 10:1 165–178 (2016).

[3] J. O'Halloran, M. Schilmoeller, Gröbner bases for constructible sets, Comm. Algebra 30:11 (2002) 5479–5483.

[4] A. Montes, M. Wibmer, Gröbner bases for polynomial systems with parameters, J. Symbolic Comput., 45 (2010) 1391–1425.

[5] W. Decker, G.-M. Greuel, G. Pfister, H. Schönemann. SINGULAR 4-0-2. A computer algebra system for polynomial computations. `http://www.singular.uni-kl.de`.

Universitat Politècnica de Catalunya
E-mail address: `antonio.montes@upc.edu`

[2]Not detailed in this extended abstract.

A GCD ALGORITHM BY VALUES

JORGE CARAVANTES, GEMA M. DIAZ-TOCA, AND HENRI LOMBARDI

Abstract. In this note we introduce an algorithm to compute the greatest common divisor by values.

Keywords: Lagrange Interpolation; working with values; and gcd.

1. An algorithm

Let $\mathbb{K}[x]$ be the space of polynomials with coefficients in a field $\mathbb{K}$. The following lines introduce an algorithm to compute the greatest common divisor of two polynomials $f(x)$ and $g(x)$ in $\mathbb{K}[x]$ different from zero.

Data: Two nonzero polynomials $f(x)$ and $g(x)$, $df \geq$degree(f), $dg \geq$degree(g)
Result: gcd(f,g)
if $df < dg$ **then**
  $faux := f$, $dfaux := df$;
  $f := g, g := faux; df := dg; dg := dfaux$;
end
if $g = 0$ **then**
  return (f)
else
  if $dg = 0$ **then**
    return 1
  end
end
Choose α in $\mathbb{K}$, $h(x) := 1$.
if $g(\alpha) \neq 0$ **then**
  $\tilde{f}(x) := \Big(g(\alpha) * f(x) - f(\alpha) * g(x)\Big)/(x - \alpha)$, $df := df - 1$;
  $\tilde{g}(x) := g(x)$;
else
  if $f(\alpha) \neq 0$ **then**
    $\tilde{f}(x) := f(x)$;
    $\tilde{g}(x) := g(x)/(x - \alpha)$, $dg := dg - 1$;
  else
    $h(x) := (x - \alpha)$;
    $\tilde{f}(x) := f(x)/h(x)$, $df := df - 1$;
    $\tilde{g}(x) := g(x)/h(x)$, $dg := dg - 1$;
  end
end
return $h * \gcd(\tilde{f}, \tilde{g})$ with df and dg;

Algorithm 1: A gcd algorithm

Lemma 1.1. *Algorithm 1 terminates and returns the greatest common divisor of f and g.*

Proof. Given the first conditional, we can suppose that $df \geq dg$ during all the proof. If $g = 0$, then $\gcd(f, g)$ is a multiple of f and the algorithm returns it.

If $df = 0$ or $dg = 0$, then one of the polynomials is constant and then $\gcd(f, g) = 1$ and the algorithm returns it.

Otherwise before the recursive call, at least one among df and dg is lowered by 1. Since both df and dg must begin as positive integers, we conclude that, in a finite number of iterations, $df = 0$ or $dg = 0$ must be reached. Hence the algorithm terminates.

Regarding the output, let us suppose that, for numbers $\tilde{df}$ and $\tilde{dg}$ such that $\tilde{df} \leq df$, $\tilde{dg} \leq dg$ and $\tilde{df} + \tilde{dg} < df + dg$, the algorithm returns the greatest common divisor of the input. Then,

- if $g(\alpha) \neq 0$, let $\hat{f}$ be $g(\alpha) * f - f(\alpha) * g$. Then $f = \frac{1}{g(\alpha)}(\hat{f} + f(\alpha)g)$, so $\gcd(\hat{f}, g) = \gcd(f, g)$. Since $(x-\alpha)$ is a factor of $\hat{f}$ but not a factor of g, we then conclude that, since $(x-\alpha)\tilde{f} = \hat{f}$, and $\mathbb{K}[x]$ is a Euclidean domain,
$$\gcd(f, g) = \gcd(\hat{f}, g) = \gcd(\tilde{f}, g).$$
- if $g(\alpha) = 0 \neq f(\alpha)$, then $(x - \alpha)$ is a factor of $g = (x - \alpha)\tilde{g}$, but not a factor of f. Then, the structure of $\mathbb{K}[x]$ grants that $\gcd(f, g) = \gcd(f, \tilde{g})$.
- if $g(\alpha) = f(\alpha) = 0$, then $(x - \alpha)$ is a common factor of $f = (x - \alpha)\tilde{f}$ and $g = (x - \alpha)\tilde{g}$. Then, $\gcd(f, g) = (x - \alpha) * \gcd(\tilde{f}, \tilde{g})$.

So we proved by induction that the algorithm returns the greatest common divisor of f and g. □

Remark 1.2. Algorithm 1 needs less resources when compared with Euclid's one. Instead of the whole euclidean division that is performed classically to find the gcd, we just need to divide by degree one polynomials. Moreover, we do not need the exact degree of the polynomials, but just a bound.

While this note just focuses on working with values, it is possible this algorithm works with references in the space of polynomials different from the monomial and Lagrange bases.

2. Working with values

With the phrase "working with values", one can think about two paradigms. One of them is having an expression that is easy to evaluate but difficult to manipulate like $p(x) = (x-3)^{10}+(x+2)^{10}$. One can easily obtain $p(\alpha)$ for any $\alpha \in \mathbb{K}$, but working with an expanded $p(x)$ requires more resources. Moreover, as a change of base it is, the expansion would produce numerical errors when working with floating point numbers. Algorithm 1 can be easily adapted to this representation of polynomials:

Example 2.1. Consider $f(x) = (x-2)^4 + (x-4)^3$, $g(x) = (x-3)^3$ in $\mathbb{R}[x]$. Then $df = 4$, $dg = 3$. For the first call, we choose $\alpha = 1$, so that $f(1) = -26 \neq 0 \neq g(1) = -8$ then we get

$$f_1(x) = \frac{-8(x-2)^4 - 8(x-4)^3 + 26(x-3)^3}{x-1}, \quad g_1(x) = (x-3)^3, \text{ with } df_1 = dg_1 = 3.$$

Now we chooose $\alpha = 2$ (evaluating $f_1(1)$ is impossible). Since $f_1(2) = 38 \neq 0 \neq g_1(2) = -1$, then

$$f_2(x) = \frac{1}{x-2}\left(\frac{8(x-2)^4 + 8(x-4)^3 - 26(x-3)^3}{(x-1)} - 38(x-3)^3\right), \; g_2(x) = (x-3)^3, \; df_2 = 2, dg_2 = 3.$$

We now swap f_2 and g_2, since $df < dg$, and choose $\alpha = 0$. Then, $f_2(0) = -27 \neq 0 \neq g_2(0) = -354$,

$$f_3(x) = \frac{1}{x}\left(-354(x-3)^3 + 27\frac{1}{x-2}\left(\frac{8(x-2)^4 + 8(x-4)^3 - 26(x-3)^3}{(x-1)} - 38(x-3)^3\right)\right), g_3 = g_2,$$

with $df_3 = 2 = dg_3$.

We next choose $\alpha = -1$, then $f_3(-1) = -6672 \neq 0 \neq g_3(-1) = -592$, so h is still 1. Then

$$f_4(x) = \frac{1}{x+1}\left(-592\left(\frac{1}{x}\left(-354(x-3)^3 + 27\frac{1}{x-2}\left(\frac{8(x-2)^4+8(x-4)^3-26(x-3)^3}{(x-1)} - 38(x-3)^3\right)\right)\right.\right.$$
$$\left.\left. +6672\frac{1}{x-2}\left(\frac{8(x-2)^4+8(x-4)^3-26(x-3)^3}{(x-1)} - 38(x-3)^3\right)\right)$$
$$g_4(x) = g_3(x), \text{ with } df_4 = 1, dg_4 = 2.$$

We now swap f_4 and g_4, since $df < dg$, and choose evaluate in $\alpha = -2$. Then $f_4(-2) = -890 \neq 0 \neq g_4(-2) = -47040$, so h remains the same again. We get then

$$f_5(x) = \frac{-47040 f_4(x) + 890 g_4(x)}{x+2}, \quad g_5(x) = g_4(x), \text{ with } df_5 = 1 = dg_5.$$

Finally $\alpha = -3$. Then $f_5(-3) = -8467200 \neq 0 \neq g_5(-3) = -56448$. We get then $f_6 = 0$, $g_6 = g_5 = 9408\ x - 28224$. Thus, we can conclude that $gcd(f, g) = 9408\ x - 28224$.

The other interpretation of the phrase "by values" consists of having several nodes $\sigma_0, ..., \sigma_d \in \mathbb{K}$. Then a polynomial f of degree $\leq d$ can be stored considering its coordinates in the **Lagrange basis** which happen to be $f(\sigma_0), ..., f(\sigma_d)$.

Remark 2.2. Working directly in the Lagrange basis is useful when the equation of the polynomials are not available. It is known the the conversion between different polynomial bases can be unstable and the instability increases with the degree (see [2]). Furthermore, it is quite fast to sum and (when possible) multiply polynomials coordinate-wise. It is also easy to divide, when the division is known to be exact. In such case, division is also coordinate-wise. Algorithm 1 performs sums of polynomials and products of scalars and polynomials. It also performs divisions, but they are always exact, and evaluations, made by using the barycentric form of the Lagrange basis (see [3]). This means that Algorithm 1 can be applied when working "by values" also in this sense. It returns the values of the GCD at these same nodes $\sigma_0, ..., \sigma_d \in \mathbb{K}$.

We now redo Example 2.1 working with Lagrange basis.

Example 2.3. We characterize a polynomial l of degree ≤ 4 with the coordinates $(l(-2), l(-1), l(0), l(1), l(2))$. In this case, we have $f = (40, -44, -48, -26, -8)$, $g = (-125, -64, -27, -8, -1)$ and suppose $df = dg = 4$. Since $f(3) = g(3) = 0$, we put $h = (-5, -4, -3, -2, -1)$ and

$$f_1 = \frac{f}{(-5,-4,-3,-2,-1)} = (-8, 11, 16, 13, 8);\ g_1 = \frac{g}{(-5,-4,-3,-2,-1)} = (25, 16, 9, 4, 1);$$

We have $df_1 = dg_1 = 3$. We consider next $\alpha = -3$: $f_1(-3) = -47 \neq 0 \neq g_1(-3) = 36$. Then

$$f_2 = \frac{36 f_1 + 47 g_1}{(1,2,3,4,5)} = \frac{(887, 1148, 999, 656, 335)}{(1,2,3,4,5)} = (887, 574, 333, 164, 67),$$

while $g_2 = g_1$. Then $df_2 = 2 < dg_2 = 3$, so we have to swap g_2 and f_2.

Now consider $\alpha = 4$. Again both are nonzero, $f_2(4) = 1 \neq 0 \neq g_2(4) = 89$, and

$$f_3 = \frac{89 f_2 - g_2}{(-6,-5,-4,-3,-2)} = \frac{(1338, 850, 468, 192, 22)}{(-6,-5,-4,-3,-2)} = (-223, -170, -117, -64, -11),$$

and $g_3 = g_2$. Then $df_3 = 2 = dg_3$. Now consider $\alpha = -4$. $f_3(-4) = -329 \neq 0 \neq g_3(-4) = 1729$. Both are nonzero, and thus

$$f_4 = \frac{1729 f_3 + 329 g_3}{(2,3,4,5,6)} = (-46872, -35028, -23184, -11340, 504), g_4 = g_3.$$

Since $df_4 = 1; dg_4 = 2$ and we swap g_4 and f_4. Now consider $\alpha = 5$, $f_4(5) = 208 \neq 0 \neq g_4(5) = 36036$. Both are nonzero, and so

$$f_5 = \frac{36036 f_4 - 208 g_4}{(-7,-6,-5,-4,-3)} = (-5959044, -4661748, -3364452, -2067156, -769860), g_5 = g_4.$$

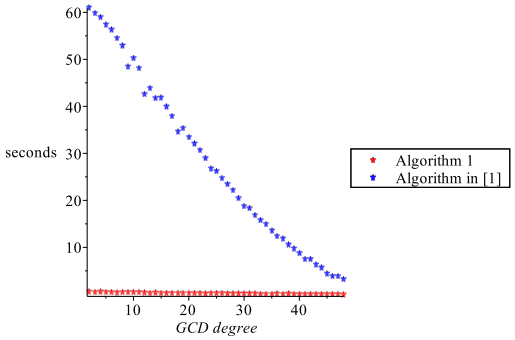
Finally consider $\alpha = -5$, $f_5(-5) = -9850932 \neq 0 \neq g_5(-5) = -82404$. Both are nonzero, and so

$$f_6 = \frac{-82404 f_5 + 9850932 g_5}{(3,4,5,6,7)} = (9772059024, 9772059024, 9772059024, 9772059024, 9772059024), g_6 = g_5.$$

Since f_6 represents a constant polynomial, we can conclude that the values of the GCD are given by $(-5,-4,-3,-2,-1)$.

3. Experiments and runtime

In the talk we will compare our algorithm with the algorithm introduced in [1]. In practice, our algorithm seems to be much faster. Observe for example Table 1 which is devoted to showing timings in the exact case " à la Lagrange". In order to generate the examples, each pair of polynomials $f(x)$ and $g(x)$ are the product of two random polynomials, one of them equal to the gcd. We have set $deg(g) = deg(f) = 50$ and the degree of the GCD range varies between 2 and 48. Moreover we will discuss the numerical behavior of both algorithms when working with "approximate" polynomials.



4. Acknowledgements

The first two authors are both partially supported by the Spanish Ministerio de Economía y Competitividad and by the European Regional Development Fund (ERDF), under the project MTM2014-54141-P. We would like to thank the anonymous referees for their helpful comments.

References

[1] H. Cheng, G. Labahn (2008), *Computing Polynomial LCM and GCD in Lagrange Basis*, ACM Commun. Comput. Algebra, Vol. 42, Issue 3, pp. 129–130.

[2] T. Hermann (1996), *On the stability of polynomial transformations between Taylor, Bézier, and Hermite forms*, Numerical Algorithms, Vol. 13, pp. 307–320.

[3] J.P. Berrut, L.N. Trefethen (2004), *Barycentric Lagrange interpolation*. SIAM Review, Vol. 46, Issue 3, pp. 501–517.

Universidad Complutense de Madrid
E-mail address: jcaravan@mat.ucm.es

Universidad de Murcia
E-mail address: gemadiaz@um.es

Université de Franche-Comté
E-mail address: henri.lombardi@univ-fcomte.fr

A REFINED ALGORITHM FOR TESTING THE LEIBNIZ n-ALGEBRA STRUCTURE

J. M. CASAS, M. A. INSUA, M. LADRA, AND S. LADRA

Abstract. We present a refinement of the algorithm given in [2] that checks if a multiplication table corresponds to a Leibniz n-algebra structure. This algorithm is based on the computation of a Gröbner basis of the ideal which is used in the construction of the universal enveloping algebra of a Leibniz algebra and it is implemented in a Mathematica notebook by means of the NCAlgebra package.

Essentially, the refinement consists of removing all the superfluous information in the generators of the ideal; this deletion allows us to decrease highly the computation time.

A comparative analysis between both implementations is provided.

Introduction

A Leibniz n-algebra [3] is a $\mathbb{K}$-vector space $\mathcal{L}$ endowed with an n-linear map $[-,\dots,-]$: $\mathcal{L}^{\otimes n} \to \mathcal{L}$ satisfying the Fundamental Identity

$$\text{(FI)} \qquad \big[[x_1,\dots,x_n],y_1,\dots,y_{n-1}\big] = \sum_{i=1}^{n} \big[x_1,\dots,x_{i-1},[x_i,y_1,\dots,y_{n-1}],x_{i+1},\dots,x_n\big]$$

for all $x_1,\dots,x_n,y_1,\dots,y_{n-1} \in \mathcal{L}$.

When the n-bracket is skew-symmetric, the structure is named Lie n-algebra. Lie (respectively, Leibniz) 2-algebras are exactly Lie (respectively, Leibniz) algebras.

For finite-dimensional Leibniz n-algebras with basis $\{e_1,\dots,e_d\}$, the n-ary bracket is determined by structure constants $c^k_{i_1,i_2,\dots,i_n}$ such that $[e_{i_1},e_{i_2},\dots,e_{i_n}] = \sum_{k=1}^{d} c^k_{i_1,i_2,\dots,i_n} e_k$.

The problem of identify a Leibniz n-algebra structure in a given n-ary bracket is the subject of the paper [2], where a computer program in Mathematica that checks if a multiplication table satisfies (FI) is implemented. The algorithm is based on the computation of a Gröbner basis of the ideal which appears in the construction of the universal enveloping algebra of a Leibniz algebra [5], by means of the NCAlgebra package [4] which enables the computation of Gröbner bases in non commutative associative algebras. This Gröbner basis provides a criterion in terms of existence of polynomials of degree 1 over convenient variables, which guarantees that the multiplication table corresponds to a Leibniz n-algebra or not. To decide whether a Leibniz n-algebra is a n-Lie algebra or not, it is necessary to check whether certain type of polynomials are equal to zero.

Nevertheless the implementation of this algorithm does not provide efficient results concerning times of computation in an Intel(R) Core(TM) i7-3770 CPU @ 3.40 GHz, 16 GB RAM, running Windows 7 (64 bits) with Mathematica®10. In Section 3 we show some low-dimensional tables of multiplication that require high times of computation. Hence our

goal in this talk is to present a refinement of the computer program in order to reduce times of computation and compare its efficiency with respect to the initial one given in [2].

1. The algorithm

Algorithm 1.1 (Leibniz n-algebra test).

`Input:` A $\mathbb{K}$-vector space $\mathcal{L}$ with basis $\{e_1, \dots, e_d\}$ and an n-linear map.

`Output:` The algorithm informs if the n-linear map endows the $\mathbb{K}$-vector space with a structure of Lie n-algebra or non-Lie Leibniz n-algebra or neither.

Step 1: Consider $\mathcal{D}_n(\mathcal{L}) = \mathcal{L}^{\otimes n-1}$ (where $\mathcal{D}_n$ is the Daletskii-Takhtajan's functor [3]) and the ideal $\Phi(I)$ [1] such that

$$UL(\mathcal{D}_n(\mathcal{L})) \cong \frac{\mathbb{K}\langle x_{1,\overset{n-1}{\dots},1}, \dots, x_{d,\overset{n-1}{\dots},d}, y_{1,\overset{n-1}{\dots},1}, \dots, y_{d,\overset{n-1}{\dots},d}\rangle}{\Phi(I)}.$$

Step 2: Compute a Gröbner basis corresponding to the ideal $\Phi(I)$ with respect to the degree lexicographical ordering on $\mathbb{K}\langle x_{1,\dots,1}, \dots, x_{d,\dots,d}, y_{1,\dots,1}, \dots, y_{d,\dots,d}\rangle$ with $y_{d,\dots,d} > \cdots > y_{1,\dots,1} > x_{d,\dots,d} > \cdots > x_{1,\dots,1}$.

Step 3: Check if there exists at least one polynomial of degree 1 in the variables $x_{1,\dots,1}, \dots, x_{d,\dots,d}$.

Step 3.1: If there does not exist such a polynomial, then the structure is a Lie n-algebra.

Step 3.2: Otherwise, check if there exists at least one polynomial of degree 1 in the variables $y_{1,\dots,1}, \dots, y_{d,\dots,d}$.

Step 3.2.1: If there exists such a polynomial, then the Fundamental Identity (FI) does not hold, thus the structure is not a Leibniz n-algebra.

Step 3.2.2: Otherwise, the Fundamental Identity (FI) holds, thus the structure is a Leibniz n-algebra.

Check if there exists at least one polynomial of degree 1 in the ideal $\langle\{g_{t_1,\dots,t_1,t_2,\dots,t_n} + g_{t_2,\dots,t_n,t_1,\dots,t_1}\}_\Omega\rangle \subset \Phi(I) \cap \mathbb{K}\langle x_{1,\dots,1}, \dots, x_{d,\dots,d}\rangle$, where $\Omega = \{e_{t_1}, \dots, e_{t_n} \in \{e_1, \dots, e_d\}, \exists i, j \in \{1, \dots, n\} \text{ with } i < j, e_{t_i} = e_{t_j}\}$.
(Note: $g_{i_1,\dots,i_{n-1},i_n,\dots,i_{2n-2}} = x_{i_1,\dots,i_{n-1}} x_{i_n,\dots,i_{2n-2}} - x_{i_n,\dots,i_{2n-2}} x_{i_1,\dots,i_{n-1}} - \Phi(r_{[e_{i_1}\otimes\cdots\otimes e_{i_{n-1}}, e_{i_n}\otimes\cdots\otimes e_{i_{2n-2}}]})$).

Step 3.2.2.1: If there exists such a polynomial, then the structure is a non-Lie Leibniz n-algebra.

Step 3.2.2.2: Otherwise, check if there exists at least one polynomial of degree 1 in the ideal:

$$\begin{aligned}\langle\{&g_{t_1,\dots,t_1,t_2,\dots,t_i,\dots,t_j,\dots,t_n} + g_{t_2,\dots,t_i,\dots,t_j,\dots,t_n,t_1,\dots,t_1} + g_{t_1,\dots,t_1,t_2,\dots,t_j,\dots,t_i,\dots,t_n} + \\ &g_{t_2,\dots,t_j,\dots,t_i,\dots,t_n,t_1,\dots,t_1}\}_{\{\forall e_{t_1},\dots,e_{t_n}\in\{e_1,\dots,e_d\} \text{ such that } \forall r,s\in\{1,\dots,n\}\ e_{t_r}\neq e_{t_s}\}}\rangle \cup \\ &\{g_{t_1,t_3,\dots,t_n,t_2,\dots,t_n} + g_{t_2,\dots,t_n,t_1,t_3,\dots,t_n} + g_{t_2,t_3,\dots,t_n,t_1,t_3,\dots,t_n} + g_{t_1,t_3,\dots,t_n,t_2,t_3,\dots,t_n}\} \\ &\subset \Phi(I) \cap \mathbb{K}\langle x_{1,\dots,1}, \dots, x_{d,\dots,d}\rangle.\end{aligned}$$

Step 3.2.2.2.1: If there exists such a polynomial, then the structure is a non-Lie Leibniz n-algebra.

Step 3.2.2.2.2: Otherwise, the structure is a Lie n-algebra.

2. Refinement of the algorithm

While we were developing the initial algorithm, it was very clear for us from the beginning [2] that the process could be quicker but the goal was not to get an efficient algorithm, at least at that point of the research. The main motivation was to prove that the Leibniz checking process can be done using Gröbner Bases. Proceeding in that way, we have enriched the problem and so it is possible to manage the situation from a different and useful point of view (Gröbner Basis Theory and Ideal Theory). Once the theoretical background is established, our interest changed from the existence to the efficiency of the algorithm.

The main idea, we followed to reduce computation time, was to avoid unnecessary computations and to remove all the superfluous information which is contained in the ideal.

The first criterion, we followed to avoid unnecessary computations, is the following one: if we examine the proof of [2, Proposition 3.6], it is possible to check that the set of the expressions $[e_i, g_t(e_1, \ldots, e_d)]$ ($g_t \in \mathcal{D}_n(\mathcal{L})^{\text{ann}}$) has an important role. If one of these brackets is not equal to zero, then the structure cannot be a Leibniz n-algebra (as if $\mathcal{L}$ is a Leibniz n-algebra, then $[-, (\mathcal{D}_n(\mathcal{L}))^{\text{ann}}] = 0$).

The second criterion, we followed to remove all the superfluous information, is the following one: if all the previous expressions are equal to zero, then it is easy to check, again following the proof of [2, Proposition 3.6], that it is possible to gather the information we need from a subideal of $\Phi(I)$, this subideal is $\langle \{x_i \cdot x_j - x_j \cdot x_i - \Phi(r_{[e_i,e_j]})\}_{i,j\in\{1,\ldots,m\},i<j} \bigcup \{y_i \cdot x_j - x_j \cdot y_i - \Phi(l_{[e_i,e_j]})\}_{i\in\{1,\ldots,d\},j\in\{1,\ldots,m\}} \rangle$, $m = \dim(\mathcal{D}_n(\mathcal{L})^{\text{ann}})$. A natural question arises at this point, is the Gröbner Basis of this subideal finite?, the answer is affirmative because all the reductions drive us to 0 or a polynomials of degree 1.

Using these two criteria and other minor computational aspects, such that, stop the computation process as soon as we obtain the information we need, helped us to construct a more efficient algorithm.

3. Some computations

This section is devoted to show some examples of computations with two programs in NCAlgebra [4] (a package running under Mathematica) that implements both algorithms and we compare its efficiency with respect to the time of computation. In Table 1 we compare the performance of both algorithms when they are applied on a same algebra. Here Algorithm 1 means the algorithm given in [2] and Algorithm 2 means the refined one described in Section 2.

Example 3.1. Let $\mathcal{L}$ be the $\mathbb{C}$-vector space with basis $\{e_1, e_2, e_3\}$ equipped with the 3-linear map $[e_{\sigma(1)}, e_{\sigma(2)}, e_{\sigma(3)}] = (-1)^{\epsilon(\sigma)} \cdot e_1 = (-1)^{\epsilon(\sigma)} \cdot (1, 0, 0), \sigma \in S_3$.

`It is a Leibniz 3-algebra and it is a Lie 3-algebra`.

Example 3.2. Let $\mathcal{L}$ be the $\mathbb{C}$-vector space with basis $\{e_1, e_2, e_3\}$ and the 3-linear map $[e_3, e_2, e_3] = -e_2 = (0, -1, 0)$, $[e_3, e_3, e_2] = e_2 = (0, 1, 0)$ and 0 otherwise.

`It is a Leibniz 3-algebra and it is not a Lie 3-algebra`.

Example 3.3. Let $\mathcal{L}$ be the $\mathbb{C}$-vector space with basis $\{e_1, e_2, e_3, e_4\}$ and the 3-linear map $[e_{\sigma(1)}, e_{\sigma(2)}, e_{\sigma(3)}] = [e_{\sigma(1)}, e_{\sigma(2)}, e_{\sigma(4)}] = (-1)^{\epsilon(\sigma)} e_3, \sigma \in S_3$, and 0 otherwise.

`It is a Leibniz 3-algebra and it is a Lie 3-algebra`.

Example 3.4. Let $\mathcal{L}$ be the $\mathbb{C}$-vector space with basis $\{e_1, e_2\}$ and the 5-linear map $[e_1, e_2, e_2, e_1, e_1] = e_2$ and 0 otherwise.

`It is not a Leibniz 5-algebra`

Example 3.5. Let $\mathcal{L}$ be the $\mathbb{C}$-vector space with basis $\{e_1, \ldots, e_{12}\}$ and the 2-linear map $[e_1, e_1] = e_{11}$ and 0 otherwise.

`It is a Leibniz 2-algebra and it is not a Lie 2-algebra`

Example	Time Algorithm 1	Time Algorithm 2
3.1	10.63 s.	1.50 s.
3.2	7.53 s.	0.92 s.
3.3	213.07 s.	19.61 s.
3.4	19.74 s.	0.09 s.
3.5	50.80 s.	9.50 s.

TABLE 1. Time comparison

REFERENCES

[1] J. M. Casas, M. A. Insua, M. Ladra, Poincaré-Birkhoff-Witt theorem for Leibniz n-algebras, J. Symbolic Comput. 42 (11–12) (2007), 1052–1065.
[2] J. M. Casas, M. A. Insua, M. Ladra, S. Ladra, Test for Leibniz n-algebra structure, Linear Algebra Appl. 494 (2016) 138–155.
[3] J. M. Casas, J.-L. Loday, T. Pirashvili, Leibniz n-algebras, Forum Math. 14 (2) (2002) 189–207.
[4] J. W. Helton, R. L. Miller, M. Stankus, NCAlgebra: A Mathematica package for doing noncommuting algebra, `http://math.ucsd.edu/~ncalg` (1996).
[5] M. A. Insua, M. Ladra, Gröbner bases in universal enveloping algebras of Leibniz algebras, J. Symbolic Comput. 44 (5) (2009) 517–526.

Dpto. Matemática Aplicada I, Univ. de Vigo, 36005 Pontevedra, Spain [1]
E-mail address: `jmcasas@uvigo.es`

Dpto. Matemática Aplicada I, Univ. de Vigo, 36310 Vigo, Spain
E-mail address: `avelino.ainsua@gmail.com`

Dpto. de Álxebra, Univ. de Santiago, 15782 Santiago, Spain
E-mail address: `manuel.ladra@usc.es`

Dpto. de Computación, Univ. de A Coruña, 15071 A Coruña, Spain
E-mail address: `susana.ladra@udc.es`

[1] The first three authors were supported by Ministerio de Economía y Competitividad (Spain), grant MTM2013-43687-P (European FEDER support included). The third author was also supported by Xunta de Galicia, grant GRC2013-045 (European FEDER support included)

ON ALGEBRAIC PROPERTIES OF THE HUMAN ABO-BLOOD GROUP INHERITANCE PATTERN

J. M. CASAS, M. LADRA, B. A. OMIROV, AND R. TURDIBAEV

Abstract. We generate an algebra on blood phenotypes with multiplication based on the human ABO-blood group inheritance pattern. We assume that gametes are not chosen randomly during meiosis. We investigate some of the properties of this algebra, namely, the set of idempotents, lattice of ideals and conditions for these algebras to be isomorphic.

1. Extended Summary

Before the discovery of blood groups more than a century ago by Karl Landsteiner [7], all human blood was assumed to be the same. Establishing the genetics of the ABO blood group system was one of the first breakthroughs in Mendelian genetics. There are three alleles or versions of the ABO-blood group genes - A, B and O. The allele O is recessive to A and B, and alleles A and B are co-dominant. It is known that humans are diploid organisms, which means that they carry a double set of chromosomes. Therefore, blood genotypes are determined by two alleles with six possible combinations: AA, BB, OO, AB, OA and OB. Since A and B dominate over O, the genotypes AO and AA express blood group A (phenotype A) and BO together with BB correspond to group B (phenotype B).

A number of papers is devoted to the study of distribution of blood group frequencies in different countries and ethnicities [3, 4, 6]. Some methods for estimating phenotype probabilities for ABO groups are developed and compared in [5]. Assuming the allele probabilities to be p, q and r for the genes A, B, and O, respectively, authors of [5] obtain some estimates on the probabilities that a person has a corresponding phenotype.

Most of the numerous papers (for example, [1, 2, 3, 5, 7, 9]) on the subject are dedicated to cases in which during the fertilization, parents' gametes are chosen randomly and in an independent way. Mendel's first law allows to quantify the types of gametes an individual can produce. For example, a person with genotype OA during meiosis produces gametes O and A with equal probability $\frac{1}{2}$, while an individual with blood group A during meiosis produces gamete O with probability $\frac{1}{4}$.

In this work we assume that all parents of blood groups A and B have equal probabilities to contribute with the allele O to a child's genotype and we denote this probability by $p_{O|A} = p_{O|B} = \alpha$. Furthermore, we assume that all parents with group AB contribute the allele A during meiosis with equal probability and we denote this probability by $p_{A|AB} = \beta$.

Consider the blood groups O, A, B and AB as basis elements of a 4-dimensional vector space and a bilinear operation $\circ$ as the result of meiosis. Under these assumptions above we have the following 10 formal equalities:

(i) $O \circ O = O$;
(ii) $O \circ A = p_{O|A}O + (1 - p_{O|A})A = \alpha O + (1 - \alpha)A$;

(iii) $O \circ B = p_{O|B}O + (1 - p_{O|B})B = \alpha O + (1 - \alpha)B$;
(iv) $O \circ AB = p_{A|AB}A + p_{B|AB}B = \beta A + (1 - \beta)B$;
(v) $A \circ A = p^2_{O|A}O + (1 - p^2_{O|A})A = \alpha^2 O + (1 - \alpha^2)A$;
(vi) $A \circ B = p_{O|A}p_{O|B}O + p_{A|A}p_{O|B}A + p_{O|A}p_{B|B}B + p_{A|A}p_{B|B}AB$
$= \alpha^2 O + \alpha(1 - \alpha)A + \alpha(1 - \alpha)B + (1 - \alpha)^2 AB$;
(vii) $A \circ AB = p_{A|AB}A + p_{O|A}p_{B|AB}B + p_{A|A}p_{B|AB}AB$
$= \beta A + \alpha(1 - \beta)B + (1 - \alpha)(1 - \beta)AB$;
(viii) $B \circ B = p^2_{O|B}O + (1 - p^2_{O|B})B = \alpha^2 O + (1 - \alpha^2)B$;
(ix) $B \circ AB = p_{O|B}p_{A|AB}A + p_{B|AB}B + p_{B|B}p_{A|AB}AB$
$= \alpha\beta A + (1 - \beta)B + (1 - \alpha)\beta AB$;
(x) $AB \circ AB = p^2_{A|AB}A + p^2_{B|AB}B + 2p_{A|AB}p_{B|AB}AB$
$= \beta^2 A + (1 - \beta)^2 B + 2\beta(1 - \beta)AB$.

Definition 1.1. A commutative four-dimensional $\mathbb{R}$-algebra with basis $\{O, A, B, AB\}$ and with multiplication $\circ$ satisfying equalities (i)–(x) is called a generalized ABO-blood group algebra (GBGA) and is denoted by $\mathcal{B}(\alpha, \beta)$.

We can consider the algebraic relations defining a GBGA from a different perspective.

Let x_1, x_2, x_3, x_4 be corresponding proportions of O, A, B, AB phenotypes in one population. Then we have the following equalities for the underlying allele frequencies

$$\begin{aligned} p_O &= x_1 + \alpha x_2 + \alpha x_3, \\ p_A &= (1 - \alpha)x_2 + \beta x_4, \\ p_B &= (1 - \alpha)x_3 + (1 - \beta)x_4. \end{aligned}$$

Straightforward computation of the frequencies of O, A, B and AB phenotypes in zygotes of the next generation (state) yields an extension of Hardy-Weinberg Law:

$$\begin{cases} x'_1 = p^2_O \\ x'_2 = p^2_A + 2p_A p_O \\ x'_3 = p^2_B + 2p_B p_O \\ x'_4 = 2p_A p_B. \end{cases}$$

Consider $\mathbb{S}^3 = \{\mathbf{x} = (x_1, x_2, x_3, x_4) \in \mathbb{R}^4 \mid x_1 + x_2 + x_3 + x_4 = 1, x_i \geq 0, 1 \leq i \leq 4\}$ a 3-dimensional canonical simplex. Following [8], we have a so-called evolutionary (quadratic stochastic) operator $V \colon \mathbb{S}^3 \to \mathbb{S}^3$ describing an evolution of the population mapping a state $\mathbf{x} = (x_1, x_2, x_3, x_4)$ to the next state $V(\mathbf{x}) = (x'_1, x'_2, x'_3, x'_4)$. By linearity V can be extended to $\mathbb{R}^4$ if necessary.

The relation that establishes a connection between the evolutionary operator V and the multiplication $\circ$ of a GBGA is $\mathbf{x} \circ \mathbf{x} = V(\mathbf{x})$ and consequently

$$\mathbf{x} \circ \mathbf{y} = \frac{1}{4}(V(\mathbf{x} + \mathbf{y}) - V(\mathbf{x} - \mathbf{y})).$$

In order to simplify our investigation of the structure of a GBGA we make some linear basis transformation and obtain a simpler table of multiplication of a GBGA:

$$\mathcal{B}'(\lambda,\beta): \begin{cases} o\circ o = o \\ o\circ a = a\circ o = \lambda a \\ o\circ b = b\circ o = \lambda b \\ a\circ a = a \\ b\circ b = b \\ a\circ b = b\circ a = \dfrac{\lambda-\beta}{\lambda}\cdot a + \dfrac{\lambda-(1-\beta)}{\lambda}\cdot b + ab, \end{cases}$$

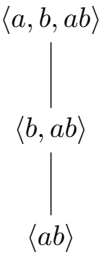
where $\lambda = 1-\alpha$ and omitted products are assumed to be zero.

Due to the convenience of the above products, we now investigate the algebraic properties of the algebra $\mathcal{B}'(\lambda,\beta)$. Note that $O \in \mathcal{B}(\alpha,\beta)$, and $o \in \mathcal{B}'(\lambda,\beta)$ are not zero of the algebras.

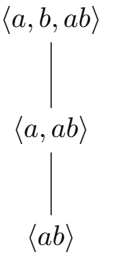
We investigate some of the properties of this algebra related to equilibrium (idempotents) and annihilation (solvable elements) of a population, dominating subpopulations (lattice of ideals) and behaviour to changes of parameters and present the results.

Theorem 1.2. *The lattices of ideals of corresponding algebras are:*

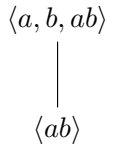
$\mathcal{B}'(\lambda,\lambda), \lambda \neq \frac{1}{2}$:



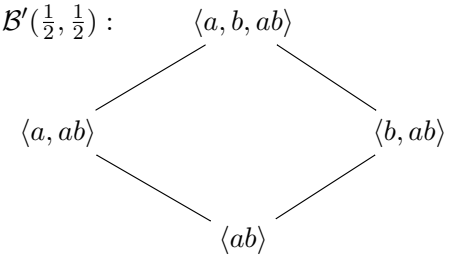
$\mathcal{B}'(\lambda,1-\lambda), \lambda \neq \frac{1}{2}$:



$\mathcal{B}'(\lambda,\beta), \beta \neq \lambda, \beta \neq 1-\lambda$:



$\mathcal{B}'(\frac{1}{2},\frac{1}{2})$:



Theorem 1.3. *For the algebra $\mathcal{B}'(\lambda,\beta)$ the set $\mathcal{I}$ of idempotents depending on the parameters λ,β is as follows:*

- $\mathcal{I} = \{o, a, b\}$ *if* $(\lambda,\beta) \in \{(\frac{1}{2},\frac{1}{4}),(\frac{1}{2},\frac{3}{4})\}$;
- $\mathcal{I} = \{o, a, b, j_0\}$ *if* $\lambda = \frac{1}{2}, \beta \neq \frac{1}{4}, \frac{3}{4}$;
- $\mathcal{I} = \{o,\ a,\ b,\ o+(1-2\lambda)a,\ o+(1-2\lambda)b\}$ *if* $(\lambda,\beta) \in P \cup \{(2\beta,\beta)|\beta \neq \frac{1}{4}\} \cup \{(2-2\beta,\beta)|\beta \neq \frac{3}{4}\}$;
- $\mathcal{I} = \{o,\ a,\ b,\ o+(1-2\lambda)a,\ o+(1-2\lambda)b,\ j_0,\ j_1\}$, *otherwise,*

where $j_\xi = \xi o + \rho_\xi(2\beta-\lambda)a + \rho_\xi(2-2\beta-\lambda)b + 2\rho_\xi^2(2\beta-\lambda)(2-2\beta-\lambda)ab$ *and* $\rho_\xi = \dfrac{\lambda(1-2\xi\lambda)}{-3\lambda^2+4\beta^2+4\lambda-4\beta}$ *for* $\xi = 0, 1$.

Denote by $P = \left\{(\lambda,\beta) \mid 0 < \lambda \leq \frac{1}{3},\ \beta = \frac{1}{2}\left(1 \pm \sqrt{(1-\lambda)(1-3\lambda)}\right)\right\}$.

Theorem 1.4. *For an algebra of ABO-blood group $\mathcal{B}'(\lambda,\beta)$ to admit a solvable element of index $n \geq 3$ it is necessary and sufficient that $(\lambda,\beta) \in P$.*

Moreover, solvable elements of degree n are

$$-2^{n-4}\left(\frac{\lambda+\beta-1}{\lambda}\right)^{n-4} ta + tb + sab, \quad \textit{where} \quad t,s \in \mathbb{R},\ t \neq 0.$$

We establish that two isolated populations with different values of initial chosen parameters (unless the probabilities that groups A and B contribute with allele O during meiosis are equal in both populations and probabilities of group AB contributing with allele A during meiosis in both populations add up to 1) have different non-isomorphic corresponding algebras that describe heredity (evolution of population).

Theorem 1.5. *Two distinct ABO-blood group algebras $\mathcal{B}'(\lambda,\beta)$ and $\mathcal{B}'(\lambda',\beta')$ are isomorphic if and only if $\lambda' = \lambda$ and $\beta' = 1-\beta$.*

References

[1] F. Bernstein, Über die Erblichkeit der Blutgruppen, *Zeitschrift für Induktive Abstammungs- und Vererbungslehre* 54(1), 1930, 400–426.

[2] S. N. Bernstein, Solution of a mathematical problem connected with the theory of heredity, *Ann. Math. Statistics* 13, 1942, 53–61.

[3] R. Chakraborty, Gene frequency estimates in the ABO system and their efficiencies, *Sankhya, Series B* 32, 1970, 21–26.

[4] Y. Fujita, M. Tanimura and K. Tanaka, The distribution of the ABO blood groups in Japan, *Japanese Journal of Human Genetics* 23, 1978, 63–109.

[5] S. R. Greenwood and G. A. F. Seber, Estimating Blood Phenotype Probabilities and Their Products, *Biometrics* 48(1), 1992, 143–154.

[6] S.H. Kang, Y. Fukumori, S. Ohnoki, H. Shibata, K.S. Han, et al., Distribution of abo genotypes and allele frequencies in a korean population, *Japanese Journal of Human Genetics* 42, 1997, 331–335.

[7] K. Landsteiner, Zur Kenntnis der antifermentativen, lytischen und agglutinierenden Wirkungen des Blutserums und der Lymphe, *Zentralblatt Bakteriologie* 27, 1900, 357–362.

[8] Y. I. Lyubich, *Mathematical structures in population genetics*, Springer-Verlag, Berlin, 1992.

[9] T. Sadykov, Polynomial dynamics of human blood genotypes frequencies, *Journal of Symbolic Computation*, 2016.

Dpto. Matemática Aplicada I, Universidad de Vigo, E. E. Forestal, Campus Universitario A Xunqueira, 36005 Pontevedra, Spain [1]
E-mail address: jmcasas@uvigo.es

Department of Algebra, University of Santiago de Compostela, 15782, Spain
E-mail address: manuel.ladra@usc.es

Institute of Mathematics, National University of Uzbekistan, Tashkent, 100125, Uzbekistan
E-mail address: omirovb@mail.ru

Department of Algebra, University of Santiago de Compostela, 15782, Spain
E-mail address: rustamtm@yahoo.com

[1] The authors were supported by Ministerio de Economía y Competitividad (Spain), grant MTM2013-43687-P (European FEDER support included). The second and fourth authors were also supported by Xunta de Galicia, grant GRC2013-045 (European FEDER support included)

ON THE RESOLUTION OF FAN ALGEBRAS OF PRINCIPAL IDEALS OVER NOETHERIAN RINGS

TERESA CORTADELLAS BENÍTEZ, CARLOS D'ANDREA, AND FLORIAN ENESCU

Abstract. We compute a minimal resolution of fan algebra over principal ideals in Noetherian rings, in the case where the support of the fan is contained in a strict halfplane.

1. Introduction

Let m, $n \in \mathbb{N}$, $p_1, \ldots, p_n \in \mathrm{R}$, $\mathcal{F}$ a fan in $\mathbb{R}^m$ such that its support $|\mathcal{F}|$ is a convex set, and $f_1, \ldots, f_n$ fan linear maps on $\mathcal{F}$ (see Definition 1.3 for more details). With all this data, one can consider the following algebra

$$\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}} = \sum_{\mathbf{v}\in|\mathcal{F}|\cap\mathbb{Z}^m} \langle p_1^{f_1(\mathbf{v})} \cdot \ldots \cdot p_n^{f_n(\mathbf{v})}\rangle \mathbf{x}^{\mathbf{v}} \subset \mathrm{R}[x_1^{\pm 1}, x_2^{\pm 1}, \ldots, x_m^{\pm 1}], \tag{1}$$

where $x_1, \ldots, x_m$ are indeterminates, $\mathbf{x} = x_1 \ldots x_m$, $\mathbf{p} = (p_1, \ldots, p_n)$, and $\mathbf{f} = (f_1, \ldots, f_n)$. We will call $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}}$ *the fan algebra* associated to the data $(\mathbf{f}, \mathcal{F}, \mathbf{p})$. It is a natural combinatorial object associated to a fan, and we are interested in its main algebraic properties. In this paper, we will describe the free resolution of $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}}$ as an-R-algebra, when $m = 2$, R is Noetherian, and $|\mathcal{F}|$ does not contain lines.

Fan algebras have been introduced in [3] (see also [2, 1]), but here we extend the definition further. In the aforementioned papers, it has been noted that they generalize intersection algebras in the case of principal ideals, an interesting class of algebras in its own right.

Definition 1.1. A fan $\mathcal{F}$ in $\mathbb{R}^m$ is a finite collection of pointed rational cones $\sigma \subset \mathbb{R}^m$ such that

(1) for all $\sigma \in \mathcal{F}$, each face of σ is also in $\mathcal{F}$.
(2) for all σ_1, σ_2 in $\mathcal{F}$, the intersection $\sigma_1 \cap \sigma_2$ is a face of each.

The support of $\mathcal{F}$ is the union of all σ in $\mathcal{F}$. We will denote this by $|\mathcal{F}|$. In our presentation, we make the additional assumption that the support $|\mathcal{F}|$ is also a convex cone.

Example 1.2. A simple example of a fan is $\{\sigma_1, \sigma_2, \tau_0, \tau_1, \tau_2\}$ where

$$\sigma_1 = \{(x, y) \in \mathbb{R}^2_{\geq 0} : 2x \geq 3y\},\ \sigma_2 = \{(x, y) \in \mathbb{R}^2_{\geq 0} : 2x \leq 3y\},$$

$$\tau_0 = \{(x, 0) \in \mathbb{R}^2_{\geq 0}\},\ \tau_1 = \{(x, y) \in \mathbb{R}^2_{\geq 0} : 2x = 3y\},\ \tau_2 = \{(0, y) \in \mathbb{R}^2_{\geq 0}\}.$$

Its support is $(\mathbb{R}_{\geq 0})^2$.

2010 *Mathematics Subject Classification.* Primary: 13A30 ; Secondary:05E40,13P10, 13P20 .
Cortadellas and D'Andrea are supported by the Spanish MEC research project MTM2013-40775-P.

Definition 1.3. Let $\mathcal{F}$ be a fan in $\mathbb{R}^m$. A function $f : |\mathcal{F}| \cap \mathbb{Z}^m \to \mathbb{N}$ is called fan linear if f is $\mathbb{N}$-linear on each face of $\mathcal{F}$ and subadditive on the support of the fan, i.e. $f(u+v) \leq f(u) + f(v)$ for all $u, v \in |\mathcal{F}| \cap \mathbb{Z}^m$.

Example 1.4. For the fan $\mathcal{F}$ from Example 1.2, let $f(x,y) = \max(2x, 3y)$, for all $(x,y) \in \mathbb{N}^2 = |\mathcal{F}| \cap \mathbb{Z}^2$. This obviously defines a fan linear map on $\mathcal{F}$.

Definition 1.5. Let $n, m \geq 1$ be integers, $\mathcal{F}$ a fan in $\mathbb{R}^m$, $\mathbf{f} = (f_1, \ldots, f_n)$ a collection of fan linear maps on $\mathcal{F}$, and $\mathbf{I} = (I_1, \ldots, I_n)$ ideals in a commutative ring R.

The *fan algebra* associated to the data $\mathcal{F}, \mathbf{f}, \mathbf{I}$ is defined as

$$\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}} := \sum_{\mathbf{v} \in |\mathcal{F}| \cap \mathbb{Z}^m} I_1^{f_1(\mathbf{v})} \ldots I_n^{f_n(\mathbf{v})} \mathbf{x}^{\mathbf{v}} \subset \mathrm{R}[x_1^{\pm 1}, x_2^{\pm 1}, \ldots, x_m^{\pm 1}], \tag{2}$$

where $x_1, \ldots, x_m$ are indeterminates and $\mathbf{x} = x_1 \ldots x_m$.

In the case I_j is the principal ideal $\langle p_j \rangle$, $j = 1, \ldots, n$, we have that $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}}$ is the fan algebra $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}}$ defined in (1).

Remark 1.6.

(1) If $|\mathcal{F}| = (\mathbb{R}_{\geq 0})^m$ and $f_i = 0$, $i = 1, \ldots, n$, it is easy to see that $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}} = \mathrm{R}[x_1, \ldots, x_m]$, the ring of polynomials in several variables. If instead we have that $|\mathcal{F}| = \mathbb{R}^m$ and the $f_i's$ identically zero as before, we obtain $\mathrm{R}[x_1^{\pm 1}, \ldots, x_m^{\pm 1}]$, the ring of Laurent polynomials in m variables.

(2) Let I and J be ideals of R. Their *intersection algebra* is defined as $B_{\mathrm{R}}(I,J) := \sum_{r,s \in \mathbb{N}} I^r \cap J^s x_1^r x_2^s \subset \mathrm{R}[x_1, x_2]$. Intersection algebras of principal ideals are fan algebras, as it was shown in [3].

(3) Set $m = n$, $|\mathcal{F}| = (\mathbb{R}_{\geq 0})^n$, and $\mathbf{I} = (I_1, \ldots, I_n)$ ideals in a commutative ring R and $f_i : (\mathbb{R}_{\geq 0})^n \cap \mathbb{Z}^n \to \mathbb{N}$ the projection on the ith coordinate, $i = 1, \ldots, n$. Then

$$\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}} = \sum_{(k_1,\ldots,k_n) \in \mathbb{N}^n} I_1^{k_1} \cdots I_n^{k_n} x_1^{k_1} \cdots x_n^{k_n}$$

is the multi-Rees algebra of the ideals $I_1, \ldots, I_n$ in R.

(4) Let I be an ideal in R. Then, for the fan and the fan linear map in Example 1.2 and, respectively, Example 1.4 we have

$$\mathcal{B}_{\mathbf{f},\mathcal{F},I} = \sum_{(r,s) \in \mathbb{N}^2} I^{\max(2r,3s)} x_1^r x_2^s.$$

Let $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}}$ be a fan algebra. This R-algebra is finitely generated, when R is Noetherian. Indeed, a set of generators of this algebra over R was given in [3, Theorem 2.3.7] (see also [2, 1]) for the case R being a domain and $|\mathcal{F}| = (\mathbb{R}_{\geq 0})^m$. It is easy to see that such a proof can be easily extended to the case of a fan $\mathcal{F}$ with the properties given above, and any Noetherian ring R, as we recall here: denote with $C_1, \ldots, C_\ell$ the maximal cones of $\mathcal{F}$, and for $i = 1, \ldots, \ell$, set $Q_i = C_i \cap \mathbb{Z}^m$. This *pointed monoid* has a unique Hilbert Basis, which we denote with:

$$\mathcal{H}_{Q_i} = \{\mathbf{v}_{i1}, \ldots, \mathbf{v}_{ik_i}\} \subset \mathbb{Z}^m. \tag{3}$$

As R is Noetherian, for each $\mathbf{v}_{ij}$, the ideal $I_1^{f_1(\mathbf{v}_{ij})} \cdot \ldots \cdot I_m^{f_m(\mathbf{v}_{ij})}$ is finitely generated, i.e.

$$I_1^{f_1(\mathbf{v}_{ij})} \cdot \ldots \cdot I_m^{f_m(\mathbf{v}_{ij})} = \langle r_{ij1}, \ldots, r_{ij\ell_{ij}} \rangle$$

The following result follows straightforwardly from Theorem 2.3.7 in [3].

Theorem 1.7. *With notations and assumptions as above, $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}}$ is generated as an algebra over* R *by the set*

$$\{r_{ij\ell}\, \mathbf{x}^{\mathbf{v}_{ij}};\ i = 0, \ldots, n,\ j = 1, \ldots, k_i,\ \ell = 1, \ldots, \ell_{ij}\}. \tag{4}$$

In general (4) is far from being a minimal set of generators of the fan algebra, although in some cases -like in the intersection algebra of principal ideals, see [2]- it has been shown to be the case. It is also of interest computing a whole resolution of $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{I}}$ as an R-algebra, but very little seems to be known at the present about this ring in general. Our main result is a full description of the case $m = 2$, with all the ideals being principal, i.e. when $I_j = \langle p_j \rangle$ for a suitable $p_j \in$ R, and $|\mathcal{F}|$ strictly contained in a half-plane. This situation includes the intersection algebras of principal ideals, hence we generalize and extend the results in [2]. Our main result given in Theorem 2.1 describes a resolution of this R-algebra. If R is $*$-local, under some mild conditions on the functions $\{f_i\}_{1 \le i \le n}$ and the elements $\{p_i\}_{1 \le i \le n}$, we show that the resolution is minimal. If in addition R is a local ring and all the p_i's are nonzero divisors in the maximal ideal, then we can describe combinatorially the $\mathbb{Z}^2$-graded Betti numbers in terms of the elements of $\mathcal{H}$.

2. Statement of the main result

From now on, we will work with principal ideals, and fans $\mathcal{F}$ of cones in $\mathbb{R}^2$ such that $|\mathcal{F}|$ is also convex, and contained in a half-plane. In $\mathbb{R}^2$ we have the advantage that there is a standard orientation for both cones and vectors which we will be use. So assume w.l.o.g. that $C_1, \ldots, C_\ell$, the maximal cones in $\mathcal{F}$, are sorted clockwise. In addition, we will also assume that for each $i = 1, \ldots, \ell$, the elements in the set $\mathcal{H}_{Q_i} = \{\mathbf{v}_{i1}, \ldots, \mathbf{v}_{ik_i}\}$ which was defined in (3), are also sorted clockwise.

We will say that the family of functions $\mathbf{f}$ is *strict* with respect to the fan $\mathcal{F}$ if for each $1 \le i < \ell$, there is $k \in \{1, \ldots, n\}$ such that f_k is not linear in $C_i \cup C_{i+1}$. Note that if $\mathbf{f}$ is not strict with respect to $\mathcal{F}$ one can take a coarser fan $\mathcal{F}'$ such that $\mathbf{f}$ is also a family of piece-wise linear functions compatible with $\mathcal{F}'$ and strict with respect to this new fan, and moreover from the definition we get

$$\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}} = \mathcal{B}_{\mathbf{f},\mathcal{F}',\mathbf{p}},$$

so we can always assume w.l.o.g. that the set of functions is strict with respect to the fan.

Set $\mathcal{H} := \cup_{i=0}^{\ell} \mathcal{H}_{Q_i}$, and denote with M the cardinality of this set. Let $p_1, \ldots, p_n \in$ R. In the sequel, we will set $\mathbf{p}^{\mathbf{f}(\mathbf{v})} = p_1^{f_1(\mathbf{v})} \ldots p_n^{f_n(\mathbf{v})}$, for short. For each $\mathbf{v} \in \mathcal{H}$, let $Y_{\mathbf{v}}$ be a new variable, and set $\mathbf{Y} = \{Y_{\mathbf{v}},\ \mathbf{v} \in \mathcal{H}\}$. In light of Theorem 1.7, we have an epimorphism

$$\begin{array}{ccc} \mathrm{R}[\mathbf{Y}] & \stackrel{\varphi_0}{\to} & \mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}} \\ Y_{\mathbf{v}} & \mapsto & \mathbf{p}^{\mathbf{f}(\mathbf{v})}\mathbf{x}^{\mathbf{v}} \end{array} \tag{5}$$

Note that φ_0 is a $\mathbb{Z}^2$-graded map if we declare $\deg(\mathbf{Y}_{\mathbf{v}}) = \mathbf{v}$, and $\deg(r) = (0,0)$ for $r \in$ R.

The following is our main result.

Theorem 2.1. *If $|\mathcal{F}|$ is a pointed rational cone in $\mathbb{R}^2$, then the following is a $\mathbb{Z}^2$-graded resolution of $\mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}}$ over $\mathrm{R}[\mathbf{Y}]$:*

(6)

$$0 \to \mathrm{R}[\mathbf{Y}]^{M-2} \xrightarrow{\varphi_{M-2}} \ldots \to \mathrm{R}[\mathbf{Y}]^{3\binom{M-1}{4}} \xrightarrow{\varphi_3} \mathrm{R}[\mathbf{Y}]^{2\binom{M-1}{3}} \xrightarrow{\varphi_2} \mathrm{R}[\mathbf{Y}]^{\binom{M-1}{2}} \xrightarrow{\varphi_1} \mathrm{R}[\mathbf{Y}] \xrightarrow{\varphi_0} \mathcal{B}_{\mathbf{f},\mathcal{F},\mathbf{p}} \to 0,$$

*If R is *-local with respect to the $\mathbb{Z}^2$-grading, then the resolution if minimal if $\mathbf{f}$ is strict with respect to $\mathcal{F}$, none of the p_i's is a zero divisor in R, and all of them belong to the unique homogeneous maximal ideal of R.*

The maps $\varphi_j : \mathrm{R}[\mathbf{Y}]^{j\binom{M-1}{j+1}} \to \mathrm{R}[\mathbf{Y}]^{(j-1)\binom{M-1}{j}}$, $j = 1, 2, \ldots$ can be constructed explicitly from the combinatorics of the fan. We omit their presentation due to lack of space.

References

[1] Enescu, Florian; Malec, Sara. Intersection algebras for principal monomial ideals in polynomial rings. J. Algebra Appl. 14 (2015), no. 7, 1550108, 23 pp.

[2] Malec, Sara. *Intersection Algebras and pointed rational cones.* Ph.D. Thesis, 2013.

[3] Malec, Sara. *On the Intersection Algebra of Principal Ideals.* Comm. Algebra 43 (2015), no. 2, 623–635.

Universitat de Barcelona, Facultat de Formació del Professorat. Passeig de la Vall d'Hebron 171, 08035 Barcelona, Spain

E-mail address: terecortadellas@ub.edu

Universitat de Barcelona, Facultat de Matemàtiques. Gran Via 585, 08007 Barcelona, Spain

E-mail address: cdandrea@ub.edu

URL: http://atlas.mat.ub.es/personals/dandrea

Department of Mathematics and Statistics, Georgia State University, Atlanta, GA 30303 USA

E-mail address: fenescu@gsu.edu

THE FORMALISM OF RATIONAL INTERPOLATION

TERESA CORTADELLAS, CARLOS D'ANDREA, AND EULÀLIA MONTORO

Abstract. Rational Interpolation can be regarded as a generalization of the classical Lagrange Polynomial Interpolation, and its algorithmic approach as well as algebraic has a lot of interest from both a geometric and computational point of view. In contrast with its polynomial counterpart, there do not always exist a solution of this problem for any initial data. The purpose of our research is to detect and classify the set of "bad points" for the Rational Interpolation Problem both geometrically and algebraically. We present here some algebraic tools which may be useful for the resolution of this problem.

1. Symbolic Rational Interpolation

Let $\mathbb{K}$ be any field, $k, n \in \mathbb{N}$, $1 \leq k \leq n$, $u_1, \ldots, u_n$, $v_1, \ldots, v_n \in \mathbb{K}$, with $u_i \neq u_j$ if $i \neq j$. Set $\mathbf{u} = (u_1, \ldots, u_n)$, and $\mathbf{v} = (v_1, \ldots, v_n)$. The Rational Interpolation Problem (RIP) is the following: decide whether there exist (and if so, compute them) polynomials $N_{\mathbf{u},\mathbf{v},k-1}(x)$, $D_{\mathbf{u},\mathbf{v},n-k}(x) \in \mathbb{K}[x]$ of degree bounded by $k-1$ and $n-k$ respectively such that $D_{\mathbf{u},\mathbf{v},n-k}(u_i) \neq 0$ for all $i = 1, \ldots, n$ and

$$\frac{N_{\mathbf{u},\mathbf{v},k-1}(u_i)}{D_{\mathbf{u},\mathbf{v},n-k}(u_i)} = v_i,\ i = 1, \ldots, n. \tag{1}$$

In a more general setting, let $U_1, \ldots, U_n$, $V_1, \ldots, V_n$ be indeterminates over $\mathbb{K}$. We will denote with $\mathbf{U}$ the list of elements $U_1, \ldots, U_n$ and with $\mathbf{V}$ the respective list with V_i, $i = 1, \ldots, n$. The Symbolic Rational Interpolation Problem (SRIP) states the following: for a given $k \in \{1, \ldots, n\}$, find polynomials $N_{k-1}(\mathbf{U}, \mathbf{V}, x)$, $D_{n-k}(\mathbf{U}, \mathbf{V}, x) \in \mathbb{K}[\mathbf{U}, \mathbf{V}, x]$ of degrees in x bounded by $k-1$ and $n-k$ respectively such that $D_{n-k}(\mathbf{U}, \mathbf{V}, U_i) \neq 0$ for all $i = 1, \ldots, n$, and

$$\frac{N_{k-1}(\mathbf{U}, \mathbf{V}, U_i)}{D_{n-k}(\mathbf{U}, \mathbf{V}, U_i)} = V_i,\ i = 1, \ldots, n. \tag{2}$$

Rational Interpolation can be regarded as a generalization of the classical Lagrange Polynomial Interpolation, and its algorithmic and algebraic approach has been treated extensively, see for instance [3, 2] and the references therein. In contrast with its polynomial counterpart, there is not always a solution of this problem for any initial data $(\mathbf{u}, \mathbf{v})$.

In principle, it is clear that if $D_{n-k}(\mathbf{u}, \mathbf{v}, u_i) \neq 0$ for all $i = 1, \ldots, n$, then one can take $N_{\mathbf{u},\mathbf{v},k-1}(x) = N_{k-1}(\mathbf{u}, \mathbf{v}, x)$ and $D_{\mathbf{u},\mathbf{v},n-k}(x) = D_{n-k}(\mathbf{u}, \mathbf{v}, x)$, the specialization of the polynomials in (2) after setting $\mathbf{U} \mapsto \mathbf{u}$ and $\mathbf{V} \mapsto \mathbf{v}$, and (1) is just a specialization of (2). What is interesting is what happens when $D_{n-k}(\mathbf{u}, \mathbf{v}, u_i) = 0$ for some $i \in \{1, \ldots, n\}$, as there still must be a solution of (1) which is not a specialization (or a "limit") of (2). The purpose of our research is to understand from both and algebraic and geometric point of

view this phenomena, and also to characterize the set of "bad points", i.e. the set of all the $(\mathbf{u}, \mathbf{v})$ such that the RIP for a given k does not have solutions.

We start by dealing with (2). In order to do that, recall some results given in [2]: for two sequences X, Y, we define $R(X,Y) := \prod_{x\in X,\, y\in Y}(x-y)$, and $V(X) = \prod_{x,\, x'\in X,\, x<x'}(x-x')$, where $x < x'$ means that the element x appears before x' in the sequence X. We set

(3)
$$\begin{aligned} N_{k-1}(\mathbf{U},\mathbf{V},x) &= \textstyle\sum_{\mathbf{U}'\subset\mathbf{U},\,|\mathbf{U}'|=k-1}(-1)^{\sigma'}R(x,\mathbf{U}')V(\mathbf{U}')V(\mathbf{U}\setminus\mathbf{U}')\prod_{U_j\notin\mathbf{U}'}V_j,\\ D_{n-k}(\mathbf{U},\mathbf{V},x) &= (-1)^{(n-k)(k-1)}\textstyle\sum_{\mathbf{U}''\subset\mathbf{U},\,|\mathbf{U}''|=n-k}(-1)^{\sigma''}R(x,\mathbf{U}'')V(\mathbf{U}'')V(\mathbf{U}\setminus\mathbf{U}'')\prod_{U_j\in\mathbf{U}''}V_j. \end{aligned}$$

Where, σ' and σ'' are, respectively, the number of transpositions needed to take the list U to: $\{\mathbf{U}\setminus\mathbf{U}'\}\sqcup\{\mathbf{U}'\}$ and $\{\mathbf{U}\setminus\mathbf{U}''\}\sqcup\{\mathbf{U}''\}$.

Proposition 1.1. *For $k\in\{1,\dots,n\}$, the polynomials $N_{k-1}(\mathbf{U},\mathbf{V},x)$, $D_{n-k}(\mathbf{U},\mathbf{V},x)$ defined in* (3) *are solutions of* (2).

Remark 1.2. If we divide each of these two polynomials by $V(\mathbf{U})$, we obtain the expressions denoted with A_0 and B_0 in the statement of [2, Theorem 3.1], Proposition 1.1 follows from that claim.

Write now

$$\begin{aligned} N_{k-1}(\mathbf{U},\mathbf{V},x) &= A_{k,0}(\mathbf{U},\mathbf{V}) + A_{k,1}(\mathbf{U},\mathbf{V})x + \ldots + A_{k,k-1}(\mathbf{U},\mathbf{V})x^{k-1},\\ D_{n-k}(\mathbf{U},\mathbf{V},x) &= B_{k,0}(\mathbf{U},\mathbf{V}) + B_{k,1}(\mathbf{U},\mathbf{V})x + \ldots + B_{k,n-k}(\mathbf{U},\mathbf{V})x^{n-k}. \end{aligned}$$

From (3), we get easily that for $j = 0,\dots,k-1$,

$$A_{k,j}(\mathbf{U},\mathbf{V}) = (-1)^{k-1-j}\sum_{\mathbf{U}'\subset\mathbf{U},\,|\mathbf{U}'|=k-1}(-1)^{\sigma'}V(\mathbf{U}')V(\mathbf{U}\setminus\mathbf{U}')\prod_{U_i\notin\mathbf{U}'}V_i\left(\sum_{\mathbf{U}_0'\subset\mathbf{U}',\,|\mathbf{U}_0'|=k-1-j}\ \prod_{U\in\mathbf{U}_0'}U\right), \tag{4}$$

and for $j = 0,\dots,n-k$,

$$B_{k,j}(\mathbf{U},\mathbf{V}) = (-1)^{(n-k)(k-1)}\sum_{\mathbf{U}''\subset\mathbf{U},\,|\mathbf{U}''|=n-k}(-1)^{\sigma''}V(\mathbf{U}'')V(\mathbf{U}\setminus\mathbf{U}'')\prod_{U_i\in\mathbf{U}''}V_i\left(\sum_{\mathbf{U}_0''\subset\mathbf{U}'',\,|\mathbf{U}_0''|=n-k-j}\ \prod_{U\in\mathbf{U}_0''}U\right). \tag{5}$$

2. Weak Rational Interpolation

The Symbolic Weak Rational Interpolation Problem (SWRIP) states the following: for a given $k\in\{1,\dots,n\}$, find polynomials $N_{k-1}(\mathbf{U},\mathbf{V},x)$, $D_{n-k}(\mathbf{U},\mathbf{V},x)\in\mathbb{K}[\mathbf{U},\,\mathbf{V},\,x]$ of degrees in x bounded by $k-1$ and $n-k$ respectively, such that $D_{n-k}(\mathbf{U},\mathbf{V},U_i)\neq 0$ for all $i=1,\dots,n$, and

$$N_{k-1}(\mathbf{U},\mathbf{V},U_i) = V_i\,D_{n-k}(\mathbf{U},\mathbf{V},U_i),\ i=1,\dots,n. \tag{6}$$

The Weak Rational Interpolation Problem (WRIP) asks for $\mathbf{u},\,\mathbf{v}\in\mathbb{K}^n$, with $u_i\neq u_j$ if $i\neq j$, to decide whether there exist (and if so, compute them) polynomials $N_{\mathbf{u},\mathbf{v},k-1}(x)$, $D_{\mathbf{u},\mathbf{v},n-k}(x)\in\mathbb{K}[x]$ of degrees bounded by $k-1$ and $n-k$ respectively such that

$$N_{\mathbf{u},\mathbf{v},k-1}(u_i) = v_i\,D_{\mathbf{u},\mathbf{v},n-k}(u_i),\ i=1,\dots,n. \tag{7}$$

Note that both (6) and (7) give a system of n homogeneous linear equations in $n+1$ unknowns (the coefficients of the polynomials $N_{k-1}(\mathbf{U},\mathbf{V},x)$, $D_{n-k}(\mathbf{U},\mathbf{V},x)$ and $N_{\mathbf{u},\mathbf{v},k-1}(x)$, $D_{\mathbf{u},\mathbf{v},n-k}(x)$ respectively), so there is always a nontrivial solution to both problems. In addition, it is clear that any solution of (2) (resp. (1)) is a solution of (6) (resp. (7)). The following claim follows straightforwardly.

Proposition 2.1. *Up to a constant in $\mathbb{K}(\mathbf{U},\mathbf{V})$, there is a unique solution to* (6) *which is given by $N_{k-1}(\mathbf{U},\mathbf{V},x)$, $D_{n-k}(\mathbf{U},\mathbf{V},x)$ defined in* (3).

Let $\mathbb{M}_k(\mathbf{U},\mathbf{V}) \in \mathbb{K}[\mathbf{U},\mathbf{V}]^{n\times(n+1)}$ (resp. $\mathbb{M}_{k,\mathbf{u},\mathbf{v}} \in \mathbb{K}^{n\times(n+1)}$) be the matrix of the linear system (6) (resp. (7)). Computing it explicitly, we have

$$\mathbb{M}_k(\mathbf{U},\mathbf{V}) = \begin{pmatrix} 1 & U_1 & U_1^2 & \dots & U_1^{k-1} & -V_1 & -V_1U_1 & \dots & -V_1U_1^{n-k} \\ 1 & U_2 & U_2^2 & \dots & U_2^{k-1} & -V_2 & -V_2U_2 & \dots & -V_2U_2^{n-k} \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & U_n & U_n^2 & \dots & U_n^{k-1} & -V_n & -V_nU_n & \dots & -V_nU_n^{n-k} \end{pmatrix},$$

and clearly we have $\mathbb{M}_{k,\mathbf{u},\mathbf{v}} = \mathbb{M}_k(\mathbf{u},\mathbf{v})$.

For $j = 1,\dots,n+1$, denote with $\mathcal{M}_j(\mathbf{U},\mathbf{V})$ the signed maximal minor obtained from removing the j-th column in $\mathbb{M}_k(\mathbf{U},\mathbf{V})$. We then have

Proposition 2.2. *Up to a constant in $\mathbb{K}(\mathbf{U},\mathbf{V})$, we have that $\mathcal{M}_j(\mathbf{U},\mathbf{V}) = A_{k,j-1}(\mathbf{U},\mathbf{V})$ for $j \in \{1,\dots,k\}$, and $\mathcal{M}_j(\mathbf{U},\mathbf{V}) = B_{k,j-k-1}(\mathbf{U},\mathbf{V})$ for $j \in \{k+1,n+1\}$, where $A_{k,j-1}(\mathbf{U},\mathbf{V})$ (resp. $B_{k,j-k-1}(\mathbf{U},\mathbf{V})$) has been defined in* (4) *(resp.* (5)*).*

Corollary 2.3. *Up to a sign, we have*

$$N_{k-1}(\mathbf{U},\mathbf{V},x) = \begin{vmatrix} \mathbb{M}_k(\mathbf{U},\mathbf{V}) \\ 1\, x \dots x^{k-1}\, 0 \dots 0 \end{vmatrix}, \text{ and } D_{n-k}(\mathbf{U},\mathbf{V},x) = \begin{vmatrix} \mathbb{M}_k(\mathbf{U},\mathbf{V}) \\ 0 \dots 0\, 1\, x \dots x^{n-k} \end{vmatrix}.$$

3. Specialized Rational Interpolation

Denote with $\Delta \subset \mathbb{K}^n$ the variety defined by $\{(u_1,\dots,u_n) \in \mathbb{K}^n : \prod_{1\le i<j\le n}(u_i-u_j) = 0\}$. Our input data $\mathbf{u}$ will be taken from $\mathbb{K}^n \setminus \Delta$. For a given $\mathbf{v} \in \mathbb{K}^n$, note that we can assume without loss of generality that $v_i \neq 0$ for all $i = 1,\dots,n$, as otherwise a factor of the form $(x-u_i)$ can be removed from the numerator $N_{\mathbf{u},\mathbf{v},k-1}(x)$, and after that we can consider an analogous problem with $n-1$ instead of n and $k-1$ instead of k.

First, will focus now in the resolution of the WRIP (7). For $n,k \in \mathbb{N}$, $1 \le k \le n$, and $(\mathbf{u},\mathbf{v}) \in (\mathbb{K}^n \setminus \Delta) \times (\mathbb{K}^\times)^n$, denote with $\mathbb{V}_{\mathbf{u},\mathbf{v},k} \subset \mathbb{K}[x]_{k-1} \oplus \mathbb{K}[x]_{n-k}$ the set defined as

$$\mathbb{V}_{\mathbf{u},\mathbf{v},k} = \{(N_{k-1}(x), D_{n-k}(x)) : N_{k-1}(u_i) = v_i D_{n-k}(u_i)\, \forall i = 1,\dots,n\}.$$

Theorem 3.1. *For a given $(\mathbf{u},\mathbf{v}) \in (\mathbb{K}^n \setminus \Delta) \times (\mathbb{K}^\times)^n$, let n_0 (resp. d_0) denote the degree of $N^0_{k-1}(x)$ (resp. $D^0_{n-k}(x)$) for a minimal element $(N^0_{k-1}(x), D^0_{n-k}(x)) \in \mathbb{V}_{\mathbf{u},\mathbf{v},k}$. Then*

- $\mathbb{V}_{\mathbf{u},\mathbf{v},k} = (N^0_{k-1}(x), D^0_{n-k}(x)) \cdot \mathbb{K}[x]_{s_0}$, *with* $s_0 := \min\{k-1-n_0,\, n-k-d_0\}$.
- $\dim(\mathbb{V}_{\mathbf{u},\mathbf{v},k}) = s_0 + 1$.
- *The RIP* (1) *has a solution if and only if* $\gcd(N^0_{k-1}(x), D^0_{n-k}(x)) = 1$.

Thanks to Theorem 3.1 we can produce the following algorithm.

Algorithm 3.2.
Input: $(\mathbf{u},\mathbf{v}) \in \big(\mathbb{K}^n \setminus \Delta\big) \times (\mathbb{K}^\times)^n$
Output: A reduced solution of the RIP associated to $(\mathbf{u},\mathbf{v})$, or a message that it does not have a solution.

(1) Compute a nontrivial element of the kernel of the matrix $\mathbb{M}_{\mathbf{u},\mathbf{v}}$ associated to the linear system (7).
(2) Extract polynomials $N_{k-1}(x)$ and $D_{n-k}(x)$ from the coordinates of the element computed in (1).
(3) Remove the common factors of these two polynomials. Denote the reduced polynomials by $N^0_{k-1}(x)$ and $D^0_{n-k}(x)$.
(4) if $(N^0_{k-1}(x), D^0_{n-k}(x)) \in \ker(\mathbb{M}_{\mathbf{u},\mathbf{v}})$, then return $(N^0_{k-1}(x), D^0_{n-k}(x))$, otherwise return "no solution".

4. Towards a characterization of bad points

From a more algebraic point of view, denote with $R = \mathbb{K}[\mathbf{U}, \mathbf{V}]$ the ring of polynomials in the $2n$ variables $U_1, \ldots, U_n, V_1, \ldots, V_n$, and let $I \subset R$ be the ideal generated by the maximal minors of $\mathbb{M}_k(\mathbf{U}, \mathbf{V})$. We have the following complex of R-modules:

$$0 \to R^n \xrightarrow{\phi_{\mathbb{M}}} R^{n+1} \xrightarrow{\mathcal{M}} R \to R/I \to 0, \tag{8}$$

with $\phi_{\mathbb{M}}$ being the map whose matrix in the canonical bases of R^n and R^{n+1} is the transpose of $\mathbb{M}_k(\mathbf{U}, \mathbf{V})$ and the matrix of $\mathcal{M}$ is defined by the maximal minors of the transpose of $\mathbb{M}_k(\mathbf{U}, \mathbf{V})$.

Theorem 4.1. *The complex* (8) *is exact, and I is a perfect ideal of degree* 2. *Moreover, the set* $\{A_{k,k-1}(\mathbf{U}, \mathbf{V}), B_{k,n-k}(\mathbf{U}, \mathbf{V})\}$ *is a maximal regular sequence in I generating* $\sqrt{I}$.

This result will be of use in our future work to detect and classify the set of "bad points" for the Rational Interpolation both from a geometric as well as algebraic point of view.

References

[1] Bruns, Winfried; Herzog, H. Jürgen. *Cohen-Macaulay rings.* Cambridge University Press; 2 edition 1998.

[2] D'Andrea, Carlos; Krick, Teresa; Szanto, Agnes. *Subresultants, Sylvester sums and the rational interpolation problem.* J. Symbolic Comput. 68 (2015) 72–83.

[3] von zur Gathen, Joachim; Gerhard, Jürgen. *Modern computer algebra.* Third edition. Cambridge University Press, Cambridge, 2013.

Universitat de Barcelona, Facultat de Formació del Professorat. Passeig de la Vall d'Hebron 171, 08035 Barcelona, Spain

E-mail address: terecortadellas@ub.edu

Universitat de Barcelona, Facultat de Matemàtiques. Gran Via 585, 08007 Barcelona, Spain

E-mail address: cdandrea@ub.edu

URL: http://atlas.mat.ub.es/personals/dandrea

Universitat de Barcelona, Facultat de Matemàtiques. Gran Via 585, 08007 Barcelona, Spain

E-mail address: eula.montoro@ub.edu

A PSEUDO-MATRIX APPROACH TO PRÜFER DOMAINS

GEMA M. DÍAZ-TOCA AND HENRI LOMBARDI

Abstract. In this extended abstract, we present the tools in order to construct an algorithm for computing the Hermite normal form of pseudo-matrices over Prüfer domains. This algorithm allows us to provide constructive proofs of the main theoretical results on finitely presented modules over Prüfer domains and to discuss the resolution of linear systems. We generalize the methodology developed by Henri Cohen for Dedekind domains in [3, Chapter 1]. Finally, we present some results for Prüfer domains of dimension one. A full paper is found on `http://arxiv.org/abs/1508.00345`.

Introduction

The algorithmic solution of linear systems over fields or over PIDs is classical and it is equivalent to transforming the system via elementary manipulations (and Bezout manipulations for PIDs), in order to obtain a convenient reduced form (Hermite normal form or Smith normal form).

We use here a generalization of this kind of process for arbitrary Prüfer domains.

We adapt for an arbitrary Prüfer domain the generalized matrix computations given by Henri Cohen [3, Chapter 1] for the algorithmics in rings of number fields (number rings).

We obtain a system of generalized matrix computations and as consequences the main "abstract" theorems for Prüfer domains. The generalization consists in replacing when necessary matrices over usual bases by matrices over decompositions of the modules as direct sums of rank one projective modules. These new matrices are called pseudo-matrices.

From a Computer Algebra viewpoint, computing with pseudo-matrices allows us to treat some examples inaccessible for usual methods: since our true computational tool is the inversion of finitely generated ideals, it is possible to work with number rings whose discriminant has no known complete factorization.

For Dedekind domains, and more generally for dimension one Prüfer domains, we obtain more precise results, similar to Smith reduction of usual matrices in PIDs.

General references for the constructive theory of Prüfer domains are found in [1, 2, 4]. Many useful constructive proofs are also found in [5].

1. Basic facts

1.1. Definitions.

A ring $\mathbf{A}$ is ***zero-dimensional*** when

$$\forall a \in \mathbf{A},\ \exists n \in \mathbb{N}\ \exists x \in \mathbf{A},\ x^n(1-ax)=0.$$

An integral domain $\mathbf{A}$ is ***of (Krull) dimension*** $\leqslant 1$ if for all $b \neq 0$ in $\mathbf{A}$, the quotient ring $\mathbf{A}/\langle b\rangle$ est zero-dimensional. E.g. number rings have dimension 1 because their quotients are finite, and consequently zero-dimensional.

Over an arbitrary ring $\mathbf{A}$ a finitely generated ideal $\mathfrak{a} = \langle a_1, \ldots, a_n\rangle$ is ***locally principal*** if there exists $s_1, \ldots, s_n \in \mathbf{A}$ such that $\sum_{i\in[\![1..n]\!]} s_i = 1$ and $s_i\mathfrak{a} \subseteq \langle a_i\rangle$ for each s_i.

A ring $\mathbf{A}$ is ***arithmetic*** if all finitely generated ideals are locally principal.

A finitely generated ideal $\mathfrak{a} = \langle a_1, \ldots, a_n\rangle$ is ***invertible*** if there exists a regular element c and an finitely generated ideal $\mathfrak{b}$ such that $\mathfrak{ab} = \langle c\rangle$. Il other words, $\mathfrak{a}$ is locally principal an contains a regular element.

A ***Prüfer domain*** is an integral arithmetic ring. In other words, it is an integral domain whose all nonzero finitely generated ideal are invertible.

The ***determinantal ideal of order*** k of a matrix M is the ideal $\mathfrak{D}_k(M)$ generated by the minors of order k of M.

The ***Fitting ideal of order*** k of a finitely presented module P, coker of a matrix $M \in \mathbb{M}_{n,m}(\mathbf{A})$ is defined by $\mathfrak{F}_k(P) := \mathfrak{D}_{n-k}(M)$.

1.2. Computations with finitely generated ideals in a Prüfer domain.

We work with an explicit Prüfer domain $\mathbf{Z}$. This means that for an arbitrary finitely generated ideal $\mathfrak{a} = \langle a_1, \ldots, a_n\rangle$ we have an algorithm that computes $s_1, \ldots, s_n \in \mathbf{Z}$ such that $\sum_i s_i = 1$ and $s_i\mathfrak{a} \subseteq \langle a_i\rangle$ for each s_i. E.g. number rings are explicit Prüfer domains. We assume also that $\mathbf{Z}$ has a divisbility test, giving an x s.t. $ax = b$ when the test gives the answer "Yes" to the question "does a divise b?".

From these basic algorithms the following computations are shown to be easy. Note that by "computing an ideal", we mean to compute a generator set and a list $(s_1, \ldots, s_n)$ as in the previous explanation.

- For $\mathfrak{a}$ finitely generated, compute an ideal $\mathfrak{b}$ s.t. $\mathfrak{ab}$ is principal.
- For $\mathfrak{a}$ and $\mathfrak{b}$ finitely generated, compute s, t s.t. $s + t = 1$, $s\mathfrak{a} \subseteq \mathfrak{b}$ and $t\mathfrak{b} \subseteq \mathfrak{a}$.
- For $\mathfrak{a}$ and $\mathfrak{b}$ finitely generated, compute $\mathfrak{a} + \mathfrak{b}$, $\mathfrak{ab}$, $\mathfrak{a} \cap \mathfrak{b}$ and $(\mathfrak{a} : \mathfrak{b})$.
- For $\mathfrak{a}$ and $\mathfrak{b}$ finitely generated, test if $\mathfrak{a} \subseteq \mathfrak{b}$.

The following computations are more tricky. We assume that $\mathbf{Z}$ is moreover explicitly of dimension 1.

- For $\mathfrak{a}$ finitely generated and a nonzero in $\mathfrak{a}$, compute $b \in \mathfrak{a}$ s.t. $\mathfrak{a} = \langle a, b\rangle$.
- For $\mathfrak{a}$ and $\mathfrak{b}$ finitely generated, compute an isomorphism between the modules $\mathfrak{a} \oplus \mathfrak{b}$ and $\mathbf{Z} \oplus \mathfrak{ab}$.

2. Pseudo-bases and pseudo-matrices

We note $\mathbf{K}$ the quotient field of $\mathbf{Z}$ and $\mathrm{Gfr}(\mathbf{Z})$ the (multiplicative) group of ***fractional ideals*** of $\mathbf{K}$. Such a fractional ideal is a sub-$\mathbf{Z}$-module of $\mathbf{K}$ equal to $\frac{\mathfrak{a}}{c}$ for a (usual) finitely generated ideal $\mathfrak{a} \subseteq \mathbf{Z}$ and c nonzero in $\mathbf{Z}$. A $\mathbf{Z}$-module E which is finitely generated and without torsion can be viewed as a sub-$\mathbf{Z}$-module of the $\mathbf{K}$-vector space $E' = \mathbf{K} \otimes_{\mathbf{Z}} E$.

A finitely generated projective $\mathbf{Z}$-module E can always be given as a direct sum $E = E_1 \oplus \cdots \oplus E_r$ with isomorphisms $E_i \simeq \mathfrak{e}_i \in \mathrm{Gfr}(\mathbf{A})$: $\mathfrak{e}_i \ni x \mapsto xe_i$ (where $e_i \in E'$). A ***pseudo-basis*** of E is by definition an r-tuple

$$\boxed{\big((e_1, \mathfrak{e}_1), \ldots, (e_r, \mathfrak{e}_r)\big)} \text{ s.t. } E = \mathfrak{e}_1e_1 \oplus \cdots \oplus \mathfrak{e}_re_r,$$

Note that $(e_1, \dots, e_r)$ is a basis of the vector space E'.

Let $\varphi : E \to H$ a linear map between projective modules with pseudo-bases

$$\mathscr{E} = \big((e_1, \mathfrak{e}_1), \dots, (e_m, \mathfrak{e}_m)\big) \text{ and } \mathscr{H} = \big((h_1, \mathfrak{h}_1), \dots, (h_n, \mathfrak{h}_n)\big).$$

Extending the scalars to $\mathbf{K}$ we get a linear map $\varphi' : E' \to H'$ with a matrix $\underline{A}$ over the $\mathbf{K}$-bases $(e_1, \dots, e_m)$ and $(h_1, \dots, h_n)$.

- We call ***matrix of φ over pseudo-bases $\mathcal{E}$ and $\mathcal{H}$*** the data

  $$A = (\mathfrak{h}_1, \dots, \mathfrak{h}_n; \mathfrak{e}_1, \dots, \mathfrak{e}_m; \underline{A}) = (\underline{\mathfrak{h}}; \underline{\mathfrak{e}}; \underline{A}), \text{ where } \underline{A} = (a_{ij})_{ij} \in \mathbb{M}_{n,m}(\mathbf{K}).$$

  We have the inclusions $a_{ij}\mathfrak{e}_j \subseteq \mathfrak{h}_i$. We note $\boxed{A = \mathscr{M}_{\mathcal{E},\mathcal{H}}(\varphi)}$.

intuitive visualization:

$$A = \begin{array}{c} \\ \mathfrak{h}_1 \\ \mathfrak{h}_2 \\ \mathfrak{h}_3 \end{array} \begin{array}{c} \begin{array}{cccc} \mathfrak{e}_1 & \mathfrak{e}_2 & \mathfrak{e}_3 & \mathfrak{e}_4 \end{array} \\ \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix} \end{array} \quad a_{ij}\mathfrak{e}_j \subseteq \mathfrak{h}_i.$$

- We call ***pseudo-matrix*** any data $(\underline{\mathfrak{h}}; \underline{\mathfrak{e}}; \underline{A})$ of this kind, i.e. with inclusions $a_{ij}\mathfrak{e}_j \subseteq \mathfrak{h}_i$. It can be viewed as the matrix of a $\mathbf{Z}$-linear map between sub-$\mathbf{Z}$-modules of $\mathbf{K}^n$ and $\mathbf{K}^m$.
- For fixed lists $\underline{\mathfrak{e}}$ et $\underline{\mathfrak{h}}$, the corresponding pseudo-matrices define a $\mathbf{Z}$-module $\boxed{\mathbb{M}_{\underline{\mathfrak{h}};\underline{\mathfrak{e}}}(\mathbf{A})}$ (isomorphic to the $\mathbf{Z}$-module of $\mathbf{Z}$-linear maps from E to H).
  The product of pseudo-matrices of convenient formats is defined in the natural way and corresponds to the composition of linear maps.
- For a square pseudo-matrix $A = (\underline{\mathfrak{h}}; \underline{\mathfrak{e}}; \underline{A})$ we define its ***determinant (ideal)*** as being

  $$\boxed{\mathbf{Z} \supseteq \mathfrak{Det}(A) := \det(\underline{A})\, \mathfrak{e}\, \mathfrak{h}^{-1}}, \text{ where } \mathfrak{e} = \prod\nolimits_j \mathfrak{e}_j \text{ and } \mathfrak{h} = \prod\nolimits_i \mathfrak{h}_i.$$

  A square pseudo-matrix A is invertible if and only if $\mathfrak{Det}(A) = \mathbf{Z}$. For square pseudo-matrices A and B with convenient formats we have $\mathfrak{Det}(AB) = \mathfrak{Det}(A)\,\mathfrak{Det}(B)$.
- Let $\beta = [\beta_1, \dots, \beta_r] \subseteq [\![1..n]\!]$ et $\alpha = [\alpha_1, \dots, \alpha_r] \subseteq [\![1..m]\!]$ subsequences in increasing order. We note $A_{\beta,\alpha}$ the pseudo-matrix extracted on the rows β and columns α.

  $$A_{\beta,\alpha} = (\mathfrak{h}_{\beta_1}, \dots, \mathfrak{h}_{\beta_r}; \mathfrak{e}_{\alpha_1}, \dots, \mathfrak{e}_{\alpha_r}, \underline{A}_{\beta,\alpha}).$$

  The ideal

  $$\mathfrak{m}_{\beta,\alpha}(A) := \mathfrak{Det}(A_{\beta,\alpha}) = \det(\underline{A}_{\beta,\alpha})(\textstyle\prod_{i=0}^{r} \mathfrak{e}_{\alpha_i})(\prod_{j=0}^{r} \mathfrak{h}_{\beta_j})^{-1}$$

  is called ***the minor (ideal) of order r of A extracted on rows β and columns α.***
- For an arbitrary pseudo-matrix and $r \leqslant \inf(m, n)$ the ***determinantal ideal of order r of*** A, noted $\mathfrak{D}_r(A)$, is the sum of minors of order r of A.
  The pseudo-matrix A represents a surjective linear map if and only if $\mathfrak{D}_n(A) = \mathbf{Z}$.
- Let $s \in \mathbf{Z}^*$ s.t. the modules $E[1/s]$ and $H[1/s]$ are free over $\mathbf{Z}[1/s]$.
  Let $\varphi_s : E[1/s] \to H[1/s]$ the extension of φ by $\mathbf{Z} \to \mathbf{Z}[1/s]$.
  Then for each r we get $\mathfrak{D}_r(\varphi)\mathbf{Z}[1/s] = \mathfrak{D}_r(\varphi_s)$ (usual determinantal ideals).
- Let $(s_1, \dots, s_n)$ be comaximal in $\mathbf{Z}$. A linear system $AX = B$ (with pseudo-matrices A, B, X) admits a solution in $\mathbf{Z}$ if and only if it admits a solution in each $\mathbf{Z}[1/s_i]$.

3. Computations with pseudo-matrices

Let $\mathfrak{a}$ and $\mathfrak{b}$ be two finitely generated ideals of $\mathbf{Z}$ and M be a module with pseudo-basis $\mathcal{E} = \big((e_1, \mathfrak{a}), (e_2, \mathfrak{b})\big)$. If $s + t = 1$, $s\mathfrak{a} \subseteq \mathfrak{b}$ and $t\mathfrak{b} \subseteq \mathfrak{a}$, another pseudo-basis of M is $\mathcal{H} = \big((f_1, \mathfrak{a} + \mathfrak{b}), (f_2, \mathfrak{a} \cap \mathfrak{b})\big)$ where $f_1 = te_1 + se_2$ et $f_2 = -e_1 + e_2$. We get the following "Bezout pseudo-matrix " of change of pseudo-bases from $\mathcal{E}$ to $\mathcal{H}$.

$$B = \mathscr{M}_{\mathcal{H},\mathcal{E}}(\mathrm{Id}_M) = \begin{array}{c} \\ \mathfrak{a} \\ \mathfrak{b} \end{array} \begin{array}{c} \begin{array}{cc} \mathfrak{a}+\mathfrak{b} & \mathfrak{a}\cap\mathfrak{b} \end{array} \\ \begin{bmatrix} t & -1 \\ s & 1 \end{bmatrix} \end{array},$$

with inverse

$$\mathscr{M}_{\mathcal{E},\mathcal{H}}(\mathrm{Id}_M) = \begin{array}{c} \\ \mathfrak{a}+\mathfrak{b} \\ \mathfrak{a}\cap\mathfrak{b} \end{array} \begin{array}{c} \begin{array}{cc} \mathfrak{a} & \mathfrak{b} \end{array} \\ \begin{bmatrix} 1 & 1 \\ -s & t \end{bmatrix} \end{array}.$$

The Bezout pseudo-matrices and the analogues of Gauss pivoting matrices allow us to compute the reduction of pseudo-matrices to convenient "normal forms", analogous to HNF (Hermite normal form) for Prüfer domains and to SNF (Smith normal form) for Prüfer domains of dimension 1.

For dealing with pseudo-matrices over Prüfer domains of dimension 1 we use an algorithm in some zero-dimensional quotient rings: *a zero-dimensional arithmetic ring is a principal ideal ring and a matrix over it can be reduced to a Smith normal form by elementary row and column manipulations.*

Two kinds of easy consequences of these reductions of pseudo-matrices:

- The general discussion of linear systems over Prüfer domains (coefficients and unknowns in $\mathbf{Z}$)
- Theoretical results on the structure of finitely presented modules, finitely generated projective modules and linear maps between these modules: a finitely generated sub-$\mathbf{Z}$-module of $\mathbf{K}^n$ is finitely generated projective, a finitely generated projective module is a direct sum of rank one projective submodules, the kernel of a linear map between finitely generated projective modules is a direct summand, and so on...

References

[1] Lombardi H. & Quitté C. *Algèbre Commutative. Méthodes constructives.* Calvage&Mounet (2011).

[2] English version of [ACMC]. Springer (2015).

[3] Cohen H. *Advanced topics in computational number theory.* Graduate texts in mathematics 193. Springer-Verlag (1999).

[4] Díaz-Toca G.-M., Lombardi H. & Quitté C. *Modules sur les anneaux commutatifs.* Calvage&Mounet (2014).

[5] Mines R., Richman F. & Ruitenburg W. *A Course in Constructive Algebra.* Universitext. Springer-Verlag, (1988).

Departamento de Matemática Aplicada, Universidad de Murcia, 30100 Murcia, Spain.
E-mail address: gemadiaz@um.es

Laboratoire de Mathématiques, Université de Franche-Comté, 25030 Besançon, France
E-mail address: henri.lombardi@univ-fcomte.fr

THE CONDITION NUMBER OF POLYNOMIALS AND ITS RELATIONSHIP WITH A SET OF POINTS ON THE SPHERE

UJUÉ ETAYO

ABSTRACT. Finding roots of polynomial equations is one of the most important tasks in theoretical and practical computational algebra. With independence of the method that we choose to solve a particular instance of a problem, the quality of the solutions depends strongly on the so called "condition number" of the input.

In this talk, we will present the condition number introduced by Shub and Smale [3] and we will explain some of its properties. One of the main open problems in the study of the condition number is that of generating perfectly conditioned polynomials, i.e. polynomials all of whose roots have a small (optimal) condition number. In an attempt to understand this problem we will present a formula from Armentano, Beltrán and Shub [2]

$$\mathcal{E}_0(\omega_n) = \frac{1}{2}\sum_{i=1}^{n} \ln(\mu(f, z_i)) + \frac{n}{2}\ln\left(\frac{\prod_{i=1}^{n}\sqrt{1+|z_i|^2}}{||f||}\right) - \frac{n}{4}\ln(n)$$

that combines the condition number and a norm of polynomials (the Bombieri-Weyl norm), relating them to yet another classical object: the logarithmic energy of a collection of points in the complex projective space.

We will present a conjecture that drops naturally from the formula and we will study it for the first non-trivial case: the problem of polynomials of degree five. Some theoretical and numerical results confirming this conjecture will be given.

INTRODUCTION

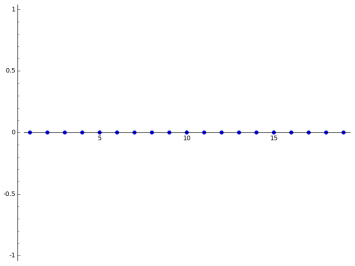
The solution of polynomials and systems of polynomial equations is one of the most important tasks in theoretical and practical computational algebra. It is a fact that, for some polynomials, a little preturbation on their coefficients causes large deviations in their roots. Let us ilustrate this sentence with an example.

Definition 0.1. The Wilkinson's polynomial is defined by

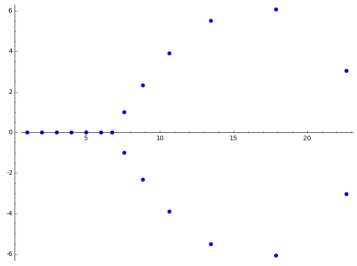
$$p_W(x) = \prod_{i=1}^{20}(x-i)$$

The roots of the Wilkinson's polynomial are the natural numbers between 1 and 20. But, what happens if we modify slightly one of its coefficients? Let us modify, for example, the coefficient of x^{18}.

As we can see in figure 1b, the roots of the modified polynomial change dramatically. This sensitivity of the roots is controlled by the condition number of the polynomial.



(A) Roots of $p_W(x)$



(B) Roots of $p_W(x) + 0.001x^{18}$

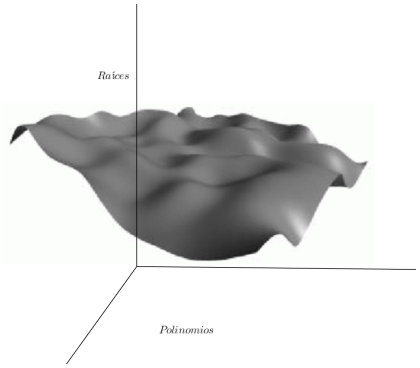
1. Condition number

Before defining the condition number, we are going to introduce the Bombieri-Weyl norm.

Definition 1.1. Let $f = z^n + a_{n-1}z^{n-1} + ... + a_1 z + a_0$ be a polynomial of $\mathbb{C}[z]$, then we can define the *Bombieri-Weyl norm* of f as follows.

$$||f||^2 = \sum_{j=0}^{n} \binom{n}{j}^{-1} |a_j|^2$$

Now, we are ready to introduce the concept of condition number.



Consider the vector space of polynomials in one varible of degree up to n. Then, we can consider the product vector space of this one and the complex numbers, the elements of this vector space will be pairs (f, z). We can prove that the set of the pairs $\{(f, z) : f(z) = 0\}$ is a Riemannian manifold that we will call the solution variety.

Definition 1.2. Let f be a univariate polynomial and let z be one of the roots of f. Then the condition number of f at its root z is given by the operator norm of the inverse of the differential of the projection $\Pi_{Pol}(f, z) = f$ at the elements of the solution variety.

$$\text{cond}_f(z) = ||D\Pi_{Pol}(f, z)^{-1}||_{op}$$

It is standard to work with a normalization of the previous condition number introduced by Shub and Smale [3].

Definition 1.3. Let f be a univariate polynomial and let z be one of its roots, then the condition number of Shub-Smale of f at its root z is called μ and is given by

$$\mu(f, z) = \sqrt{n} \ \ \text{cond}_f(z)$$

The following characterization is also given in [3].

Proposition 1.4. *Let f be a univariate polynomial and let z be one of its roots, then*

$$\mu(f,z) = \frac{\sqrt{n}(1+|z|^2)^{\frac{n-2}{2}}}{|f'(z)|}||f||$$

where $||f||$ is the Bombieri-Weyl norm of f.

Note that $\mu(p_W, 15) \approx 10^{25}$, which gives a concise explanation to the behavior of that zero under perturbations of the Wilkinson's polynomial. A similar computation can be done for other zeros.

One of the most important problems related to the condition number, first proposed in [3], is to find explicitly a family $\{f_n\}_{n\in\mathbb{N}}$ with $\mu(f_n) = \max\{\mu(f_n, z) : z \text{ root of } f_n\} \leq n$. This problem drops directly from the study of condition number and it is also related to a problem involving homotopy methods: Bürgirser and Cucker proved in [4] that

$$\text{complexity of } \textit{path following} \text{ starting at } (g,z) \leq \text{cte.} n^{\frac{5}{2}} \mu^2_{\max}(g,z).$$

So, finding polynomials with small condition number is an interesting task, with remarkable aplications to other problems.

2. Armentano-Beltrán-Shub formula

In our way to find polynomials of degree n with a small condition number, we will present a formula from Armentano, Beltrán and Shub [2] (see [5] for a proof):

$$\mathcal{E}_0(\omega_n) = \underbrace{\frac{1}{2}\sum_{i=1}^{n} \ln(\mu(f,z_i))}_{\mathcal{M}} + \underbrace{\frac{n}{2}\ln\left(\frac{\prod_{i=1}^{n}\sqrt{1+|z_i|^2}}{||f||}\right)}_{\mathcal{N}} - \frac{n}{4}\ln(n), \tag{1}$$

where $\omega_n = \{x_1, ..., x_n\}$ is the collection of points of the complex projective space, z_i is a projection of x_i onto the complex plane, $f = \prod_{i=1}^{n}(z - z_i)$ is a complex polynomial of degree n, $\mathcal{E}_0$ denotes the logarithmic energy of the subset of points ω_n (see definition 2.1 below), an expression from *potential theory*, $\mathcal{M}$ belongs to the field of *numerical stability* and $\frac{\prod_{i=1}^{n}\sqrt{1+|z_i|^2}}{||f||}$ is the quotient bewteen the product of the Bombieri-Weyl norms of the factors of f and the Bombieri-Weyl norm of f, a classical object in *number theory* [6].

Definition 2.1. The logarithmic energy of a collection of points $\omega_n = \{x_1, ..., x_n\}$ in the Riemann Sphere is defined by

$$\mathcal{E}_0(\omega_n) = \sum_{i,j=1, i<j}^{n} \ln\left(\frac{1}{||x_i - x_j||}\right)$$

Now we present our hypothesis.

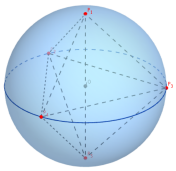
Hypothesis 2.2. *The following are equivalent:*

(1) *Minimize* $\mathcal{E}_0(\omega_n)$ (2) *Minimize* $\sum_{i=1}^{n} \ln(\mu(f, z_i))$ (3) *Maximize* $\frac{\prod_{i=1}^{n} \sqrt{1+|z_i|^2}}{||f||}$

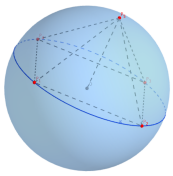
Little is known about problems (2) and (3), but problem (1) has been studied as Whyte problem, or, equivalentely, the Elliptic Fekete points problem.

$$\boxed{\operatorname{argmin} \sum_{i,j=1, i<j}^{n} \ln\left(\frac{1}{||x_i - x_j||}\right) \quad \textit{Whyte's problem}}$$

The minimal logarithmic energy problem is only solved for $2, 3, 4, 5, 6$ and 12 points and we have tested our hypothesis in the first interesting case: 5 points. In order to do that, we have worked with the pair of configurations of five points that minimize some Riesz energy (a classical generalization of the electromagnetic potential): a bipyramidal structure and pyramidal structure depending on the heigh of the pyramide. We got these values for each structure:



Bipyramidal structure	**Pyramidal structure**
$\mathcal{E}_0^{\Diamond}(\omega_5) = 2.511$	$\mathcal{E}_0^{\triangle}(\omega_5) = 2.520$
$\mathcal{M}^{\Diamond} = 4.\widehat{740}$	$\mathcal{M}^{\triangle} \approx 4.897$
$\mathcal{N}^{\Diamond} \approx 4.472$	$\mathcal{N}^{\triangle} \approx 4.459$



We conclude that our hypothesis works for the first interesting case, that is, the polynomials of degree five and, by using this formula, we have also found a polynomial with small condition number employing a new procedure.

References

[1] Dragnev, Peter D and Legg, DA and Townsend, DW *Discrete logarithmic energy on the sphere*, Pacific journal of mathematics **207**, 345–358, 2002.

[2] Diego Armentano and Carlos Beltrán and Michael Shub *Minimizing the discrete logarithmic energy on the sphere: the role of random polynomials.*, Transactions of the American Mathematical Society **363**, 2955–2965, 2011.

[3] Shub, Michael and Smale, Steve *Complexity of Bezout's Theorem: III. Condition Number and Packing.*, J. Complexity **9**, 4-14, 1993.

[4] Bürgisser, Peter and Cucker, Felipe *Condition: The Geometry of Numerical Algorithms*, Springer Publishing Company, Incorporated, 2013.

[5] Carlos Beltrán *A Facility Location Formulation for Stable Polynomials and Elliptic Fekete Points*, Foundations of Computational Mathematics **15**, 125-157, 2015.

[6] Beauzamy, Bernard and Bombieri, Enrico and Enflo, Per and Montgomery, Hugh L. *Products of polynomials in many variables*, Journal of Number Theory **36**, 219–245, 1990.

[7] L. L. Whyte *Unique Arrangements of Points on a Sphere*, The American Mathematical Monthly **9**, 606-611, 1952.

Universidad de Cantabria

E-mail address: etayomu@unican.es

ON THE UNIVERSAL ENVELOPING ALGEBRA OF AN n-LIE ALGEBRA

XABIER GARCÍA–MARTÍNEZ, RUSTAM TURDIBAEV, AND TIM VAN DER LINDEN

ABSTRACT. The aim of this paper is to investigate in which sense, for $n \geq 3$, n-Lie algebras admit universal enveloping algebras. There have been some attempts at a construction but after analysing those we come to the conclusion that they cannot be valid in general. We give counterexamples and sufficient conditions.

We then study the problem in its full generality, showing that universality is incompatible with the wish that the category of modules over a given n-Lie algebra L is equivalent to the category of modules over the associated algebra $U(L)$. Indeed, an *associated algebra functor* $U\colon n\text{-}\mathtt{Lie}_{\mathbb{K}} \to \mathtt{Ass}_{\mathbb{K}}$ inducing such an equivalence does exist, but this kind of functor never admits a right adjoint.

We end the paper by introducing a (co)homology theory based on the associated algebra functor U.

The algebraic structure of n-Lie algebras (also named Filippov algebras and Nambu algebras) is a natural generalization of Lie algebras. It is a $\mathbb{K}$-module with an n-ary, skew-symmetric operation which is also a derivation. There exist different generalizations of Lie algebras to n-ary brackets, such as Lie triple systems or the generalized Lie algebras. In the recent years it has been found that this structure is highly relevant in some areas of physics such as Nambu mechanics or string and membrane theory.

In this talk we generalize to n-Lie algebras a useful tool to study Lie algebras, the universal enveloping algebra. Let L be a Lie algebra. The universal enveloping algebra of L, denoted by $U(L)$ has two interesting properties. The first one is that the category of Lie modules over L is equivalent to the category of standard modules over the universal enveloping algebra of L. The second property is that the functor $U\colon \mathtt{Lie}_{\mathbb{K}} \to \mathtt{Ass}_{\mathbb{K}}$ has a right adjoint $(-)_{Lie}\colon \mathtt{Ass}_{\mathbb{K}} \to \mathtt{Lie}_{\mathbb{K}}$, which endows an associative algebra with a Lie algebra structure via the bracket $[a,b] = ab - ba$. This adjoint pair gives us a universal property. For every Lie algebra homomorphism $f\colon L \to A_{Lie}$ there is a unique $g\colon U(L) \to A$ such that $f = g \circ i$, where i is the inclusion of L into $U(L)$.

There has been some attempts to introduce this concept ([1], [2]), we give an example where they are not valid. The problem is that a functor from $n\text{-}\mathtt{Lie}_{\mathbb{K}}$ to $\mathtt{Lie}_{\mathbb{K}}$, analogue to the Daletskii functor in Leibniz algebras, is given, but the image is not always a Lie algebra. We give some conditions to stablish when the construction of [1] is or is not a Lie algebra. We explain that this imprecise definition is not an obstruction to their papers since they work only with the simple n-Lie algebra over the complex numbers, and we prove that it works in that case. However, the definition of universal enveloping algebra is not correct.

In order to generalize the concept of universal enveloping algebras to the category of n-Lie algebras we face that a natural generalization of the functor $(-)_{Lie}$ is not known. Therefore, we define a functor $U\colon n\text{-}\mathtt{Lie}_{\mathbb{K}} \to \mathtt{Ass}_{\mathbb{K}}$ such that the category of modules over an n-Lie

algebra L is equivalent to the category of $U(L)$-modules. But it happens that this functor does not have a right adjoint. Moreover, we prove that any functor with this property, can not have a right adjoint. Then, our definition of $U(L)$ is called the universal enveloping algebra since the construction is natural and it is not possible to find a functor with the two desired properties.

Finally, we extend the Lie algebras (co)homology to a (co)homology theory using this new definition of universal enveloping algebra and we prove that it is different from the classical (co)homology theories.

References

[1] D. Bălibanu, J. van de Leur, Irreducible highest weight representations of the simple n-Lie algebra. Transform. Groups, 17 (3), 2012, p. 593–613.

[2] A. S. Dzhumadil'daev, Representations of vector product n-Lie algebras. Comm. Algebra, 32 (9), 2004, p. 3315–3326.

[3] X. García–Martinez, R. Turidbaev, T. van der Linden, Do n-Lie algebras have universal enveloping algebras. arXiv:1508.06940 (2015).

Universidade de Santiago de Compostela
E-mail address: xabier.garcia@usc.es

Universidade de Santiago de Compostela

Université Catholique de Louvain

SEPARABILITY TEST AND CYCLIC CONVOLUTIONAL CODES

JOSÉ GÓMEZ-TORRECILLAS, F. J. LOBILLO, AND GABRIEL NAVARRO

Abstract. We design an algorithm that decides if the sentence-ambient algebra, with finite semi-simple word-ambient algebra, of an ideal code is separable over its canonical commutative polynomial subring.

Introduction: Split ideal codes and separable ring extensions

Convolutional codes transmit sentences composed of a variable number of words. Since each of these words is an n–tuple of elements in a given finite field $\mathbb{F}$, the aforementioned sentences may be viewed as polynomials in $\mathbb{F}[z]$, where the indeterminate z is interpreted as a delay operator. Thus, a convolutional code is often defined as a submodule of the free module $\mathbb{F}[z]^n$. In order to avoid problems such as catastrophicity, the submodule is assumed to be an $\mathbb{F}[z]$–direct summand of $\mathbb{F}[z]^n$ (see e.g. [8]). A straightforward extension of the notion of a cyclic block code to the convolutional setting is to consider ideals of the ring $A[z]$, where $A = \mathbb{F}[x]/\langle x^n - 1\rangle$, with n coprime with the characteristic of $\mathbb{F}$. However, Piret observed [10] that this construction does not produce non-block codes, which led to his proposal of deforming the commutative multiplication of polynomials to obtain a sentence-ambient algebra that turned out to be an Ore extension $A[z;\sigma]$ of A by an $\mathbb{F}$–algebra automorphism σ. This approach has been further developed in [2, 5, 9] among others, allowing even a non-commutative word-ambient algebra A. Thus, an *ideal code* is defined in [9] as a left ideal of $A[z;\sigma]$ such that it is a direct summand as an $\mathbb{F}[z]$–submodule, where σ is an automorphism of a semi-simple finite $\mathbb{F}$–algebra A. Obviously, every left ideal of $A[z;\sigma]$ generated by an idempotent (we call it a *split ideal code*) gives an ideal code. If the ring extension $\mathbb{F}[z] \subseteq A[z;\sigma]$ is separable in the sense of [7], then every ideal code is split. We give a separability test for this ring extension. Our algorithm also computes, if it exists, a suitable element of separability that, according to [3], can be used to compute an idempotent generator and a parity-check matrix for every ideal code.

1. A ring extension separability test

Let A be an algebra over a field $\mathbb{F}$, and σ an $\mathbb{F}$–algebra automorphism of A. Consider $R = A[z;\sigma]$ the corresponding Ore extension, whose elements are polynomials in the variable z with the coefficients of A written on the right, and whose multiplication is based on the rule $az = z\sigma(a)$, for $a \in A$. Observe that $\mathbb{F}[z]$ is a subring of R, but R is not an $\mathbb{F}[z]$–algebra unless σ is the identity. The ring extension $\mathbb{F}[z] \subseteq R$ is separable if the multiplication map $\mu : R \otimes_{\mathbb{F}[z]} R \to R$ is a split epimorphism of R–bimodules, equivalently, there exits $q \in R \otimes_{\mathbb{F}[z]} R$ such that $qr = rq$ for all $r \in R$ and $\mu(q) = 1$. Such a q is called a *separability element of the extension* $\mathbb{F}[z] \subseteq R$. We use the notation $\sigma^{\otimes} = \sigma \otimes \sigma$.

Theorem 1.1. [6] *The ring extension $\mathbb{F}[z] \subseteq R$ is separable if and only if A is a separable $\mathbb{F}$–algebra and there exists a separability element $p \in A \otimes_{\mathbb{F}} A$ such that $\sigma^{\otimes}(p) = p$.*

An automorphism σ of A such that there exists a separability element $p \in A \otimes_{\mathbb{F}} A$ with $\sigma^{\otimes}(p) = p$ will be said to be a *separable*. We also say that p is a *separability element invariant under* $\sigma^{\otimes}$.

In what follows we assume that A is a finite semi-simple algebra over a finite field $\mathbb{F}$. Then A is a separable $\mathbb{F}$–algebra and, hence, the set of separability elements of the extension $\mathbb{F} \subseteq A$ is non empty. In view of Wedderburn-Artin's Theorem, we may expect that the search of a separability element invariant under $\sigma^{\otimes}$ can be reduced to the case where A is a simple finite algebra. Thus, supposse that A as a matrix algebra $\mathbb{F}$–algebra $\mathcal{M}_n(\mathbb{K})$, where $\mathbb{K}$ is a field extension of $\mathbb{F}$ of degree t. Fix a normal basis $\left\{\alpha^{q^0}, \ldots, \alpha^{q^{t-1}}\right\}$ of $\mathbb{K}$ over $\mathbb{F}$, and let $\left\{\beta^{q^0}, \ldots, \beta^{q^{t-1}}\right\}$ be its dual basis (q denotes the number of elements of $\mathbb{F}$). Let $\{E_{ij} : 0 \leq i, j \leq n-1\}$ be the standard basis of $\mathcal{M}_n(\mathbb{K})$. Consider the following sets

$$(A \otimes_{\mathbb{F}} A)^A = \{p \in A \otimes_{\mathbb{F}} A : ap = pa,\ \forall a \in A\}$$

$$E_0 = \{p \in (A \otimes_{\mathbb{F}} A)^A : \mu(p) = 0\}, \qquad E_1 = \{p \in (A \otimes_{\mathbb{F}} A)^A : \mu(p) = 1\}$$

A basis of $(A \otimes_{\mathbb{F}} A)^A$ as an $\mathbb{F}$–vector space is $\{p_{ijk} : 0 \leq, i, j \leq n-1, 0 \leq k \leq t-1\}$, where

$$p_{ijk} = \sum_{l=0}^{n-1} \sum_{h=0}^{t-1} E_{li} \alpha^{q^k} \alpha^{q^h} \otimes \beta^{q^h} E_{jl} \in A \otimes_{\mathbb{F}} A.$$

By [4, Proposition 1], E_0 is an $(n^2-1)t$–dimensional $\mathbb{F}$–vector subspace of $(A \otimes_{\mathbb{F}} A)^A$ with basis $\mathcal{E} = \{p_{ijk} \mid 0 \leq i \neq j \leq n-1, 0 \leq k \leq t-1\} \cup \{p_{00k} - p_{iik} \mid 1 \leq i \leq n-1, 0 \leq k \leq t-1\}$. Moreover the set E_1 of all separability elements is an affine subspace of the same dimension. In fact, $E_1 = \{p_1 + q \mid q \in E_0\}$ where $p_1 = \sum_{k=0}^{t-1} \mathrm{Tr}_{\mathbb{K}/\mathbb{F}}(\beta) p_{00k}$ ($\mathrm{Tr}_{\mathbb{K}/\mathbb{F}}$ denotes the trace function of the extension).

Now, there is $p \in E_1$ such that $\sigma^{\otimes}(p) = p$ if and only if there exists $q \in E_0$ such that $\sigma^{\otimes}(p_1 + q) = p_1 + q$, equivalently,

$$(\sigma^{\otimes} - id)(p_1) \in (id - \sigma^{\otimes})(E_0), \tag{1}$$

and, since we know an explicit basis $\mathcal{E}$ of E_0, we may compute a system of generators of the $\mathbb{F}$–vector space $(id - \sigma^{\otimes})(E_0)$. Of course, in order to make effective the condition (1) we need a good expression in coordinates of the automorphism $\sigma^{\otimes} = \sigma \otimes \sigma$ as an $\mathbb{F}$–linear map. This is done in [4] by using the Kronecker product of matrices, which allows us to design an algorithm to solve an $\mathbb{F}$–linear system equivalent to the condition (1), see [4, Algorithm 1].

Let us now show how to extend [4, Algorithm 1] to any finite semisimple $\mathbb{F}$–algebra A. It follows from Artin–Wedderburn's Theorem and Wedderburn's Little Theorem that A can be expressed, up to isomorphisms, as a direct sum of two-sided ideals

$$A = A^{(1)} \oplus \cdots \oplus A^{(s)} \text{ with } \sigma(A^{(j)}) = A^{(j)} \text{ for all } 1 \leq j \leq s, \tag{Hyp 1}$$

such that for each $1 \leq j \leq s$ there exist m_j, n_j and a field extension $\mathbb{F} \subseteq \mathbb{K}_j$ of degree t_j such that

$$A^{(j)} = A_1^{(j)} \oplus \cdots \oplus A_{m_j}^{(j)},\ A_i^{(j)} = \mathcal{M}_{n_j}(\mathbb{K}_j) \text{ and } \sigma(A_i^{(j)}) = A_{i+1}^{(j)}, \tag{Hyp 2}$$

where $A^{(j)}_{m_j+1} = A^{(j)}_1$. By [1, Theorem 2.4], for each pair i, j there exists a regular matrix $U_{ij} \in \mathcal{M}_{n_j}(\mathbb{K}_j)$ and $0 \leq h_{ij} \leq t_j - 1$ such that

(Hyp 3) $$\sigma_{|A^{(j)}_i} = \sigma_{ij} = \sigma_{U_{ij}} \sigma_{\tau_{h_{ij}}},$$

where $\sigma_{\tau_{h_{ij}}}$ denotes the component-wise matrix extension of the h_{ij}–th power of the suitable Frobenius automorphism.

We need to fix some notation. Let $e_1, \ldots, e_m$ be a set of central ortogonal idempotents of an $\mathbb{F}$–algebra B, and let $B = B_1 \oplus \cdots \oplus B_m$ the corresponding direct sum decomposition with $B_i = Be_i$ for all $i = 1, \ldots, m$. Then

$$B \otimes_{\mathbb{F}} B = \bigoplus_{1 \leq i,j \leq m} B_i \otimes_{\mathbb{F}} B_j \tag{2}$$

is a B–bimodule direct sum decomposition.

Proposition 1.2. *Assume that B is a separable $\mathbb{F}$–algebra and let σ be an algebra automorphism over B such that $\sigma(B_i) = B_i$ for all $1 \leq i \leq m$. Let $\sigma_i = \sigma_{|B_i} : B_i \to B_i$. Then σ is separable if and only if σ_i is separable for all $1 \leq i \leq m$. Moreover, every separability element $p \in B \otimes_{\mathbb{F}} B$ invariant under σ is obtained as $p = \sum_{i=1}^m p_i$, where $p_i \in B_i \otimes_{\mathbb{F}} B_i$ is a separability element invariant under $\sigma_i^{\otimes}$ for every $i = 1, \ldots, m$.*

By applying Proposition 1.2 to (Hyp 1), we get that the separability of a given σ reduces to the separability of its restrictions to $A^{(j)}$ for all $1 \leq j \leq s$. Now, let us consider the situation described in (Hyp 2). So assume now that B is a finite semisimple $\mathbb{F}$–algebra and there exists a complete set of central ortogonal idempotents $\{e_1, \ldots, e_m\}$ such that $\sigma(e_i) = e_{i+1}$ where $e_{m+1} = e_1$. Thus, for all $1 \leq i \leq m$, $\sigma_i = \sigma_{|B_i} : B_i \to B_{i+1}$ is an isomorphism and $B_i = Be_i = \mathcal{M}_n(\mathbb{K})$ for suitable n and field extension $\mathbb{F} \subseteq \mathbb{K}$.

Proposition 1.3. *With the previous assumptions, the automorphism $\sigma : B \to B$ is separable if and only if the composition $\sigma_m \circ \cdots \circ \sigma_1 : B_1 \to B_1$ is a separable automorphism.*

Remark 1.4. Following [3, Proposition 13], if p_1 is a separability element for the extension $\mathbb{F} \subset B_1$ invariant under $(\sigma_m \circ \cdots \circ \sigma_1)^{\otimes}$, then $p = p_1 + \sum_{i=1}^{m-1} (\sigma_i \circ \cdots \circ \sigma_1)^{\otimes}(p_1)$ is a separability element for $\mathbb{F} \subset B$ invariant under $\sigma^{\otimes}$.

It remains to obtain an appropriate description of $\sigma_m \circ \cdots \circ \sigma_1$ as an automorphism on the $\mathbb{F}$–algebra $B_1 = \mathcal{M}_n(\mathbb{K})$. As observed before, [1, Theorem 2.4] gives that $\sigma_i = \sigma_{U_i} \sigma_{\tau_{h_i}}$. In order to apply [4, Algorithm 1] to the automorphism $\sigma_m \circ \cdots \circ \sigma_1$ we need a description of this automorphism as the composition of an inner automorphism and the component-wise extension of an $\mathbb{F}$–automorphism of $\mathbb{K}$ to $\mathcal{M}_n(\mathbb{K})$. So let $U \in \mathcal{M}_n(\mathbb{K})$ be a non-singular matrix and let $0 \leq h \leq t - 1$. A straightforward computation shows that

$$\sigma_{\tau_h} \circ \sigma_U = \sigma_{\sigma_{\tau_h}(U)} \circ \sigma_{\tau_h},$$

whence we get easily Proposition 1.5 below.

Proposition 1.5. *If $\sigma_i = \sigma_{U_i} \sigma_{\tau_{h_i}}$ for all $i = 1, \ldots, m$, then $\sigma_m \circ \cdots \circ \sigma_1 = \sigma_U \sigma_{\tau_h}$, where $U = U_m \sigma_{\tau_{h_m}}(U_{m-1}) \sigma_{\tau_{h_m + h_{m-1}}}(U_{m-2}) \ldots \sigma_{\tau_{h_m + \cdots + h_2}}(U_1)$ and $h = h_m + \cdots + h_1$ (all the sums computed module t).*

Propositions 1.2, 1.3 and 1.5 directly imply the correctness of the following extension of [4, Algorithm 1].

Algorithm 1.6. (`Separable Automorphism`)

Input: A finite semisimple algebra A and an $\mathbb{F}$–algebra automorphism σ of A, presented as in (Hyp 1), (Hyp 2) and (Hyp 3).

Output: A separability element invariant under $\sigma^{\otimes}$ if σ is a separable automorphism. 0 otherwise.

1: **for** $j = 1, \ldots, r$ **do**

2: Use Proposition 1.5 to compute U_j and h_j such that

$$\sigma_{U_j}\sigma_{\tau_{h_j}} = \sigma_{U_{jm_j}}\sigma_{\tau_{h_{jm_j}}} \circ \cdots \circ \sigma_{U_{j1}}\sigma_{\tau_{h_{j1}}}.$$

3: Let q be the output of Algorithm [4, Algorithm 1] applied to $\sigma_{U_j}\sigma_{\tau_{h_j}}$.

4: **if** $q = 0$ **then**

5: **return** 0

6: $p_j = q + \sum_{i=1}^{m_j-1}(\sigma_{U_{ji}}\sigma_{\tau_{h_{ji}}} \circ \cdots \circ \sigma_{U_{j1}}\sigma_{\tau_{h_{j1}}})^{\otimes}(q)$

7: **return** $p_1 + \cdots + p_r$

Remark 1.7. Non trivial explicit examples of application of [4, Algorithm 1] and Algorithm 1.6 can be seen in [4] and [6].

Acknowledgement. Research supported by grants MTM2013-41992-P and TIN2013-41990-R from Ministerio de Economía y Competitividad and from Fondo Europeo de Desarrollo Regional FEDER.

References

[1] Cauchon, G., Robson, J. C., 1978. Endomorphisms, derivations, and polynomial rings. Journal of Algebra 53 (1), 227–238.

[2] Gluesing-Luerssen, H., Schmale, W., 2004. On cyclic convolutional codes. Acta Applicandae Mathematica 82 (2), 183–237.

[3] Gómez-Torrecillas, J., Lobillo, F. J., Navarro, G., 2014. Ideal codes over separable ring extensions. CoRR abs/1408.1546.

[4] Gómez-Torrecillas, J., Lobillo, F. J., Navarro, G., 2015a Separable automorphisms on matrix algebras over finite field extensions: Applications to ideal codes. In: Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation. ISSAC '15. ACM, New York, NY, USA, pp. 189–195.

[5] Gómez-Torrecillas, J., Lobillo, F. J., Navarro, G., 2016. Convolutional codes with a matrix-algebra word-ambient. Advances in Mathematics of Communications, 10 (1), 29–43.

[6] Gómez-Torrecillas, J., Lobillo, F. J., Navarro, G., 2015b. Computing separability elements for the sentence-ambient algebra of split ideal codes. Journal of Symbolic Computation. To appear.

[7] Hirata, K., Sugano, K., 1966. On semisimple extensions and separable extensions over non commutative rings. Journal of the Mathematical Society of Japan 18 (4), 360–373.

[8] Johannesson, R., Zigangirov, K. S., 1999. Fundamentals of Convolutional Coding. Wiley-IEEE Press.

[9] López-Permouth, S. R., Szabo, S., 2013. Convolutional codes with additional algebraic structure. Journal of Pure and Applied Algebra 217 (5), 958 – 972.

[10] Piret, P., 1976. Structure and constructions of cyclic convolutional codes. IEEE Transactions on Information Theory 22 (2), 147–155.

University of Granada
E-mail address: `gomezj@ugr.es, jlobillo@ugr.es, gnavarro@ugr.es`

RESULTANTS AND SUBRESULTANTS THROUGH EVALUATION

LAUREANO GONZALEZ–VEGA

Abstract. We show how resultants and subresultants can be described in terms of the evaluation of the considered two polynomials in any set of points not necessarily the roots of one of the polynomials and not necessarily different.

Introduction

Resultants and subresultants play a fundamental role in Computer Algebra: for example they provide fraction free algorithms for computing the gcd of two polynomials or they are basic tools in quantifier elimination and cylindrical algebraic decomposition algorithms.

Recently there has been an increasing interest on deriving formulae connecting the resultant and the subresultant of two polynomials with their roots. For example if

$$P(x)=\sum_{i=0}^{p} a_i x^i = a_p \prod_{i=1}^{p}(x-\alpha_i) \qquad Q(x)=\sum_{j=0}^{q} b_j x^j = b_q \prod_{j=1}^{q}(x-\beta_i)$$

and $\mathbf{Res}(P,Q)$ denotes the resultant of P and Q then it is easy to prove that

$$\mathbf{Res}(P,Q)=a_p^q\prod_{i=1}^{p}Q(\alpha_i)=b_q^p\prod_{j=1}^{q}P(\beta_j)=a_p^q b_q^p\prod_{i=1}^{p}\prod_{j=1}^{q}(\alpha_i-\beta_j)\ . \tag{1}$$

If $P(x)$ is monic and squarefree, $k\in\{0,\ldots,\min(p,q)-1\}$, $N=\{1,\ldots,p\}$,

$$P_I(x)=\prod_{i\in I}(x-\alpha_i)\ ,$$

and $\mathbf{Sres}_k(P,Q)$ denotes the k–th subresultant of P and Q then:

$$\mathbf{Sres}_k(P,Q)=\sum_{\substack{I\uplus J=N\\ |J|=k}}\frac{\mathrm{res}(P_I,Q)}{\mathrm{res}(P_I,P_J)}P_J(x)=\sum_{\substack{I\uplus J=N\\ |J|=k}}\frac{\prod_{i\in I}Q(\alpha_i)}{\prod_{i\in I,j\in J}(\alpha_i-\alpha_j)}P_J(x), \tag{2}$$

Proofs of this formula, introduced by J. J. Sylvester, can be found in [4, 5, 6] where a more general expression is derived by using the Double Sylvester Sums introduced by J. J. Sylvester too (see [7, 8]). In [2], formulae similar to (1) and (2) are introduced by evaluating the involved polynomials in an arbitrary set of nodes instead of the roots of one of the polynomials: the main drawback of these formulae is that they involve an exponential number of summands. In the next two sections, from the proof in [2], we identify two

Partially supported by the Spanish Ministerio de Economía y Competitividad and by the European Regional Development Fund (ERDF), under the project MTM2014-54141-P.

new formulae for resultants and subresultants following the same idea (relying only on the evaluation of the considered polynomials in an arbitrary set of nodes) together with two applications. The presented proofs are a direct adaptation of the proofs in [2] for the formulae there sharing a similar structure with the one in (2).

1. The resultant through the evaluation in an arbitrary set of nodes

We introduce here a new formula for the resultant (which was implicitely used in [2]).

Theorem 1.1. *Let $\{x_1,\ldots,x_{p+q}\}$ be a set with $p+q$ elements. Then:*

$$\mathbf{Res}(P,Q)=\frac{\begin{vmatrix} x_1^{q-1}P(x_1) & \ldots & x_{p+q}^{q-1}P(x_{p+q})\\ \vdots & & \vdots\\ P(x_1) & \ldots & P(x_{p+q})\\ x_1^{p-1}Q(x_1) & \ldots & x_{p+q}^{p-1}Q(x_{p+q})\\ \vdots & & \vdots\\ Q(x_1) & \ldots & Q(x_{p+q}) \end{vmatrix}}{V(x_1,\ldots,x_{p+q})}$$

where $V(x_1,\ldots,x_{p+q})$ denotes the Vandermonde determinant of $\{x_1,\ldots,x_{p+q}\}$.

The proof is based on the following presentation of the resultant

$$\mathbf{Res}(P,Q)=\begin{vmatrix} a_p & \ldots & a_0 & & \\ & \ddots & & \ddots & \\ & & a_p & \ldots & a_0\\ b_q & \ldots & b_0 & & \\ & \ddots & & \ddots & \\ & & b_q & \ldots & b_0 \end{vmatrix}=\begin{vmatrix} a_p & \ldots & \ldots & \ldots & \star & x^{q-1}P(x)\\ & \ddots & & & \vdots & \vdots\\ & & a_p & \ldots & a_1 & P(x)\\ b_q & \ldots & \ldots & \ldots & \star & x^{p-1}Q(x)\\ & \ddots & & & \vdots & \vdots\\ & & b_q & \ldots & b_1 & Q(x) \end{vmatrix}.$$

If $\{x_1,\ldots,x_{p+q}\}$ are considered as variables then expanding the last determinant with respect to all columns except the last one and keeping the highest possible degree of each variable, we obtain ("ldt" means "lowest degree terms")

$$\begin{vmatrix} x_1^{q-1}P(x_1) & \ldots & x_{p+q}^{q-1}P(x_{p+q})\\ \vdots & & \vdots\\ P(x_1) & \ldots & P(x_{p+q})\\ x_1^{p-1}Q(x_1) & \ldots & x_{p+q}^{p-1}Q(x_{p+q})\\ \vdots & & \vdots\\ Q(x_1) & \ldots & Q(x_{p+q}) \end{vmatrix}=\begin{vmatrix} a_p & \ldots & \ldots & \ldots & \star & x_{p+q}^{q-1}P(x_{p+q})\\ & \ddots & & & \vdots & \vdots\\ & & a_p & \ldots & a_1 & P(x_{p+q})\\ b_q & \ldots & \ldots & \ldots & \star & x_{p+q}^{p-1}Q(x_{p+q})\\ & \ddots & & & \vdots & \vdots\\ & & b_q & \ldots & b_1 & Q(x_{p+q}) \end{vmatrix}.$$

$$\cdot(x_1^{p+q-1}x_2^{p+q-2}\cdots x_{p+q-1})+\text{ldt}=\text{Res(P,Q)}\cdot(x_1^{p+q-1}x_2^{p+q-2}\cdots x_{p+q-1})+\text{ldt}$$

Dividing this determinant by

$$V(x_1,\ldots,x_{p+q})=x_1^{p+q-1}x_2^{p+q-2}\cdots x_{p+q-1}+\text{lowest degree terms},$$

the corresponding degree comparison produces the searched equality.

2. Subresultants through the evaluation in an arbitrary set of nodes

We use the following characterization for the subresultant of P and Q:

$$\mathbf{Sres}_t(P,Q) = \begin{vmatrix} a_0 & \dots & \dots & \dots & \star & x^{q-t-1}P(x) \\ & \ddots & & & \vdots & \vdots \\ & & a_p & \dots & \star & P(x) \\ b_0 & \dots & \dots & \dots & \star & x^{p-t-1}Q(x) \\ & \ddots & & & \vdots & \vdots \\ & & b_0 & \dots & \star & Q(x) \end{vmatrix}$$

where $0 \leq t \leq \min\{p,q\}-1$. If $\{x_1, x_2, \dots, x_r\}$ is a set of r different variables then we define

$$\underline{F}(x_1, x_2, \dots, x_r) = \begin{pmatrix} x_1^{q-t-1}P(x_1) & x_2^{q-t-1}P(x_2) & \dots & x_r^{q-t-1}P(x_r) \\ \vdots & \vdots & & \vdots \\ f(x_1) & f(x_2) & \dots & f(x_r) \\ x_1^{p-t-1}Q(x_1) & x_2^{p-t-1}Q(x_2) & \dots & x_r^{m-t-1}Q(x_r) \\ \vdots & \vdots & & \vdots \\ g(x_1) & g(x_2) & \dots & g(x_r) \end{pmatrix}$$

and, when $r = p+q-2t$, we define

$$\beta_{\underline{F}}(x_1, x_2, \dots, x_{p+q-2t}) = \frac{\det(\underline{F}(x_1, x_2, \dots, x_{p+q-2t}))}{V(x_1, x_2, \dots, x_{p+q-2t})} .$$

We introduce here a new formula for subresultants (which was implicitely used in [2]).

Theorem 2.1. *Let $\{x_1, x_2, \dots, x_{p+q-t}\}$ be a set with $p+q-t$ elements. Then:*

$$\mathbf{Sres}_t(P,Q) = \sum_{\substack{[p+q-t]=I \uplus J \\ |I|=p+q-2t, |J|=t}} \beta_{\underline{F}}((x_i)_{i\in I}) \cdot \frac{\prod_{j\in J}(x-x_j)}{\prod_{i\in I, j\in J}(x_i-x_j)} .$$

The proof of this theorem needs the following two lemmas.

Lemma 2.2.

$$\beta_{\underline{F}}(x_1, x_2, \dots, x_{p+q-2t}) = \mathbf{Sres}_t(P,Q)(x_{p+q-2t}) \cdot (x_1 \cdot x_2 \cdot \ldots \cdot x_{p+q-2t-1})^t + \text{ldt} .$$

Proof. Using the multilinearity of the determinant with respect to the sums on the columns of the numerator of $\beta_{\underline{F}}(x_1, x_2, \dots, x_{p+q-2t})$ and degree comparisons gives the coefficient of $(x_1 \cdot x_2 \cdot \ldots \cdot x_{p+q-2t-1})^t$ in $\beta_{\underline{F}}(x_1, x_2, \dots, x_{p+q-2t})$ as desired. □

Lemma 2.3. *Let $\{T_1, T_2, \dots, T_{p+q-t}\}$ be a set of $p+q-t$ different variables. Then:*

$$\beta_{\underline{F}}(x_1, x_2, \dots, x_{p+q-2t}) = \sum_{\substack{[p+q-t]=I \uplus J \\ |I|=p+q-2t, |J|=t}} \beta_{\underline{F}}((T_i)_{i\in I}) \cdot \frac{\prod_{l\in[p+q-2t], j\in J}(x_l - T_j)}{\prod_{i\in I, j\in J}(T_i - T_j)} .$$

Proof. Each polynomial in $\underline{F}$ is described in terms of the Lagrange Basis wrt the "nodes" $\{T_1, T_2, \ldots, T_{p+q-t}\}$. This shows that the matrix in the numerator of $\beta_{\underline{F}}(x_1, \ldots, x_{p+q-2t})$ is the product of the matrix $\underline{F}(T_1, \ldots, T_{p+q-t})$ (no longer square unless $t = 0$) and another matrix B. By using Cauchy–Binet formula and a quite complicated determinant computation on the minors of B, the desired formula is obtained. □

Proof of Theorem 2.1 is a consequence of these two lemmas.

3. Applications and further work

Theorem 1.1 can be used to deal with resultants when the considered polynomials are easy to evaluate but difficult to expand. For example, when computing the implicit equation of a parametric curve $(x(t), y(t))$: it requires only to replace $P(t) = x(t) - x$ and $Q(t) = y(t) - y$ in Theorem 1.1 and a set of $p + q$ different numbers where $P(t)$ and $Q(t)$ will be evaluated.

Apart from reducing the complexity of these expressions (at least for the formula in Theorem 2.1) we are also looking for formulae similar to those included here allowing "multiplicities" in the set of nodes (see [1] and [3]). First results are very promising and, for example, it is not very difficult to prove (when, in Theorem 1.1, $x_1 = x_2$):

$$\mathbf{Res}(P,Q) = \frac{\begin{vmatrix} [x^{q-1}P(x)]'_{x=x_1} & x_1^{q-1}P(x_1) & x_3^{q-1}P(x_3) & \ldots & x_{p+q}^{q-1}P(x_{p+q}) \\ \vdots & \vdots & \vdots & & \vdots \\ P'(x_1) & P(x_1) & P(x_3) & \ldots & P(x_{p+q}) \\ [x^{p-1}Q(x)]'_{x=x_1} & x_1^{p-1}Q(x_1) & x_3^{p-1}Q(x_3) & \ldots & x_{p+q}^{p-1}Q(x_{p+q}) \\ \vdots & \vdots & \vdots & & \vdots \\ Q'(x_1) & Q(x_1) & Q(x_3) & \ldots & Q(x_{p+q}) \end{vmatrix}}{V(x_3, \ldots, x_{p+q}) \prod_{k \geq 3} (x_1 - x_k)^2}.$$

The denominator can be also described in the same way than the numerator: it is enough to replace, in $V(x_1, \ldots, x_{p+q})$, the first column by its derivative with respect to x_1 and $x_2 = x_1$.

References

[1] D. A. Aruliah, R. M. Corless, G. M. Diaz-Toca, , L. Gonzalez-Vega, A. Shakoori: *The Bézout matrix for Hermite interpolants*. Linear Algebra and Its Applications 474, 12–29, 2015.

[2] F. Apery, J.–P. Jouanolou: *Élimination. Le cas d'une variable*. Collection Methodes. Hermann, 2006.

[3] C. D'Andrea, T. Krick, A. Szanto: *Subresultants in multiple roots*. Linear Algebra and Its Applications 438, 1969–1989, 2013.

[4] C. D'Andrea, H. Hong, T. Krick, A. Szanto: *An elementary proof of Sylvester's double sums for subresultants*. Journal of Symbolic Computation 42, 290–297, 2007.

[5] C. D'Andrea, H. Hong, T. Krick, A. Szanto: *Sylvester's double sums: The general case*. Journal of Symbolic Computation 44, 1164–1175, 2009.

[6] G. M. Diaz–Toca, L. Gonzalez–Vega: *Various new expressions for subresultants and their applications*. Applicable Algebra Engineering Communication Computing 15, 233–266, 2004.

[7] A. Lascoux, P. Pragacz: *Double Sylvester sums for subresultants and multi-Schur functions*. Journal of Symbolic Computation 35, 689–710, 2003.

[8] M.–F. Roy, A. Szpirglas: *Sylvester double sums and subresultants*. Journal of Symbolic Computation 46, 385–395, 2011.

Universidad de Cantabria, Spain
E-mail address: laureano.gonzalez@unican.es

A DECISION ALGORITHM FOR RATIONAL GENERAL SOLUTIONS OF FIRST-ORDER ALGEBRAIC ODES

GEORG GRASEGGER, N. THIEU VO, AND FRANZ WINKLER

Abstract. The objects under consideration in this paper are algebraic ordinary differential equations (AODEs). Rational general solutions of first-order AODEs depend on an arbitrary constant. In case this constant appears rationally as well we call the solution strong. We give an algorithm for deciding the existence of strong rational general solutions of first-order AODEs. In the affirmative case it computes such a solution. The main approach intrinsically uses rational parametrizations of algebraic curves.

Introduction

Rational solutions of algebraic differential equations have been studied in various papers. One approach is to consider the differential equation to be an algebraic one by replacing derivatives by independent variables. The resulting equation defines an algebraic variety for which parametrizations can be considered. Algorithms using this approach can be found in [3, 13, 15, 14, 8, 11, 5, 6, 2]. So far, however, there is no general algorithm. Even for first-order AODEs [13, 17, 2] present only procedures which might fail in certain cases. In [17] the goal is to determine algebraic solutions of first-order AODEs. Existence of solutions can be determined up to a given degree bound for the algebraic function. Here the goal is to determine rational solutions for strongly parametrizable first-order AODEs. In this paper we essentially collect some known ideas and equip them with a suitable parametrization algorithm in order to gain a full decision algorithm for strong rational general solutions of first-order AODEs. We consider first-order AODEs, $F(x, y, y') = 0$, i. e. F is an irreducible polynomial in x, y, y'. Replacing y' by an independent variable z we get an algebraic equation $F(x, y, z) = 0$ which defines a plane curve $\mathcal{C} = \left\{(a, b) \in \mathbb{A}^2(\overline{\mathbb{K}(x)}) \mid F(x, a, b) = 0\right\}$ over $\overline{\mathbb{K}(x)}$ for some algebraically closed field $\mathbb{K}$. We call it the *corresponding curve*. In a similar way one can define the corresponding surface.

A parametrization of a curve $\mathcal{C}$ in $\mathbb{A}^2(\mathbb{F})$ is a rational map $\mathcal{P} : \mathbb{A}^1(\mathbb{F}) \to \mathcal{C} \subseteq \mathbb{A}^2(\mathbb{F})$ such that the image of $\mathcal{P}$ is dense in $\mathcal{C}$ (with respect to the Zariski topology). If, furthermore, $\mathcal{P}$ is a birational equivalence, $\mathcal{P}$ is called a *proper* parametrization. A parametrization is called *optimal*, if the algebraic extension degree of its coefficient field over the field of definition is minimal (see [16] for further details).

We call an AODE *strongly parametrizable* if there exists a rational parametrization $\mathcal{P}(t) := (p_1(x, t), p_2(x, t))$ of the corresponding curve, where $p_1, p_2 \in \mathbb{K}(x, t)$. Note, that almost all first-order AODEs in the collection of Kamke [9] are strongly parametrizable.

A *rational solution* of an AODE, $F(x, y, y') = 0$, is a rational function $u \in \mathbb{K}(x)$, such that $F(x, u, u') = 0$. A solution u of the AODE is called a *strong rational general solution*, if $u = u(x, c) \in \mathbb{K}(x, c)$ where c is a transcendental constant over $\mathbb{K}(x)$.

1. Associated systems

Let $F(x, y, y') = 0$ be a first-order AODE and let $\mathcal{P}(t) = (p_1(t), p_2(t))$ be a rational parametrization of the corresponding curve. We assume further, that $y(x)$ is a rational solution of the AODE. Then one of the following has to be fulfilled.

(i) $(y(x), y'(x)) \notin \operatorname{im}(\mathcal{P})$, where $\operatorname{im}(\mathcal{P})$ is the image of $\mathcal{P}$. Then $(y(x), y'(x))$ can be determined from the finite set $\mathcal{C} \setminus \operatorname{im}(\mathcal{P})$.

(ii) $(y(x), y'(x)) = \mathcal{P}(\omega(x))$ for some $\omega(x) \in \overline{\mathbb{K}(x)}$. In this case we identify the algebraic function $\omega(x)$ with a point on the affine line $\mathbb{A}^1(\overline{\mathbb{K}(x)})$.

Let us now assume that $(y(x), y'(x)) = \mathcal{P}(\omega(x))$. By simple computations one can show that ω fulfills the following differential equation.

$$\omega' = \frac{p_2(x, \omega) - \frac{\partial p_1}{\partial x}(x, \omega)}{\frac{\partial p_1}{\partial t}(x, \omega)} .$$

We call it the *associated ODE*. Already Fuchs [4] describes a one-to-one correspondence between solutions of the AODE and the associated ODE. This correspondence can be made more precise for rational general solutions.

Theorem 1.1. *Let $F(x, y, y') = 0$ be a first-order AODE and let $\mathcal{P}$ be a proper rational parametrization of the corresponding curve. Then there is a one-to-one correspondence between rational general solutions of the AODE and rational general solutions of its associated differential equation.*

2. Rational General Solutions

Given a first-order AODE, $F(x, y, y') = 0$ we would like to find strong rational general solutions. For this we first need a strong parametrization. This can be achieved by parametrization algorithms which do computations in the lowest possible field extension (see for instance [16]). A rational curve is birationally equivalent to a line or a conic. It is known that the optimal field extension depends on the existence of rational points on this conic. We show that a similar approach as in [16, 7] can be used for finding rational points on this conic and hence for computing optimal parametrizations over $\mathbb{K}(x)$. Once we have such a parametrization we can compute the associated ODE. The following theorem gives rise to the final algorithm.

Theorem 2.1. *Let $F(x, y, y') = 0$ be a first-order AODE.*

(i) *If $F = 0$ has a strong rational general solution, then its associated quasi-linear differential equation is of the form*

$$\omega' = a_0(x) + a_1(x)\omega + a_2(x)\omega^2 ,$$

for some $a_0, a_1, a_2 \in \mathbb{K}(x)$.

(ii) *If $F = 0$ is strongly parametrizable and has a rational general solution, then its associated quasi-linear differential equation is of the same form.*

This theorem is a direct consequence of results of [1] or [12]. A Riccati equation is a first-order AODE of the form

$$\omega' = a_0(x) + a_1(x)\omega + a_2(x)\omega^2 ,$$

where $a_0, a_1, a_2 \in \mathbb{K}(x)$, $a_2 \neq 0$. It is well known that every Riccati equation can be transformed into an equation in normal form

$$y' - y^2 = a(x)\,,$$

where $a \in \mathbb{K}(x)$. The problem of finding rational solutions of a Riccati equation has been intensively studied in literature. We follow the approach presented in [2] for determining a rational general solution of the Riccati equation if there is any. This approach is based on [10] and a bound for the number of poles of a solution y which are not poles of a.

Combining all this information, the final algorithm can be given.

Algorithm 2.2 (Strong Rational General Solution)**.** Given a first-order AODE, $F(x, y, y') = 0$, where $F \in \mathbb{K}[x, y, z]$ is an irreducible polynomial. The output is either a strong rational general solution $y(x)$ if it exists, or an answer that such a solution cannot exist.

(1) Compute the genus of the corresponding curve. If it is non-zero, no strong solution exists.
(2) Compute an optimal parametrization $\mathcal{P}(t) = (p_1, p_2)$ of the corresponding curve.
(3) If p_1 or p_2 is not in $\mathbb{K}(x, t)$ there is no strong solution.
(4) Compute the associated ODE.
    (a) If the associated ODE is linear, use well known algorithms to compute a rational general solution ω.
    (b) If the associated ODE is a Riccati ODE, transform it into normal form and use ideas from [2, 10] to get a rational general solution ω.
    (c) Otherwise, there is no strong solution.
(5) If a rational solution ω of the associated ODE was found, then return $p_1(x, \omega(x))$. Otherwise there is no strong solution.

Example 2.3 (Example 1.537 in Kamke [9])**.** We consider the differential equation

$$F(x, y, y') = (xy' - y)^3 + x^6 y' - 2x^5 y = 0\,.$$

Its corresponding curve has a strong rational parametrization

$$\mathcal{P}(t) = \left(-\frac{t^3x^5 - t^2x^6 + (t - x)^3}{t^3x^5}, -\frac{2t^3x^5 - 2t^2x^6 + (t - x)^3}{t^3x^6} \right)\,.$$

Hence, the associated differential equation with respect to $\mathcal{P}$ is

$$\omega' = \frac{1}{x^2}\omega(2\omega - x)\,.$$

This is a Riccati equation and we can determine a rational general solution $\omega(x) = \frac{x}{1+cx^2}$. Hence, the differential equation $F(x, y, y') = 0$ has the rational general solution $y(x) = cx(x + c^2)$.

3. Conclusion

We have presented an algorithm for deciding whether a first-order AODE has a strong rational general solution. In the affirmative case the algorithm also computes such a solution. For strongly parametrizable first-order AODEs the method even decides the existence of rational general solutions.

References

[1] D. Behloul and S. S. Cheng. Computation of rational solutions for a first-order nonlinear differential equation. *Electronic Journal of Differential Equations (EJDE) [electronic only]*, 2011:1–16, 2011.

[2] G. Chen and Y. Ma. Algorithmic reduction and rational general solutions of first order algebraic differential equations. In *Differential equations with symbolic computation*, pages 201–212. Birkhäuser, Basel, 2005.

[3] R. Feng and X.-S. Gao. Rational General Solutions of Algebraic Ordinary Differential Equations. In J. Gutierrez, editor, *Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation*, ISSAC '04, pages 155–162, New York, 2004. ACM.

[4] L. Fuchs. Über Differentialgleichungen, deren Integrale feste Verzweigungspunkte besitzen. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften*, 1884:699–710, 1884.

[5] G. Grasegger, A. Lastra, J. R. Sendra, and F. Winkler. On Symbolic Solutions of Algebraic Partial Differential Equations. In V.P. Gerdt et al., editor, *Computer Algebra in Scientific Computing*, volume 8660 of *Lecture Notes in Computer Science*, pages 111–120. Springer International Publishing, 2014.

[6] G. Grasegger, A. Lastra, J. R. Sendra, and F. Winkler. A solution method for autonomous first-order algebraic partial differential equations. *Journal of Computational and Applied Mathematics*, 300:119–133, 2016.

[7] E. Hillgarter and F. Winkler. Points on algebraic curves and the parametrization problem. In *Automated Deduction in Geometry*, Lecture Notes in Computer Science, pages 189–207, New York, 1997. Springer Berlin Heidelberg.

[8] Y. Huang, L. X. C. Ngô, and F. Winkler. Rational General Solutions of Higher Order Algebraic ODEs. *Journal of Systems Science and Complexity*, 26(2):261–280, 2013.

[9] E. Kamke. *Differentialgleichungen: Lösungsmethoden und Lösungen*. B. G. Teubner, Stuttgart, 1997.

[10] J. J. Kovacic. An algorithm for solving second order linear homogeneous differential equations. *Journal of Symbolic Computation*, 2:3–43, 1986.

[11] A. Lastra, L. X. C. Ngô, J. R. Sendra, and F. Winkler. Rational General Solutions of Systems of Autonomous Ordinary Differential Equations of Algebro-Geometric Dimension One. *Publicationes Mathematicae Debrecen*, 86(1–2):49–69, 2015.

[12] J. Malmquist. Sur les fonctions a un nombre fini de branches définies par les équations différentielles du premier ordre. *Acta Mathematica*, 42(1):317–325, 1920.

[13] L. X. C. Ngô and F. Winkler. Rational general solutions of first order non-autonomous parametrizable ODEs. *Journal of Symbolic Computation*, 45(12):1426–1441, 2010.

[14] L. X. C. Ngô and F. Winkler. Rational general solutions of parametrizable AODEs. *Publicationes Mathematicae Debrecen*, 79(3–4):573–587, 2011.

[15] L. X. C. Ngô and F. Winkler. Rational general solutions of planar rational systems of autonomous ODEs. *Journal of Symbolic Computation*, 46(10):1173–1186, 2011.

[16] J. R. Sendra, F. Winkler, and S. Pérez-Díaz. *Rational Algebraic Curves, A Computer Algebra Approach*, volume 22 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin Heidelberg, 2008.

[17] N. T. Vo and F. Winkler. Algebraic General Solutions of First Order Algebraic ODEs. In P. V. Gerdt, W. Koepf, and V. E. Seiler, M. W.and Vorozhtsov, editors, *Computer Algebra in Scientific Computing*, pages 479–492, Cham, 2015. Springer International Publishing.

Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences (ÖAW)

E-mail address: georg.grasegger@ricam.oeaw.ac.at

Research Institute for Symbolic Computation (RISC), Johannes Kepler University Linz, Austria

E-mail address: tvongoc@risc.jku.at

E-mail address: franz.winkler@risc.jku.at

SELF-OVERLAYS AND SHAPE OF THE JULIA SET OF A RATIONAL MAP

LUIS JAVIER HERNÁNDEZ PARICIO AND MARÍA TERESA RIVAS RODRÍGUEZ

Abstract. When a semi-flow is induced by a rational map f of degree d defined on the Riemann sphere, the associated Julia set $J(f)$ is right-invariant and the restriction map $f|_{J(f)}\colon J(f) \to J(f)$ has d sheets and it admits a branched overlay structure. Our algorithms use the notion of spherical multiplier to provide an inverse system of cubic complexes approaching $J(f)$, and the shape invariants of the Julia space are used to study the branched overlay structure of the map $f|_{J(f)}$.

Introduction

Let $X = (X, d)$ be a metric space. Given a continuous map $f\colon X \to X$, the iteration of f induces a *discrete semi-flow* on X denoted by (X, f). A point $x \in X$ is said to be a *fixed point* if $f(x) = x$; x is said to be a *periodic point* if there exists $n \in \mathbb{N}$, $n \neq 0$, such that $f^n(x) = x$ and x is said to be a *p-cyclic point* if $f^p(x) = x$ and $f^j(x) \neq x$ if $1 \leq j < p$.

Given a discrete semi-flow (X, f), a point $x_0 \in X$ is said to be a *Fatou point* if there is an open neighborhood U at x_0 verifying that for every $\epsilon > 0$ there is n_ϵ such that for every $x, y \in U$ and for every $n \geq n_\epsilon$, $d(f^n(x), f^n(y)) < \epsilon$. Denote $F(f)$ the open subset of all the Fatou points x_0 of X. The *Julia set* $J(f)$ is defined to be the closed subset $J(f) = X \setminus F(f)$.

Let $S^2 = \{(r_1, r_2, r_3) \in \mathbb{R}^3 \mid r_1^2 + r_2^2 + r_3^2 = 1\}$ be the unit 2-sphere, $\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$ the Alexandroff compactification of the space of complex numbers $\mathbb{C}$ and $\mathbf{P}^1(\mathbb{C})$ the complex projective line. The stereographic projection and the change from homogenous to absolute coordinates (explicit formulas can be seen in [5]) give the canonical isomorphisms $\tilde{\theta}\colon S^2 \to \hat{\mathbb{C}}$, $\theta\colon \mathbf{P}^1(\mathbb{C}) \to \hat{\mathbb{C}}$. We recall that a surface with a 1-dimensional complex structure is said to be a *Riemann surface* and a Riemann surface of genus 0 is said to be a *Riemann sphere*. Since $\mathbf{P}^1(\mathbb{C})$ has a canonical structure of 1-dimensional complex manifold, we can use the isomorphisms above to give to S^2 and $\hat{\mathbb{C}}$ the structure of a Riemann sphere. We also recall that, since S^2 is a subspace of $\mathbb{R}^3$, the usual Euclidean metric of $\mathbb{R}^3$ induces the Euclidean metric d on S^2, which is called the *chordal metric*. Using the isomorphisms $\tilde{\theta}, \theta$, one can translate these metric structures from S^2 to $\hat{\mathbb{C}}$ and $\mathbf{P}^1(\mathbb{C})$. In a similar way, the canonical Riemannian structure of S^2 can be translated to $\hat{\mathbb{C}}$ and $\mathbf{P}^1(\mathbb{C})$.

Let $\varphi\colon \mathbb{C} \to \mathbb{C}$ be a rational function of the form $\varphi(u) = a\,\dfrac{F(u)}{G(u)}$, where $a \in \mathbb{C}$, $a \neq 0$, $F(u), G(u) \in \mathbb{C}[u]$ are monic polynomials and F and G have not a common root. The complex function $\varphi(u) = a\frac{F(u)}{G(u)}$ has a canonical extension $\varphi^+\colon \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ applying the roots of $G(u) = 0$ to ∞ and $\varphi^+(\infty) \in \{\infty, 0, a\}$, where the value $\varphi^+(\infty)$ depends on the degrees of F and G. The isomorphism $\theta\colon \mathbf{P}^1(\mathbb{C}) \to \hat{\mathbb{C}}$ induces the map $g\colon \mathbf{P}^1(\mathbb{C}) \to \mathbf{P}^1(\mathbb{C})$

defined by $g = \theta^{-1}\varphi^{+}\theta$, which is expressed in homogeneous coordinates by the formula $g([z,t]) = [F_1(z,t), G_1(z,t)]$, where $F_1, G_1 \in \mathbb{C}[z,t]$ are homogeneous polynomials, with the same degree of homogeneity, such that $F_1(z,1) = F(z)$ and $G_1(z,1) = G(z)$, $z \in \mathbb{C}$. In a similar way, we can consider the isomorphism $\tilde{\theta}$ and the map $f = \tilde{\theta}^{-1}\varphi^{+}\tilde{\theta}\colon S^2 \to S^2$. In all cases, it is said that the maps $\varphi, \varphi^{+}, g, f$ are rational maps. Since $\tilde{\theta}$, θ are also isometries, the induced discrete semi-flows (S^2, f), $(\hat{\mathbb{C}}, \varphi^{+})$ and $(\mathbf{P}^1(\mathbb{C}), g)$ are isomorphic. We refer the reader to [1] for a study of the iteration of a rational map on the Riemann sphere.

The algebraic and geometric properties of these three models of Riemann sphere are used to develop some algorithms and implementations to study the topological and symmetry properties of the Julia compact metric space $J(f)$ induced by a rational map f. We are developing a research project about the study of the Julia set of rational maps and our first objective is to study the *shape* (see [10]) of the Julia set $J(f)$ by using a consecutive cubic subdivision procedure of the 2-sphere and the spherical multiplier $sm(f)$ which is defined in next section. The Julia compact metric space $J(f)$ is invariant by the action induced by f and there is an induced map $f|_{J(f)}\colon J(f) \to J(f)$. Our second objective is the study of the *branched d-overlay* structure (see [9]) of $f|_{J(f)}$ and its classification by a representation of the shape fundamental pro-group on the d-symmetric group, where d is the degree of the rational map f. Many of the symmetry properties of the fractal geometry of the Julia set can be expressed by this branched overlay structure and by the corresponding representation of its fundamental shape pro-group.

In this communication we will present some of the *computational algorithms and implementations* developed by our research team for the study of these objectives and we will also describe some lines of our computational research program directed to classify the shape of $J(f)$ and the branched overlay structure of $f|_{J(f)}$.

1. Spherical multipliers

Recall that if $(V, <,>_V)$ and $(W, <,>_W)$ are Euclidean vectorial spaces provided with an scalar product and the corresponding norm $\|v\|_V =< v, v >^{\frac{1}{2}}$, then the norm of a linear transformation $T : V \to W$ is defined by $\|T\| = \sup\{\|T(v)\|_W \mid \|v\|_V \leq 1\}$. Since S^2 has a canonical Riemannian structure, if $f\colon S^2 \to S^2$ is a rational function, one has that for a given point $x \in S^2$, there is an induced linear transformation $T_x(f)\colon T_x(S^2) \to T_{f(x)}(S^2)$ on the Euclidean tangent spaces at x and $f(x)$ of the Riemannian manifold S^2. Then, the *spherical multiplier* of a rational function $f\colon S^2 \to S^2$ at a point $x \in S^2$ is given by

$$sm(f)(x) = \|T_x(f)\|\,.$$

We remark the following facts: (i) the spherical multiplier is a bounded function from the 2-sphere to $\mathbb{R}$ (notice that the standard multiplier (see [1]) in general is not a bounded function); (ii) if x is a p-cyclic point, the spherical multiplier of f^p at x agrees with the absolute value of the standard multiplier of $(\varphi^{+})^p$ at $\tilde{\theta}(x) \in \hat{\mathbb{C}}$, $\varphi^{+} = \tilde{\theta} f (\tilde{\theta})^{-1}$.

Let $x \in S^2$ be a a p-cyclic point. If $sm(f)(x) = 0$ it is said that x is a *super-attracting point*; if $0 < sm(f)(x) < 1$, x is said to be an *attracting point*; if $sm(f)(x) = 1$, x is an *indifferent point* and when $sm(f)(x) > 1$, x is said to be a *repelling point*.

2. Algorithms for computing repelling cyclic points, shape of Julia sets and overlay structures

Our approach to the Julia set of a rational function is based in two different types of algorithms: the first one computes repelling p-cyclic points, $1 \leq p \leq n$ (the Julia set is the closure of the set of repelling cyclic points, see Theorem 6.9.2. of [1]); the second algorithm looks for points $x \in S^2$ such that $sm(f^p)(x) > 1$.

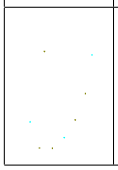
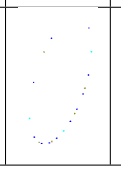
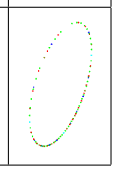
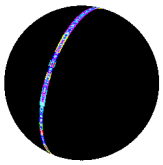
$n = 3$	$n = 4$	$n = 5$	$n = 6$	
				

Table 1. Spherical plots of $J(f)$ where $f(z) = \frac{2z}{-(1+3z^2)}$. On the left side, we have repelling p-cyclic points for $1 \leq p \leq n$, $n = 3, 4, 5, 6$. On the right side, a small neighborhood of $J(f)$ has been constructed as the union of small 2-spherical 2-cubes.

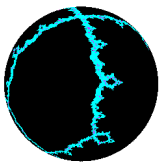
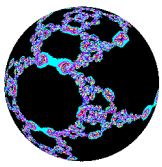
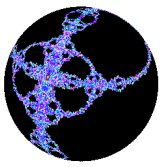
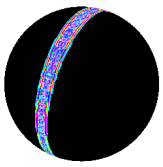
$h = 1.5$	$h = 2$	$h = 2.5$	$h = 3$
			

Table 2. Small neighborhoods (non black color) of the Julia set of the relaxed Newton's method applied to $z^3 + z$ for the values of the relaxing parameter $h \in \{1.5, 2, 2.5, 3\}$. Different values of h can determine different shapes of $J(f_h)$.

Table 1 contains on the left side the graphic output corresponding to four executions of the corresponding implemented function (of the first algorithm) for $n \in \{3, 4, 5, 6\}$. We can see that for $1 \leq p \leq n$ the repelling p-cyclic points are contained in a great circle of the unit sphere and therefore this suggests that the corresponding Julia set is this great circle. This algorithm has to deal with rational functions of high degree and this fact increases the execution time.

To avoid these difficulties, a second algorithm has been developed based on the spherical multiplier function: Given a point $x \in S^2$ and a rational function $f\colon S^2 \to S^2$, we can

compute the spherical multiplier by the formula:

$$sm(f^p)(x) = \|T_x(f^p)\| = \|T_x(f)\| \left\|T_{f(x)}(f)\right\| \cdots \left\|T_{f^{p-1}(x)}(f)\right\|.$$

Note that if x is a repelling p-cyclic point, then $\|T_x(f^p)\| > 1$ and this in-equality holds in a small neighborhood at x. Using this property of the spherical multiplier and subdivisions of the canonical cubic structure of the 2-sphere we can find neighborhoods of repelling p-cyclic points and also neighborhoods at the Julia set contained in S^2. On the right side of Table 1 we have a neighborhood of the Julia set. Taking an increasing sequence of values of p and new iterated subdivisions of the cubic structure of the 2-sphere we can construct an inverse system of cubic complexes $N_0 \supset N_1 \supset N_2 \supset \cdots$ such that $J(f) = \cap_{i=0}^{+\infty} N_i$ and $f(N_{i+1}) \subset N_i$. Up to homotopy type, $f|_{N_{i+1}}$ is a branched d-covering where d is the degree of f and each N_i has the homotopy type of a finite graph. In this way the map $f|_{J(f)}$ admits a branched d-overlay structure that can be obtained as a inverse limit of branched d-coverings of graphs (see [9]). The inverse system $\{N_i\}$ is a resolution of the compact metric space $J(f)$ and determines its shape type (see [10]). The family $\{N_i\}$ also gives an exterior structure to the Riemann sphere and the techniques and results developed in [2, 3] can also be applied.

We can see in Table 2 graphical outputs which correspond to neighborhoods of the Julia sets of rational functions f_h obtained when the relaxed Newton's method is applied to the polynomial $z^3 + z$ for the paramenter values $h = 1.5, 2, 2.5, 3$, respectively. The different colors are assigned depending on the values of the spherical multipliers $m(f_h^p)$; the black color corresponds to the region where the spherical multiplier is ≤ 1. For a description of these algorithms and its implementations we refer the reader to [4, 5, 6, 7, 8].

References

[1] A. F. Beardon, Iteration of rational functions, Springer-Verlag, New York, 2000.

[2] J. M. García-Calcines, L. J. Hernández and M. T. Rivas, Limit and end functors of dynamical systems via exterior spaces, Bull. Belg. Math. Soc.–Simon Stevin, 20 (2013), 937–959.

[3] J. M. García-Calcines, L. J. Hernández and M. T. Rivas, A completion construction for continuous dynamical systems, Topol. Methods Nonlinear Anal., 44 (2), (2014) 497–526.

[4] J. M. García-Calcines, J. M. Gutiérrez. L. J. Hernández and M. T. Rivas, Graphical Representations for the Homogeneous Bivariate Newton's Method, Appl. Math. Comput., 269, 988–1006 (2015). DOI: 10.1016/j.amc.2015.07.102

[5] J. M. Gutiérrez. L. J. Hernández, Á.A. Magreñán and M. T. Rivas, Measures of the basins of attracting n-cycles for the relaxed Newton's method, to appear, Springer, 2016.

[6] L. J. Hernández, Bivariate Newton-Raphson method and toroidal attraction basins, Numer. Algorithms, 71 (2), 349-38 (2016). DOI: 10.1007/s11075-015-9996-3

[7] L. J. Hernández, Plotting Basins of Univariate Rational Functions with Julia (User manual of the package PBURF03.jl), `https://www.researchgate.net/profile/Luis_Hernandez_Paricio/contributions`, `https://github.com/luisjavierhernandez/GVBURF03.jl`

[8] L. J. Hernández, M. Marañón and M. T. Rivas, Plotting basins of end points of rational maps with Sage, Tbilisi Math. J, 5 (2) (2012), 71–99.

[9] L. J. Hernández and V. Matijević, Fundamental groups and finite sheeted coverings, J. Pure Appl. Algebra, 214, (2010) 281–296.

[10] S. Mardesić, J. Segal, Shape Theory –The Inverse Limit Approach, North-Holland, 1982.

Departamento de Matemáticas y Computación, Universidad de La Rioja, Logroño, España.
E-mail address: {`luis-javier.hernandez,maria-teresa.rivas`}`@unirioja.es`

RECENT DEVELOPMENTS IN THE RUBI INTEGRATION PROJECT

DAVID J. JEFFREY AND ALBERT D. RICH

Abstract. RUBI (**RU**le **B**ased **I**ntegrator) is a long-term project that demonstrates the feasibility and desirability of organizing mathematical knowledge as a rule-based decision tree. The 'proof-of-concept' is provided by the development and implementation of a system of rules for finding optimal integrals (anti-derivatives or primitives) for a large class of integrands. RUBI demonstrates this through the evaluation of indefinite integrals. We give an overview of the project and its current status. We discuss different ways of implementing the project in different systems.

1. Introduction

The RUBI integration project has been running for a number of years, and now covers a large class of indefinite integration problems. A website has been created where all resources can be found [1]. The project has several aims, in addition to the obvious one of evaluating – symbolically – indefinite integrals, also known as primitives or anti-derivatives.

Within the context of integration, the project has the additional aim of finding the *simplest* or *optimal* primitive. An elementary example illustrates this.

The latest version of Maple (Maple 2016) produces the following result.

$$\int \frac{x^{10}}{(1+x)^{12}}\,\mathrm{d}x = -\frac{1}{11(1+x)^{11}} + \frac{1}{(1+x)^{10}} - \frac{5}{(1+x)^9} + \frac{15}{(1+x)^8} - \frac{30}{(1+x)^7} + \frac{42}{(1+x)^6} - \frac{42}{(1+x)^5} + \frac{30}{(1+x)^4} - \frac{15}{(1+x)^3} + \frac{5}{(1+x)^2} - \frac{1}{1+x}\,. \tag{1}$$

Mathematica obtains the same result, putting it over a common denominator:

$$-\frac{11x^{10}+55x^9+165x^8+330x^7+462x^6+462x^5+330x^4+165x^3+55x^2+11x+1}{11(x+1)^{11}}\,.$$

If the requirement is only that a system find *some* primitive, then both systems are satisfactory, but they both miss the eligant, optimal result

$$\int \frac{x^{10}}{(1+x)^{12}}\,\mathrm{d}x = \frac{x^{11}}{11(1+x)^{11}}\,.$$

This last result is from RUBI.

A significant, and continuing, part of the development of RUBI has been the search for the optimal form for each primitive. Considerations, such as shown in the above examples, have led to the construction of a test suite of over 55,000 integrands and their optimal primitives, and constitutes a significant resource in itself. It is freely available from the project website [1], where it can be found expressed in the syntaxes of several popular computer algebra systems.

Within the broader context of symbolic computation, the project has an additional aim, concerning the methods of computation. Computational methodologies in computer-algebra systems are frequently described as either based on term-rewriting (rule based), or computationally based. For example, MATHEMATICA is widely recognized as a rewrite language [2], whereas MAPLE is rarely described this way. The distinction is mostly one of emphasis, since all available systems include elements of both styles of programming. The dichotomy can be seen more specifically in programming to evaluate indefinite integrals. For symbolic integration, some of the best-known approaches are computationally based. For example, the Risch algorithm [8] and the Rothstein-Trager-Lazard-Rioboo algorithm [9, 10, 6] are both computational algorithms. These algorithms and others like them are not universally applicable, however, and for many integrals rule-based rewriting is needed and has advantages, some of which we discuss below. An important part of the operation of computer-algebra systems is the recognition of patterns. Mathematica provides powerful pattern-matching functions which are available to re-writing systems.

Serious doubts have been raised about the viability of large-scale term rewriting [3]. Earlier term rewriting schemes used various heuristics and back-tracking that is inherently inefficient and often leads to infinite recursion. However RUBI is totally deterministic, so we prefer to call it an algorithmic rule-based system.

Software that can display the steps of a calculation, variously called display step or single step, is very popular for pedagogical applications, as well as for debugging a system during its development. Examples of software that can display integration steps include WOLFRAM ALPHA and DERIVE, as well as many calculus tutorial programs. Rule-based systems such as RUBI can easily show steps by simply displaying a rule when it is applied, and then temporarily suspending evaluation to return and display the partially evaluated result.

The rule-based integration scheme that is considered here [7, 1] consists of a public-domain repository of precisely defined integration rules. The rules are expressed in human-readable standard mathematical notation, as well as machine-readable Mathematica program code. There are also utility files specific to various computer-algebra systems. The repository is *not* a table of integrals, such as one sees in the backs of calculus textbooks; rather it is the minimal set of reduction rules and terminal rules required to integrate a large class of mathematical expressions.

2. FORMAT FOR RULES

RUBI at present consists of over $6,000$ rules. Each entry in the repository has three functional parts, which together define a rule. The repository entries may contain other information, such as rule derivations and literature citations, which do not have a functional role.

(1) The transformation. This maps an integral to a mathematically equivalent expression which contains terms that are free of integrals and terms containing new (simpler) integrals. In the case of terminal transformations, the new expression is free of integrals.
(2) Validity conditions. Since the integrals in part 1 usually contain parameters, these conditions ensure the correctness of the transformation.

(3) Simplification conditions. These conditions ensure that the transformation is desirable, meaning that any new integral or integrals will lead, after further transformations, to a solution of the original problem.

An important design objective is that the conditions defining the rules are mutually exclusive, meaning that once parameters are specified for any integrand, only one set of conditions will evaluate to true, and therefore only one rule can be applied to any particular case.

3. Implementation of the rules

Allowing a particular system, such as Mathematica or Maple, to use the rules requires implementation. A first, straightforward way to proceed is to use the pattern-matching facilities that are offered by each system. Thus, for a particular system, Rubi becomes a list of system instructions, each of which has the following form. Suppose a rule applies to an integrand f, containing a set of parameters P. Each parameter $p_i \in P$ has validity rules v_{ij} and simplification conditions s_{ij}. Then for a given integrand G, each line of the program becomes, in pseudocode,

(1) Test whether G is an example of function f.
(2) If false, then go to next rule; if true then identify any numerical values given to the parameters in P.
(3) Test whether parameters that have values satisfy the validity rules v_{ij}.
(4) If false, then go to next rule; if true then test simplification conditions.
(5) If false, then go to next rule; if true then re-write expression using the transformation given in the rule.
(6) Recursively call the integration function again.

This implementation looks simple, but it has several disadvantages.

- It relies on the system having a strong pattern matching facility. Without naming individual systems, some systems do not have good pattern matchers.
- The system would perform a linear search through the list each time. Every computer scientist knows that a linear search is inefficient.
- From the point of view of software, or human, engineering, a linear list is very difficult to manage. For example, it is very difficult to ensure that there is an entry for every possible case. It becomes very tedious and error prone to check the completeness of the database.

4. Recent advances

The latest version of RUBI is shifting to a new implementation based on a decision-tree structure. This brings several advantages. The search changes from a linear search to a binary search, with a large gain in efficiency (initial testing indicates an increase of almost 2 orders of magnitude). In addition, the database of rules gains a more easily comprehended structure, which makes organising the rules easier, and makes it easier to check for completeness. If one is writing an `if ... then ... else` statement, then a missing `else` clause becomes obvious (glaringly obvious!).

It is important to appreciate that every time a new rule is added to the system, it is not simply a matter of inserting the rule at the appropriate place in the decision-tree. To

ensure that the new rule produces optimal results and does not conflict with existing rules, one must add to the test suite examples of integrals that test all aspects of the new rule, together with the optimal anti-derivatives as produced by the new rule. This is how the test suite has grown to over 55,000 entries.

Another new line of development is to replace dependence on generic pattern-matching functions in the various systems, with new code for analysing the structure of the integrand to be evaluated. This has allowed RUBI to be ported to more systems. During the presentation, Maple functions evaluating integrals directly in Maple will be demonstrated.

5. Conclusion

RUBI is an open source project, and readers are invited and encouraged to visit the web site and download the current system. Development of the system is active. As shown on the website, RUBI can already obtain optimal primitives for 99.8% of 55,000 test problems, but both the test suite itself and the rule-bank that is RUBI continue to expand and evolve.

References

[1] A. D. Rich. Rule based mathematics. Website: `www.apmaths.uwo.ca/~arich`

[2] B. Buchberger. Mathematica as a Rewrite Language. In T. Ida, A. Ohori, and M. Takeichi, editors, *Functional and Logic Programming (Proceedings of the 2nd Fuji International Workshop on Functional and Logic Programming, November 1-4, 1996, Shonan Village Center)*, pages 1–13. Copyright: World Scientific, Singapore - New Jersey - London - Hong Kong, 1996.

[3] Richard J. Fateman. A review of Mathematica. *J. Symb. Computation*, 13(5):545–579, 1992.

[4] D. J. Jeffrey. Integration to obtain expressions valid on domains of maximum extent. In Manuel Bronstein, editor, *ISSAC '93: Proceedings of the 1993 International Symposium on Symbolic and Algebraic Computation*, pages 34–41. ACM Press, 1993.

[5] D. J. Jeffrey. The importance of being continuous. *Mathematics Magazine*, 67:294–300, 1994.

[6] Daniel Lazard and Renaud Rioboo. Integration of rational functions: Rational computation of the logarithmic part. *J. Symb. Computation*, 9:113–115, 1990.

[7] A. D. Rich and D. J. Jeffrey. A knowledge repository for indefinite integration based on transformation rules. In *Intelligent Computer Mathematics*, volume 5625 of *LNCS*, pages 480–485. Springer, 2009.

[8] Robert H. Risch. The problem of integration in finite terms. *Trans. Amer. Math. Soc.*, 139:167–189, 1969.

[9] Michael Rothstein. A new algorithm for the integration of exponential and logarithmic functions. In *Proceedings of the 1977 MACSYMA users conference*, pages 263–274, 1977.

[10] Barry M. Trager. Algebraic factoring and rational function integration. In R. D. Jenks, editor, *Proceedings of the 1976 ACM Symposium on symbolic and algebraic computation SYMSAC '76*, pages 219–226. ACM Press, 1976.

Department of Applied Mathematics, University of Western Ontario, London, Canada
E-mail address: `djeffrey@uwo.ca`

TOWARDS A VERIFIABLE TOPOLOGY OF DATA

L. LAMBÁN, F. J. MARTÍN-MATEOS, J. RUBIO, AND J. L. RUIZ-REINA

Abstract. In this note we report on a project to get a verified ACL2 program to compute persistent homology, one of the tools for Topology of Data. We put attention both on generality (homology over a constructive elementary divisor ring) and performance (by using efficient ACL2 data structures).

Acknowledgements. This work has been partially supported by Ministerio de Economía y Competitividad, Spain, projects TIN2013-41086-P and MTM2014-54151.

Introduction

With the dramatic growing of Big Data applications, scientific computation has become an important focus of research and development interest. Among the techniques used to process big volume of data, one can find those coming from Topology [3], and more specifically persistent homology. These techniques, being based on well understood mathematical theories and methods, can produce, in principle, reliable outcomes. Nevertheless, in the path from algorithms to programs, some inaccuracies can occur; this is a common flaw with the rest of data processing techniques, making it difficult the reproducibility of computer experiments, and trusting the obtained results [6].

Thus it is interesting to build data processing programs verified by means of formal methods. Our objective is to write programs inside the ACL2 theorem prover [1] to compute persistent homology. This problem has been also tackled using the Coq system in [5], but there the emphasis was put in the formalization, whereas our approach looks for programs with a good performance (relative to other verified proposals).

Another distinguishing feature of our project is we attack the problem in its full generality: with matrices whose entries are elements from a elementary divisor ring. The main trend when computing persistent homology is in the context of field-valued matrices. There, a variation of Gaussian elimination provides very efficient incremental algorithms, and the results can be collected in an economic manner in so called *barcodes* [3]. With coefficients in a elementary divisor ring, things become harder and finding a good algorithm is trickier, implying that more time consuming programs are obtained. It is another good reason to look for efficiency in our ACL2 implementations.

In the next section, we explain an algorithm to compute a persistent homology group with coefficients in a elementary divisor ring. The essential bricks of the construction are two algorithms for matrix handling: Hermite and Smith normal forms. In Section 2 we report on the state of our project to get verified versions of these tools. The note ends with the bibliography.

1. An algorithm to compute persistent homology over elementary divisor rings

Given a filtered free chain complex C_*^* over a ring R (that is to say, each C_n^i, where i is the filtration index and n the degree or dimension, is a free R-module with an explicit finite basis, and the filtration respects those bases), to compute its persistent homology groups amounts to calculate a number of standard homology groups (more concretely, $H_n^{i,j}(C_*^*) = Im(H_n(C^i) \to H_n(C^j))$, where $H_n(C^i) \to H_n(C^j)$ is the arrow induced by the filtration between the standard homology groups; so, they are image groups). If R is an elementary divisor ring, each coordinate matrix can be transformed into its Smith normal form, and this allows computing kernels, images and homology groups. In particular, if we know an algorithm to compute the Smith normal form of any matrix, we can determine the persistent homology. The drawback of this approach, without using the filtration process, is that we are re-processing many times the same sub-matrices, without reusing any of the previously performed manipulations. In addition, it is well-known that some implementations of the Smith normal form (even if it is polynomial as for the number of arithmetic operations) can exhibit an exponential behavior, due to the growing of the size of intermediary entries.

Therefore, our idea is to proceed carefully, using as pre-processing a Hermite step (a Hermite algorithm gets a triangular matrix, in an echelon form, by only performing column operations), in such a way that the eventual final Smith normal form occurs on small sub-matrices, and the Hermite step is common for different groups.

With more detail, to compute the groups $H_n^{i,j}$, with $j \geq i$, it is necessary to study the coordinate matrices (with respect to the explicit bases of the filtered chain complex) of the differential maps d_n^k (as before, subindexes refer to degrees in the chain complex, and superindexes to indexes in the filtration). Since the bases are compatible with the filtration, we can collect, for each degree n, all the coordinate matrices corresponding to d_n^k, for all k, producing a unique "echelon matrix" denoted by M_n (let us observe that the image of each d_n^k is inside of C_*^k given to M_n an echelon form according the filtration indexes). The first step when computing $H_n^{i,j}$ consists of obtaining a basis of C_n^* which contains compatible bases for all the kernels $Z_n^i = \mathrm{Ker} d_n^i$. This calculation can be carried out by means of an algorithm computing a Hermite matrix equivalent to M_n that respects the filtration indexes. To this aim, we can apply, index by index, Bézout operations and permutations of the columns of M_n. Once this step has been achieved, we perform the corresponding operations on the rows of M_{n+1} (in order to take into account the basis exchange made in C_n^*), and we get a new coordinate matrix M'_{n+1} of d_{n+1}^*, where the non-null rows correspond to members of the basis of Z_n^i computed in the first step.

The difficulty is now to find a matrix equivalent to M'_{n+1} such that some submatrices are presentations of the groups $H_n^{i,j}$; in other words, we have to find relations characterising the image subgroups defining the persistent homology.

Let us settle firstly a couple $i \leq j$, and focus on $H_n^{i,j}$. We start from M'_{n+1} keeping only the columns up to index j; they form a generating system for the image $\mathrm{Im} d_{n+1}^j$, expressed in terms of the computed basis of Z_n^j. Our objective is then to determine $\mathrm{Im} d_{n+1}^j \cap Z_n^i$. For this purpose, it is enough to get a Hermite matrix equivalent to the submatrix consisting of the columns from index $i+1$ to index j. In fact, it is sufficient to obtain a Hermite matrix

equivalent to the submatrix going from the rows of biggest index until reaching the rows corresponding to Z_n^i. In order to finish the calculation of $H_n^{i,j}$, we compute the Smith normal form of the submatrix defined by the rows until the index i and the columns of height less or equal than the cardinality of the basis of Z_n^i.

Let us stress that the whole described process, except the final Smith normal form step, can be performed in an incremental way (without repeating calculations). Thus, it is possible to get jointly $H_n^{i,j}$, for all $j \geq i$ (keeping apart the Smith normal form step that, let us repeat it, must be done in an ad-hoc manner). Indeed, only one process can be used to compute $H_n^{i',j}$, for all $i \leq i' \leq j$. The bottleneck of the method is in the computation of the Smith normal form, where an incremental (by filtration index) algorithm does not prevent re-computations.

2. Towards an ACL2 program to compute persistent homology

As it is explained in the previous section, the two main building blocks to compute persistent homology are Hermite and Smith normal forms programs. As a first step, we have implemented in ACL2 algorithms for these two normal forms, having in mind that our goals are both formal verification and a good performance. For these two implementations, we have also used different data structures to represent matrices: list of lists for the Smith normal form and arrays for the Hermite normal form. These two different approaches have a strong influence both in the efficiency of the algorithm and in the difficulty of the formal verification, as we explain now.

In principle, the ACL2 programming language is an extension of an applicative subset of Common Lisp, and the logic of its prover describes it. This means, among other implications, that the only way to build a non-atomic data is by means of *conses* (lists), and that accesses and updates to lists are basically done via `nth` and `update-nth`, respectively, which take time linear in the length of the lists. Also, updates are not destructive and need copying, following the usual *copy-on-write* applicative semantics.

Taking this into account, a natural representation in ACL2 for bidimensional matrices is by means of the list of its rows, and each row by the list of its elements. For example, `((1 0 0 0) (0 1 0 0) (0 0 1 0) (0 0 0 1))` is the representation of the identity matrix of size 4. This is known as the *list of lists* representation and it is a convenient representation for reasoning in ACL2 about matrices, for the reasons described above. Nevertheless, from the point of view of efficiency of execution, this representation is impractical: it is essential to be able to access and update in constant time, and also to update *in-place*. Fortunately, this problem can be solved in ACL2 using *arrays* [1]. Under some syntactic restrictions that preserve the applicative semantics, ACL2 arrays allows accesses and updates in constant time. From the logic point of view, an array is an association list, but for execution, Common Lisp arrays are used.

To compare both approaches, we have implemented the Smith normal form computation using lists of lists, and the Hermite normal form using bidimensional arrays. We have applied them to compute the determinant (up to a unit of the ring) of a number of integer matrices, comparing the time and space taken in each case. For example, for matrices of size 30×30, the performance with the array representation is much better than with the list of lists representation, both in time and space. For matrices of size 40×40, usually the list of list

representation is so time and space consuming that we were not able to obtain a result. Nevertheless, the array representation returns the result in a reasonable time.

We used these programs to compute the determinant of a concrete matrix, famous because some versions of the Mathematica scientific computing software computes it wrongly; see [2] for details. Both programs can compute (rightly!) that determinant, but their performance differ, in the same line as explained above:

```
ACL2 !>>; (EV-REC (smith varona-matrix))
; 374.29 seconds realtime, 374.51 seconds runtime
; (647,212,775,072 bytes allocated).
ACL2 !>>; (EV-REC (hermite varona-matrix))
; 58.30 seconds realtime, 58.23 seconds runtime
; (94,705,808,368 bytes allocated).
```

Let us note that the comparison is totally unfair, because the Smith normal form is more expensive, as an algorithm, than the Hermite one. Nevertheless, our experimental study shows that much of the poor performance of Smith is due to the chosen representation.

As we have seen, we have here the usual trade-off between a suitable representation for reasoning and another one for execution efficiency. From ACL2 version 5.0, a new feature was introduced, that allows dealing with this issue: *abstract stobjs* [4]. Single-threaded objects (*stobjs*) in ACL2 are an alternative way, different from arrays, that allows implementation of mutable objects, with applicative semantics for reasoning, but with accesses and updates in constant time. Abstract stobjs are a way to provide an interface between a convenient representation for reasoning, and an efficient one for execution, that could be in principle completely different, but whose equivalence is supported by the system if some correspondence theorems are previously proved. Following the ideas already carried out in [7] for the case of Gaussian elimination, our next step is to build an efficient implementation of Smith and Hermite normal forms based on single threaded objects and to formally verify them.

References

[1] ACL2 User Manual. `http://www.cs.utexas.edu/users/moore/acl2/v7-2/combined-manual`

[2] A. J. Durán, M. Pérez, J. L. Varona, *Misfortunes of a mathematicians' trio using Computer Algebra Systems: Can we trust?*, Notices of the AMS 61 (10) (2014) 1249–1252.

[3] H. Edelsbrunner, J. Harer, *Computational Topology: An Introduction*, Applied Mathematics, American Mathematical Society, 2010.

[4] S. Goel, W. A. Hunt Jr., Matt Kaufmann, *Abstract Stobjs and Their Application to ISA Modeling*, Proceedings International Workshop on the ACL2 Theorem Prover and its Applications, ACL2 2013 (2013) 54–69.

[5] J. Heras, T. Coquand, A. Mörtberg, V. Siles, *Computing Persistent Homology within Coq/SSReflect*, ACM Transactions on Computational Logic 14 (4) (2013), paper 26.

[6] R. D. Peng, *Reproducible Research in Computational Science*, Science 334 (6060) (2011) 1226–1227.

[7] J. L. Pro-Martín, J. R. Ruiz-Reina, F. J. Martín-Mateos, *Formalization in ACL2 of Matrix Algebra Basic Concepts*, Proceedings ESCIM 2015, Universidad de Cádiz (2015) 137–142.

Departamento de Matemáticas y Computación. Universidad de La Rioja
E-mail address: `lalamban,julio.rubio@unirioja.es`

Departamento de Ciencias de la Computación e Inteligencia Artificial. Universidad de Sevilla
E-mail address: `fjesus,jruiz@us.es`

A NUMERIC-SYMBOLIC ALGORITHM FOR COMPUTING THE LIOUVILLIAN SOLUTIONS OF DIFFERENTIAL EQUATIONS AND SYSTEMS

ALBERTO LLORENTE AND JORGE MOZO-FERNÁNDEZ

Abstract. We give an algorithm for deciding if an explicitable higher-order system of linear differential equations over the complex rational functions, symbolically given, admits non-null Liouvillian solutions, computing symbolically one in the positive case, by numeric-symbolic methods in the sense of J. van der Hoeven.

1. Introduction

We present the work of the PhD thesis [4] of the first author, defended in 2014 and advised by the second author. An advance of the progress of this work was presented at ISSAC 2012 [5], so we shall highlight the differences with the final version.

The aim of this work is giving a numeric-symbolic algorithm for finding the Liouvillian solutions of a linear system of differential equations

$$\mathtt{A}_0(x)\,\mathbf{y} + \mathtt{A}_1(x)\,\mathbf{y}' + \cdots + \mathtt{A}_r(x)\,\mathbf{y}^{(r)} = \mathbf{0} \tag{1}$$

with $\mathtt{A}_0(x), \mathtt{A}_1(x), \ldots, \mathtt{A}_1(x) \in \mathbb{C}(x)^{n\times n}$ and $\det \mathtt{A}_r(x) \not\equiv 0$. Here *numeric-symbolic* means a numerical stage followed by a symbolic stage for a symbolic correct output.

2. Foundations

Cyclic Vector Lemma allows to extend a theorem of Singer [6] to systems.

Theorem 2.1. *There exists a function $I : \mathbb{N} \to \mathbb{N}$ such that, if (1) has a non-zero Liouvillian solution, then there exist an algebraic extension $F/\mathbb{C}(x)$ of degree $I(rn)$ at most and a non-zero solution $(y_1, y_2, \ldots, y_n)^\intercal$ of (1) and, for each i and j with $y_i \neq 0$, $y_i'/y_i \in F$ and $y_j/y_i \in F$.*

Remark 2.2. Recent group-theoretical results [1, 2] allow us to take as I the function $I(k) = (k+1)!$ for $k \geq 14$, with a table of values for $k < 14$ that appears in the first author's thesis. These values can be further reduced, according to ongoing research, which will be useful in the practical setting.

Following the ideas of van der Hoeven [9], we use differential Galois theory and effective complex numerics. Differential Galois theory associates to (1) an algebraic group of matrices G, with identity component G° and Lie algebra $\mathfrak{g}$, such that a solution of Theorem 2.1 is a common eigenvector of G° (and, equivalently, of $\mathfrak{g}$) whose images by G fall in $I(rn)$ lines at most. The differential Galois group G is the Zariski closure of the group generated by certain groups: the monodromy, the exponential tori and the Stokes matrices. These groups are computable in effective complex numerics. The computation of the monodromy uses [7];

the exponential tori and the Stokes matrices required [10] in [9], but in this work we avoid their full computation, in such a way that the techniques in [7] are enough, since we only need the solutions of Theorem 2.1.

The effective complex numerics are introduced (in the real case) in [8]. These numbers are a black box with input $\varepsilon > 0$ and output an approximation within ε of the complex number represented. Effective complex numbers can be added, multiplied and, if the denominator is bounded from below, divided. Also roots of polynomials with effective complex coefficients are effective, so these numbers form an algebraically closed field. What cannot be computed in finite time is whether an effective complex number is zero, so we substitute $|a| < \mathsf{tol}$ for $a = 0$ when needed, for certain global variable tol. This introduces certain controlled error in linear algebra: the rank of a set of vectors may be undercomputed, but never overcomputed, and, for tol small enough, the computation is exact.

Derksen [3] and van der Hoeven [9] gave an algorithm for computing the Zariski closure of a finitely generated group, but this algorithm requires computing the multiplicative syzygies of the eigenvalues, a problem harder than checking if an eigenvalue is a root of unity, which is insolvable in finite time because the roots of unity are dense and codense in the unit circle. In order to make the Derksen–van der Hoeven algorithm effective, we propose waiving the objective of the Zariski closure and to compute a greater closure, the eurymeric closure, to be defined.

Definition 2.3. A *eurymeric group* is a Zariski-closed group of invertible matrices whose Lie algebra is closed under the product of matrices and contains the identity. The *eurymeric closure* of G is the smallest eurymeric group H containing G.

Eurymeric groups have good properties that linearize some non-linear steps in the Derksen–van der Hoeven algorithm, as reviewed in Section 3, reducing the computation of all the multiplicative syzygies of the eigenvalues to testing if any quotient of the eigenvalues is a root of unity. This is still insolvable in effective complex numerics, but we can truncate the process (essentially, truncate the Euclidean algorithm) when we get that, in case of being a root of unity, its order would be greater than $I(rn)$. Thus we compute an augmentation H_0 of H, with Lie algebra $\mathfrak{h}_0$, and we can prove that it keeps the solutions of Theorem 2.1: if $\mathfrak{g}$ has nonzero common eigenvectors, then $\mathfrak{h}_0$ has nonzero common eigenvectors and they are common eigenvectors of $\mathfrak{g}$.

3. The algorithm for computing the eurymeric closure

For computing the eurymeric group generated by finitely many matrices, we need to know how to compute the eurymeric group generated by a single matrix.

Definition 3.1. Let $\mathtt{A} \in \mathrm{GL}(k, \mathbb{C})$ have eigenvalues λ_i. For any quotient λ_i/λ_j that is a root of unity, we consider its order. The least common multiple of these orders and 1 will be called the *resonance order* of $\mathtt{A}$.

Theorem 3.2. *Let* $\mathtt{A} \in \mathrm{GL}(k, \mathbb{C})$ *have resonance order* p*. The smallest eurymeric group* E *containing* $\mathtt{A}$ *has the Lie algebra* $\mathbb{C}[\mathtt{A}^p]$*, and* $\{\mathtt{I}, \mathtt{A}, \ldots, \mathtt{A}^{p-1}\}$ *is a system of representatives of* E/E°*.*

Let $\mathcal{M}$ be a finite set of matrices, E the eurymeric group they generate and $\mathfrak{e}$ its Lie algebra. The algorithm works with a finite family $\mathcal{F}$, initially $\mathcal{M}$, and a Lie algebra $\mathfrak{a}$, initially $\mathbb{C}\mathtt{I}$, which are modified by the following loop until they stabilize.

(1) For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as an algebra, with the Lie algebra $\mathfrak{b}$ of the eurymeric group generated by $\mathtt{A}$; i.e. $\mathfrak{a} := \mathfrak{a}[\mathfrak{b}]$.
(2) For each $\mathtt{A} \in \mathcal{F}$ we augment $\mathfrak{a}$, as an algebra, with $\mathtt{A}\mathfrak{a}\mathtt{A}^{-1}$; i.e. $\mathfrak{a} := \mathfrak{a}[\mathtt{A}\mathfrak{a}\mathtt{A}^{-1}]$.
(3) For each $\mathtt{A} \in \mathcal{F}$ we check if $\mathtt{A}$ is equivalent to any element of $\mathcal{F}$ modulo the connected group corresponding to $\mathfrak{a}$, which is $\mathfrak{a} \cap \mathrm{GL}(nr, \mathbb{C})$; if it is, we eliminate $\mathtt{A}$ from $\mathcal{F}$.
(4) For each ordered pair $\mathtt{A}, \mathtt{B} \in \mathcal{F}$ we check if $\mathtt{AB}$ is equivalent to any element of $\mathcal{F}$ modulo $\mathfrak{a} \cap \mathrm{GL}(nr, \mathbb{C})$; if it is not, we add $\mathtt{AB}$ to $\mathcal{F}$.

Theorem 3.3. *The loop stabilizes and the algorithm terminates with $\mathfrak{a} = \mathfrak{e}$ and $\mathcal{F}$ a system of representatives of E/E°.*

We can avoid computations of $\mathbb{Z}$-bases for the exponential tori by precomputing their respective eurymeric closure, which are other algebraic tori but with simpler generators.

4. News on the main algorithm

The errors coming from linear algebra computations, for tol too large, might undercompute H_0, and hence overcompute the common eigenvectors of $\mathfrak{h}_0$, but for tol small enough the computation is exact. As the termination of Derksen–van der Hoeven algorithm in [9] is proved only for exact computations, for its linearized version for computing the eurymeric group we use a global variable G as a truncation parameter bounding how many times the loop is repeated. Let H_* be the computed version of H_0, and $\mathfrak{h}_*$ its Lie algebra. We choose a common eigenvector of $\mathfrak{h}_*$, in such a determined way that the same Lie algebra yields always the same eigenvector, and compute its orbit by H_*. Then we reconstruct symbolically the solutions and check if they are solutions of (1) in the way explained below.

A solution of Theorem 2.1 is represented by a Darboux polynomial, described in [11], easy to check without computing symmetric powers. The symbolic reconstruction of the rational functions in the Darboux polynomial is done by the methods exposed in [9]. The coefficients of the Darboux polynomial are rational functions and they are reconstructed using Padé approximation and a global variable B as the bound for the degree of the numerator and the denominator. The coefficients of the aforesaid numerators and denominators are granted to be algebraic over the constants in (1) and van der Hoeven proposes reconstructing them using the LLL algorithm, but we add to this method the HJLS and the PSLQ algorithms. These three methods look for additive syzygies that yield an annihilating polynomial of the constant to reconstruct, using two global variables D and size as bounds for the degree of the annihilating polynomial and for the size of the syzygy, respectively. Finally, we check if the reconstructed Darboux polynomial is actually Darboux, and it is also checked with the Brill equations in order to know if it could come from a solution of Theorem 2.1.

If we find solutions, we have succeeded. If we find that there is no common eigenvector of $\mathfrak{h}_*$, we say that zero is the only Liouvillian solution of (1). If there are common eigenvectors but our candidate solutions are wrong, then we have computed with tol too large or the rest of the global variables too small, so we restart the algorithm with finer values of the global variables. In [5], as reporting ongoing research, we thought it could be enough to

refine jointly all the global variables so that the successive values of tol converges to zero and the successive values of the rest of the global variables diverge to infinity, but such an implementation is not proved to terminate. The first author's thesis fixes this issue by giving a scheme for the successive values of the global variables, in such a way that the computations will be eventually exact and the algorithm terminates with a correct output.

5. Conclusion

The main result of this work is the following.

Theorem 5.1. *The algorithm introduced above terminates with a non-zero Liouvillian solution, if such a solution exists, or with the statement that zero is the only Liouvillian solution if this is the case.*

The algorithm avoids computing the symmetric powers and it may end early when the only Liouvillian solution is zero, an advantage over the usual methods. The algorithm also works directly with systems, without converting them into scalar equations.

References

[1] Michael J. Collins. «On Jordan's theorem for complex linear groups.» *Journal of Group Theory*, 10(4):411–423, 2007.

[2] Michael J. Collins. «Bounds for finite primitive complex linear groups.» *Journal of Algebra*, 319(2):759–776, 2008.

[3] Harm Derksen, Emmanuel Jeandel, and Pascal Koiran. «Quantum automata and algebraic groups.» *Journal of Symbolic Computation*, 39(3-4):357–371, 2005.

[4] Alberto Llorente. *Métodos numérico-simbólicos para calcular soluciones liouvillianas de ecuaciones diferenciales lineales*. PhD thesis, Universidad de Valladolid, 2014. Main chapters in English.

[5] Alberto Llorente, and Jorge Mozo-Fernández. «Numeric-symbolic methods for computing the Liouvillian solutions of differential equations and systems.» *ACM Communications in Computer Algebra*, 46(3):112–113, 2012.

[6] Michael F. Singer. «Liouvillian solutions of n-th order homogeneous linear differential equations.» *American Journal of Mathematics*, 103(4):661–682, 1981.

[7] Joris van der Hoeven. «Fast evaluation of holonomic functions.» *Theoretical Computer Science*, 210(1):199–215, 1999.

[8] Joris van der Hoeven. «Computations with effective real numbers.» *Theoretical Computer Science*, 351(1):52–60, 2006.

[9] Joris van der Hoeven. «Around the numeric-symbolic computation of differential Galois groups.» *Journal of Symbolic Computation*, 42(1-2):236–264, 2007.

[10] Joris van der Hoeven. «Efficient accelero-summation of holonomic functions.» *Journal of Symbolic Computation*, 42(4):389–428, 2007.

[11] Jacques-Arthur Weil. *Constantes et polynômes de Darboux en algèbre différentielle: applications aux systèmes différentiels linéaires*. PhD thesis, École Polytechnique, 1995.

Universidad Isabel I, www.ui1.es, Calle Fernán González n° 76, 09003 Burgos, Spain
E-mail address: alberto.llorente@ui1.es, llorente.alberto@gmail.com

Universidad de Valladolid, Dpto. Álgebra, Análisis Matemático, Geometría y Topología, Facultad de Ciencias, Campus Miguel Delibes, Paseo de Belén n° 7, 47011 Valladolid, Spain
E-mail address: jmozo@maf.uva.es

GENERATORS OF MULTIPLE FAILURE IDEALS OF k-OUT-OF-n AND CONSECUTIVE k-OUT-OF-n SYSTEMS

F. MOHAMMADI, E. SÁENZ-DE-CABEZÓN, AND H. WYNN

Abstract. We give an explicit description of the lcm-filtration of the failure ideal of k-out-of-n and consecutive k-out-of-n systems. Based on this description we implement efficient algorithms for the computation of the reliability of such systems under the assumption of multiple simultaneous failures.

Introduction

Given a monomial ideal I generated by r monomials, the i-fold lcm ideal of I, denoted I_i, is generated by all the least common multiples of i generators of I. These ideals, for all $1 \leq i \leq r$ constitute a filtration of I, which we call the lcm-filtration. This filtration provides insight on the structure of the ideal, giving more homological information than the usual Betti numbers cf. [3]

If $I = I_S$ is the failure ideal of a coherent system S then the generators of I_i correspond to the ideal of the system that fails whenever there are i simultaneous failures of the original system S. This allows us to use commutative algebra techniques to study simultaneous failures in coherent systems. This extends and generalizes previous applications of computational commutative algebra to reliability theory cf. [5, 6]

A major difficulty of this approach is that the number of generators of I_i is $\binom{r}{i}$ which grows exponentially and makes this method impractical for large r. In many cases, most of these generators are redundant and one must perform a reduction of the generating set of I_i, which makes the procedure more time demanding. Besides, a direct construction of I_i would profit of the fact that the minimal generating set consists of less than $\binom{r}{i}$ generators.

In this contribution we focus on two of the most important systems in industry, namely k-out-of-n and consecutive k-out-of-n systems, cf. [1]. Both examples give rise to square free ideals, which are somewhat easier to handle. The structures of the lcm-filtrations of the failure ideals of these two kinds of systems are very different, as was observed in [2]. We give explicit descriptions of the generating sets of their multiple failure ideals I_i for all i, which allows us not only to rigorously describe the structure of their lcm-filtrations but also to implement efficient algorithms that compute their minimal generating sets.

1. The lcm-filtration of k-out-of-n systems

A k-out-of-n system is a system with n components that fails whenever k of them fail. Let $S_{k,n}$ be a k-out-of-n system. The failure ideal of $S_{k,n}$ is given by $I_{k,n} = \langle \prod_{i\in\sigma} x_i | \sigma \subseteq \{1,\ldots,n\}, |\sigma| = k\rangle$. Let $I_{k,n,i}$ be the i-fold lcm-ideal of $I_{k,n}$.

Theorem 1.1. *Let $k < j \leq n$. For all $\binom{j-1}{k} < i \leq \binom{j}{k}$ we have that $I_{k,n,i} = \langle \prod_{s\in\sigma} x_s | \sigma \subseteq \{1,\ldots,n\}, |\sigma| = j\rangle = I_{j,n}$.*

Proof. Let $\mathcal{S}^k$ be the set of subsets of $\{1,\ldots,n\}$ of cardinality k. For all $k < i \leq n$. For every $\sigma \subset \{1,\ldots,n\}$ of cardinality i we have that there are $\binom{i}{k}$ subsets of cardinality k the union of which forms σ, i.e. taking the union of more than $\binom{i}{k}$ elements of $\mathcal{S}^k$ we always get a set of cardinality strictly bigger than i. On the other hand, there are always $i-k+1$ subsets of $\mathcal{S}^k$ the union of which is σ. To see this just choose any k elements of σ, let us call this subset σ_1 we can now choose $i-k$ new elements of $\mathcal{S}^k$ each of which contributes with a new variable to build σ. Observe that in fact we only need $\lceil \frac{i}{k} \rceil$ elements of $\mathcal{S}^k$. □

The fact that the ideals in the lcm-filtration of $I_{k,n}$ correspond to the failure ideals of other systems of the same type allows us to make use of the known formulas for the Betti numbers of these systems, which can be found for instance in [4]. Hence we can obtain all the Betti numbers of the simultaneous failure ideals for k-out-of-n ideals.

Corollary 1.2. *The Betti numbers of $I_{k,n,i}$ are given by*

$$\beta_a(I_{k,n,i}) = \beta_a(I_{j,n}) = \binom{n}{j+a}\binom{a+j-1}{j-1}$$

for all $\binom{j-1}{k} < i \leq \binom{j}{k}$.

2. The lcm-filtration of consecutive k-out-of-n systems

Let us now consider a consecutive k-out-of-n linear system. The ideal of $\mathcal{S}_{k,n}$ is given by $J_{k,n} = \langle x_1\cdots x_k, x_2\cdots x_{k+1}, \ldots, x_{n-k+1}\cdots x_n\rangle = \langle m_1, m_2, \ldots, m_{n-k+1}\rangle$. Let $J_{k,n,i}$ be the i-fold lcm-ideal of $J_{k,n}$, which is generated by the monomials corresponding to i simultaneous minimal failures of $S_{k,n}$.

Let us denote by $\mathcal{S}$ the set of subsets of $\{1,\ldots,n-k+1\}$, and let $\mathcal{S}^i$ the elements of $\mathcal{S}$ of cardinality i. Let $\sigma \subseteq \{1,\ldots,n-k+1\}$. We say that σ has a gap of size s if there is a subset of s consecutive elements of $\{\min(\sigma),\ldots,\max(\sigma)\}$ that are not in σ. Let $\mathcal{S}_a$ be the set of subsets σ of $\{1,\ldots,n-k+1\}$ such that the smallest gap in σ has size a. Let $\mathcal{S}_a^i$ be the elements in $\mathcal{S}_a$ of cardinality i.

Theorem 2.1. *For any subset $\sigma \subseteq \{1,\ldots,r\}$ we denote m_σ the monomial $m_\sigma = \mathrm{lcm}(\{m_i | i \in \sigma\})$. $J_{k,n,i}$ is minimally generated by the monomials m_σ such that $\sigma \in \mathcal{S}_0^i \cup \mathcal{S}_k^i \cup \mathcal{S}_{k+1}^i \cup \cdots \cup \mathcal{S}_{n-k+1}^i$ i.e. the minimal generators of $J_{k,n,i}$ corresponds to the* lcm*'s of sets of monomials of cardinality i with no gaps of sizes between 1 and $k-1$ both included.*

Proof. We know that the set $\{m_\sigma$ such that $\sigma \subseteq \{1,\ldots,n-k+1\}, |\sigma| = i\}$ is a generating set of $J_{n,k,i}$. Let $G_{n,k,i}$ be the minimal set of generators of $J_{n,k,i}$. To prove the theorem we must show that

i) $\mathcal{S}_0^i \subseteq G_{n,k,i}$
ii) If $\sigma \in \mathcal{S}_a^i$ with $a \leq k$ then there is another element τ in $\mathcal{S}_b^i$ with $b = 0$ or $b > a$ that m_τ divides m_σ.
iii) If $\sigma \in \mathcal{S}_a^i$ with $a \geq k+1$ then $m_\sigma \in G_{n,k,i}$.

To see i) observe that if $\sigma \in \mathcal{S}_0^i$ then m_σ is the lcm of i consecutive minimal generators of $I_{k,m}$, say $m_a, \ldots, m_{a+i}$ then $m_\sigma = x_a \cdots x_{a+i+k-1}$ hence it is a product of $i+k-1$ variables. Using i different subsets of $\{1, \ldots, n-k+1\}$ we obtain at least $i+k-1$ variables, and we can only achieve this minimum if the i elements of σ are consecutive, i.e. if we are in $\mathcal{S}_0^i$. Hence, if there is another $\sigma' \in \mathcal{S}^i$ that divides σ then it must be σ itself for any other element of $\mathcal{S}_0^i$ produces a different set of variables and would not divide m_σ.

Now, to see ii) let us assume without loss of generality that σ is formed by two blocks σ_1 and σ_2 of consecutive elements, separated by a gap of size a with $a \leq k$. For ease of notation we can say $\sigma_1 = \{1, \ldots, i_1\}$ and $\sigma_2 = \{i_1+a, \ldots, i_2\}$. We have then that $m_\sigma = x_1 \cdots x_{i_1+k-1} \cdot x_{i_1+a} \cdots x_{i_2+k-1}$. But now, since $a \leq k$ we have that $\sigma' = \{1, \ldots, i_1+1\} \cup \{i_1+a+1, \ldots, i_2\}$ is in $\mathcal{S}_b^i$ with $b \leq a$ (in fact it is either in $\mathcal{S}_0$ or in $\mathcal{S}_a$) and $m_{\sigma'}$ divides m_σ. Repeating this procedure, we see that $m_{\{1,\ldots,i\}}$ divides m_σ and hence m_σ is not a minimal generator of $I_{n,k,i}$. Observe that $\{1, \ldots, i\} \in \mathcal{S}_0$. Observe that we can also proceed in a symmetric way, i.e. using $\sigma' = \{1, \ldots, i_1-1\} \cup \{i_1+a-1, \ldots, i_2\}$ in $\mathcal{S}_b^i$ and we would obtain another element in $\mathcal{S}_0^i$ that divides σ.

Finally, to see iii) let us assume again, without loss of generality, that $\sigma_1 = \{1, \ldots, i_1\}$ and $\sigma_2 = \{i_1+a, \ldots, i_2\}$ with $a \geq k+1$. Now, if $\tau \in \mathcal{S}^i$ is different from σ then there is at least one index τ_j such that $\tau_j \neq \sigma$ if $\tau_j > i_2$ and there exists at least a variable bigger than x_{i_2+k-1} in τ which is then not in σ and hence m_τ does not divide m_σ. The same happens if $i_1 < j < i_1 + a$: since $a \geq k+1$ every m_j with $i_1 < j < i_1 + a$ has a variable that does not belong to $\{x_1, \ldots, x_{i_1+k-1}\} \cup \{x_{i_1+a}, \ldots, x_{i_2+k-1}\}$. To see this observe that $i_1 + a > i_i + k - 1$ and every m_j with $i_1 < j < i_1 + a$ has at least one variable in the nonempty set $\{x_{i_1+1}, \ldots, x_{i_1+a-1}\}$. Hence τ does not divide σ and σ is in $G_{n,k,i}$. □

Remark 2.2. Let $\tau \subseteq \{1, \ldots, n\}$ such that it is formed by blocks of (decreasing) sizes $\tau_1, \tau_2 \ldots, \tau_l$ then m_τ has degree $\tau_1 + 1 + \tau_2 + 1 + \cdots + \tau_l + 1$. We say that the *pattern* of τ is $\tau_1, \tau_2 \ldots, \tau_l$.

Example 2.3. $I_{2,9,4}$ is minimally generated by the 26 monomials that correspond to taking lcm's of the following sets of generators of $I_{2,9}$. Observe that this ideal is generated by 8 generators in 9 variables. Each of the sets in the central column of the table refers to the set of generators of which we compute the least common multiple, e.g. 2345 means $\mathrm{lcm}(m_2, m_3, m_4, m_5)$.

Pattern	sets	deg. of generators
4	1234,2345,3456,4567,5678	5
3,1	1236,1237,1238,2347,2348,3458,1456,1567,2567,1678,2678,3678	6
2,2	1256,1267,1278,2367,2378,3478	6
2,1,1	1258,1458,1478	7

Observe that if we consider all possible subsets of 4 elements of $\{1, \ldots, 8\}$ we would have considered 70 sets among which we should have made the corresponding finding and elimination of the 44 redundant ones.

3. Computer implementations

The implementation of algorithms for computing the generators and Betti numbers of the ideals of the lcm-filtration of an ideal are in general expensive due to the size of the ideals involved. If the number of generators of I is r then the number of generators of I_i can be $\binom{r}{i}$. In fact, generally we compute all $\binom{r}{i}$ least common multiples of i generators of I and then reduce the nonminimal ones. So that we are performing redundant computations.

Using the results above we can design efficient algorithms to list the minimal generators of the ideals in the lcm-filtration of ideals corresponding to k-out-of-n and consecutive k-out-of-n systems, which allow us to apply our methods to problems of reasonable size.

In the case of k-out-of-n systems, the problem amounts to list all the j-sets of variables for the pertinent j's as seen in Section 1.

The case of consecutive k-out-of-n can also be reduced to a combinatorial enumeration algorithm. In order to list all the generators of $J_{k,n,i}$ we have to list the i-subsets of $\{1,\ldots,n-k+1\}$ that have no gaps of size up to $k-1$. We describe here an algorithm to perform this task:

The initial element of a set of i elements of $\{1,\ldots,n-k+1\}$ can be 1 up to $n-k-i+2$. For such an initial value, say j, the sum of gaps between elements can be from 0 to $n-k-i+1-(j-1)$. Now, for each of these, let's call them m, we have to construct all possible ways to attain this total gap using gaps bigger than or equal to k. This amounts to count all compositions of m in at most i parts such that the smallest part is k. Those compositions with a number of parts a strictly smaller than i must be counted $\binom{i}{i-a}$ times. Summing up all these sets for all j we obtain the minimal generators of $J_{k,n,i}$.

Our initial computer experiments show that this algorithm gives the minimal generating sets of the ideals in the lcm-filtration of consecutive k-out-of-n in a reasonable time for fairly big systems. This makes the algebraic approach usable for the study of multiple failures of k-out-of-n and consecutive k-out-of-n systems.

References

[1] W. Kuo, M.J. Zuo, Optimal reliability modelling, Wiley and Sons, New Jersey, 2003
[2] F. Mohammadi, E. Sáenz-de-Cabezón and H. Wynn, Types of signature analysis in reliability based on Hilbert series, http://arxiv.org/abs/1510.04427
[3] F. Mohammadi, E. Sáenz-de-Cabezón and H. Wynn, Computing the homology of the lcm-filtration of a monomial ideal, *in preparation*
[4] E. Sáenz-de-Cabezón and H. Wynn, Betti numbers and minimal free resolutions for multi.state system reliability bounds, Journal of Symbolic Computation 44 (2009) 1311–1325
[5] E. Sáenz-de-Cabezón and H. Wynn, Computational algebraic algorithms for the reliability of generalized k-out-of-n and related systems, Mathematics and Computers in Simulation 82 (2011) 68–78
[6] E. Sáenz-de-Cabezón and H. Wynn, Hilbert functions in design for reliability, IEEE Transactions on Reliability 64 (2015) 83–93

Technische Universität Berlin
E-mail address: mohammad@math.tu-berlin.de

Universidad de La Rioja
E-mail address: eduardo.saenz-de-cabezon@unirioja.es

London School of Economics
E-mail address: h.wynn@lse.ac.uk

A NOTE ON BURCHNALL-CHAUNDY POLYNOMIALS AND DIFFERENTIAL RESULTANTS

JUAN J. MORALES-RUIZ, SONIA L. RUEDA, AND Mª ÁNGELES ZURRO

Abstract. We review the definition of the differential resultant of two ordinary differential operators and its main properties. We revisit Enma Previato's result about the computation of the spectral curve of two commuting differential operators using differential resultants. We use these results to establish the appropriate fields were commuting operators have a common factor, which can be computed using differential subresultants. These results will allow us to give new explanations to some well known results related with the celebrated KdV hierarchy.

Introduction

In 1928, J.L. Burchall and T.W. Chaundy [3] established a correspondence between commuting differential operators and algebraic curves. With the discovery of solitons and the integrability of the KdV equation, in [5], using the inverse spectral methods, their theory found applications to the study of partial differential equations called integrable (or with solitonic type solutions: Sine-Gordon, non lineal Schrödinger, etc). Burchall and Chaundy had discovered the spectral curve (defined by the so called Burchall and Chaundy polynomial), which was later computed by E. Previato (1991) using differential resultants and allows an algebraic approach to handling the inverse spectral problem for the finite-gap operators, with the spectral data being encoded in the spectral curve and an associated line bundle. In this work, we explore the benefits of using differential resultants to compute Burchall and Chaundy polynomials.

1. Differential resultant of ODO's

Differential resultants for ordinary differential operators were defined by Berkovich and Tsirulik [1] and studied by Chardin [4], who also defined the subresultant sequence. We summarized next the definition and some main properties of differential resultants to be used in this note.

Let K be a differential field with derivation ∂ and field of constants C, algebraically closed and of zero characteristic. We denote by $K[\partial]$ the ring of differential operators with coefficients in K and commutation rule $\partial a - a\partial = \partial(a), a \in K$. Given differential operators P and Q in $K[\partial]$ of orders n and m respectively, the Sylvester matrix $S(P,Q)$ is the coefficient matrix of the extended system of differential operator

$$\Xi(P,Q) = \{P, \partial P, \ldots, \partial^{m-1}P, Q, \partial Q, \ldots, \partial^{n-1}Q\}.$$

Observe that $S(P,Q)$ is an squared matrix of size $n+m$ and entries in K. We define the differential resultant of P and Q to be $\partial\mathrm{Res}(P,Q) := \det(S(P,Q))$. The next result was proved in [4].

Proposition 1.1. *Let (P,Q) be the left ideal generated by P,Q in $K[\partial]$.*

(1) $\partial\mathrm{Res}(P,Q) = AP + BQ$ *with* $A,B \in K[\partial]$, $\mathrm{ord(A)} < \mathrm{m}$, $\mathrm{ord(B)} < \mathrm{n}$, *that is* $\partial\mathrm{Res}(P,Q)$ *belongs to the elimination ideal* $(P,Q)\cap K$.
(2) $\partial\mathrm{Res}(P,Q) = 0$ *if and only if* $P = P_1R$, $Q = Q_1R$, *with* $\mathrm{ord(R)} > 0$, $P_1, Q_1, R \in K[\partial]$.

2. Burchnall-Chaundy polynomial

Given two commuting differential operators P and Q, it is a well known result of Burchnall and Chaundy [3] that there exists a polynomial $f(\lambda,\mu) \in C[\lambda,\mu]$ such that $f(P,Q) = 0$. Such polynomial is called a Burchnall-Chaundy polynomial and it is the polynomial that defines the spectral curve. Computing the Burchnall-Chaundy polynomial requires noncommutative differential elimination, the elimination of the derivation ∂ from the differential operators $P-\lambda$ and $Q-\mu$. Thus, it is natural that E. Previato in [9] used the differential resultant to compute the Burchnall-Chaundy polynomial. We revisit Previato's proof but using Poisson's formula for differential resultants.

Given a fundamental system of solutions $y_1,\dots,y_n$ of P, let us denote by $w(y_1,\dots,y_n)$ their Wronskian. The next result gives a Poisson formula for $\partial\mathrm{Res}(P,Q)$, see [4] and [9].

Proposition 2.1. *Given $P,Q \in K[\partial]$ with respective orders n and m, leading coefficients a_n and b_m and fundamental systems of solutions $y_1,\dots,y_n$ and $z_1,\dots,z_m$ respectively. It holds,*

$$\partial\mathrm{Res}(P,Q) = (-1)^{nm} a_n^m \frac{w(Q(y_1),\dots,Q(y_n))}{w(y_1,\dots,y_n)} = b_m^n \frac{w(P(z_1),\dots,P(z_m))}{w(z_1,\dots,z_m)}.$$

Let $(P-\lambda, Q-\mu)$ be the ideal generated by $P-\lambda$ and $Q-\mu$ in $K[\lambda,\mu][\partial]$. Observe that

$$\partial\mathrm{Res}(P-\lambda, Q-\mu) \in (P-\lambda, Q-\mu)\cap K[\lambda,\mu].$$

In addition, if $\mathrm{ord(P)} = \mathrm{n}$ with leading coefficient a_n and $\mathrm{ord(Q)} = \mathrm{m}$ with leading coefficient b_m

$$\partial\mathrm{Res}(P-\lambda, Q-\mu) = a_n^m\mu^n - b_m^n\lambda^m + \ldots \neq 0. \tag{1}$$

Theorem 2.2 ([9])**.** *Given $P,Q \in K[\partial]$ such that $[P,Q] = PQ - QP = 0$ then $g(\lambda,\mu) := \partial\mathrm{Res}(P-\lambda, Q-\mu) \in C[\lambda,\mu]$ and $g(P,Q) = 0$.*

Seen as differential operators in $K[\lambda,\mu][\partial]$, by (1) the operators $P-\lambda$ and $Q-\mu$ have no common nontrivial solution. Let f be the square free part of g. The algebraic curve

$$\Gamma := \{(\lambda,\mu) \in C^2 \mid f(\lambda,\mu) = 0\}$$

is known as the spectral curve, see [8]. Let $K(\Gamma)$ be the fraction field of the domain $\frac{K[\lambda,\mu]}{(f(\lambda,\mu))}$. By Proposition 1.1, as elements of $K(\Gamma)[\partial]$, the differential operators $P-\lambda, Q-\mu$ have a common non constant factor. We can compute such factor using differential subresultants (see [4]), it is the greatest common left divisor $\mathrm{gcld(P-\lambda, Q-\mu)}$. The rank of the pair

$P-\lambda, Q-\mu$ is the dimension of the space of common solutions of $P-\lambda$ and $Q-\mu$, the order of the common nontrivial factor of $P-\lambda$ and $Q-\mu$. By [8], the rank of a commutative pair P, Q is $r = \gcd(\mathrm{ord(P)}, \mathrm{ord(Q)})$.

3. Stationary KdV Hierarchy

We study some important families of pairs of commuting differential operators of rank one, in particular, we study the centralizer of the an Schrödinger operator $L(u) = \partial^2 - u$ with u a differential indeterminate (we can think of the Schrödinger operator in the stationary case, where $u = u(x)$ and $\partial = \partial/\partial x$). Let us consider the operator

$$A_3(u) = -\partial^3 + \left(\frac{3}{2}u + 1\right)\partial + \frac{3}{4}u_x.$$

The pair $L(u), A_3(u)$ is known as the first Lax pair of the celebrated KdV hierarchy (see for instance [6]). The commutator of this Lax pair equals the differential polynomial $\mathrm{KdV}_1(u)$ defining the KdV equation $KdV_1(u) = 0$,

$$[L(u), A_3(u)] = -\frac{1}{4}u_{xxx} + \frac{3}{2}uu_x + u_x = KdV_1(u). \tag{2}$$

Furthermore, we can compute (we did it using Maple) the coefficients of $\partial\mathrm{Res}(L(u), A_3(u)) = \mu^2 - \lambda^3 + a_2(u)\lambda^2 + a_1(u)\lambda + a_0(u)$, which are the next differential polynomials in u

$$a_0(u) = (1/8)u_{xx}u + (1/4)u_{xx} - u^2 - u - (1/16)u_x^2 - (1/4)u^3,$$
$$a_1(u) = u - 1 + 3/4(u)^2 - (1/4)u_{xx}, \;\; a_2(u) = 2.$$

We check that their derivatives are multiples of $KdV_1(u)$ (observe that this is the Novikov equation forcing the commutativity of the operators $L(u)$ and $A_3(u)$, see [8]). Therefore $\partial\mathrm{Res}(L(u), A_3(u))$ has constants coefficients under the assumption $KdV_1(u) = 0$. In particular, for the Rosen-Morse potential $u_1 = -2/\cosh^2(x)$ the operators commute and $f(L(u_1), A_3(u_1)) = 0$ with $f(\lambda, \mu) = \partial\mathrm{Res}(L(u_1) - \lambda, A_3(u_1) - \mu) = \mu^2 - \lambda(\lambda+1)^2$.

See [6] for the whole sequence of operators $A_{2n+1}(u)$, $n \in \mathbb{N}$ that satisfy $[L(u), A_{2n+1}(u)] = KdV_n(u)$, which is a differential polynomial in u. For instance if $u_s = \frac{-s(s+1)}{\cosh^2(x)}$ then $L_s = L(u_s)$ commutes with $A_{2n+1}(u_s)$, for $n \geq s$, that is

$$[L(u_s), A_{2n+1}(u_s)] = KdV_n(u_s) = 0, \;\; n \geq s. \tag{3}$$

For example u_1 satisfies the whole KdV Hierarchy $KdV_n(u_s) = 0$, $n \geq 1$ and by results in [7] we can prove that the centralizer $\mathcal{C}(L(u_1))$ of $L(u_1)$ is the free $\mathbb{C}[L(u_1)]$-module with basis $\{1, A_3(u_1)\}$. It is well known that there are other families of potentials satisfying (3) (for example the rational potentials $s(s+1)/x^2$, $s \in \mathbb{N}$).

Theorem 3.1. *Let $L_s = L(u_s)$ be a stationary Schrödinger operator defined by a potential u_s verifying $[L_s, A_{2n+1}(u_s)] = KdV_n(u_s) = 0$ for $n \geq s$. Then*

$$\partial\mathrm{Res}(L_s - \lambda, A_{2n+1}(u_s) - \mu) = \mu^2 - \partial\mathrm{Res}(L_s - \lambda, A_{2n+1}(u_s)).$$

If in addition we suppose that $A_{2n+1}(u_s) = \pm L_s^{2(n-s)} A_{2s+1}(u_s)$ then

$$\partial\mathrm{Res}(L_s - \lambda, A_{2n+1}(u_s) - \mu) = \mu^2 - \lambda^{2(n-s)}\partial\mathrm{Res}(L_s - \lambda, A_{2s+1}(u_s)). \tag{4}$$

The family of operators $A_{2n+1}(u)$ is defined via recursive formulas (see for instance [6]) which in certain cases are simplified to $A_{2n+1}(u) = \pm L(u)A_{2n}(u)$, this is the case for the next families of potentials (computations done with Maple).

(1) **For** $u_s = s(s+1)/x^2$, $s \geq 1$ **let** $L_s := L(u_s)$. By Theorem 3.1

$$\partial\mathrm{Res}(L_1 - \lambda, A_{2n+1}(u_1) - \mu) = \mu^2 - \lambda^{(n-1)}\lambda^3, \ \ n \geq s = 1 \text{ and}$$
$$\partial\mathrm{Res}(L_2 - \lambda, A_{2n+1}(u_2) - \mu) = \mu^2 - \lambda^{(n-2)}\lambda^5, \ \ n \geq s = 2.$$

(2) **For** $u_s = \frac{-s(s+1)}{\cosh^2(x)}$, $s \geq 1$ **let** $L_s := L(u_s)$. By Theorem 3.1

$$\partial\mathrm{Res}(L_1 - \lambda, A_{2n+1}(u_1) - \mu) = \mu^2 - \lambda^{(n-1)}\lambda(\lambda-1)^2, \ \ n \geq s = 1 \text{ and}$$
$$\partial\mathrm{Res}(L_2 - \lambda, A_{2n+1}(u_2) - \mu) = \mu^2 - \lambda^{(n-2)}\lambda(\lambda-1)^2(\lambda-4)^2, \ \ n \geq s = 2.$$

Once we have written the spectral curve in terms of the differential resultant, we will use the ideas in Section 2 to factor an stationary Schrödinger operator over the fraction field of the spectral curve, showing the connection with the results in Brezhnev [2]. The main techniques for this purpose will be differential subresultants (see [4]) and rational parametrizations of curves (in the talk we will exhibit some concrete examples).

References

[1] Berkovich, L.M. and Tsirulik, V.G., 1986. Differential resultants and some of their applications. Differential Equations, Plenum Publ. Corp., 22, 750-757.

[2] Brezhnev, Y.,2013. Elliptic solitons, Fuchsian equations, and algorithms. St. Petersburg Mathematical Journal, 24(4), 555-574.

[3] Burchnall, J.L., Chaundy, T.W., 1928. Commutative ordinary differential operators. Porc. R. Soc. A 118, 557-583.

[4] Chardin, M., 1991. Differential Resultants and Subresultants. Proc. FCT'91, Lecture Notes in Computer Science, 529, Springer-Verlag.

[5] Gardner, C.S., Greene, J.M., Kruskal, M.D., Miura, R.M. (1967). Method for solving the Korteweg-de Vries equation. Physical Review Letters, 19(19), 1095.

[6] Gesztesy, F., Holden, H., 2003. Soliton Equations and their Algebro-Geometric Solutions: Volume 1, (1+1)-Dimensional Continuous Models. Cambridge University Press.

[7] Goodearl, K.R., 1983. Centralizers in differential, pseudo-differential and fractional differential operator rings. Rocky Mountain Journal of Mathematics, 13 (4), 573-618.

[8] Krichever, I.M., 1978. Commutative rings of ordinary linear differential operators. Funct. Anal. Appl. 12, no. 3, 175-185.

[9] Previato E., 1991. Another algebraic proof of Weil's reciprocity. Atti Accad. Naz. Lincei Cl. Sci. Fis. Mat. Natur. Rend. Lincei (9) Mat. Appl. 2, no. 2, 167-171.

Universidad Politécnica de Madrid. Member of the Research Group "Modelos matemáticos no lineales", UPM.

E-mail address: juan.morales-ruiz@upm.es

Universidad Politécnica de Madrid. Member of the Research Group "Modelos matemáticos no lineales", UPM. Partially supported by the "Ministerio de Economía y Competitividad" under the project MTM2014-54141-P.

E-mail address: sonialuisa.rueda@upm.es

Universidad Autónoma de Madrid

E-mail address: mangeles.zurro@uam.es

ARE ALL SECANT VARIETIES OF SEGRE PRODUCTS ARITHMETICALLY COHEN-MACAULAY?

LUKE OEDING

ABSTRACT. The general problem of arithmetic Cohen-Macaulayness for secant varieties of Segre products is investigated. A inductive procedure based on the work of Landsberg and Weyman is given. New computational results are presented for rank 4 tensors of format $3 \times 3 \times 4$, together with a certain generalization.

INTRODUCTION

After finding many equations in low degree, the question then remains: What is the maximal degree of minimal defining equations of a given secant variety, and when do the known equations suffice? This question is well studied in some cases such as monomial ideals, for curves, and in some infinite dimensional cases, however the general question is still very open. It may be possible to obtain upper bounds (via Castelenuovo-Mumford regularity, for example), but these computations are often also difficult. Another approach undertaken by Aschenbrenner and Hillar, Draisma and Kutler, Sam and Snowden, and others is to investigate these questions in an infinite dimensional setting. This method has been used to determine when certain ideals are "Noetherian up to symmetry" and in turn, this can provide a (non-constructive) guarantee that tensors of bounded rank are defined by equations in bounded degree not depending on the number of tensor factors. This method, however, does not typically give an explicit bound. The varying degrees of success of these approaches are indicators that secant varieties of Segre products may all share particularly nice structural property that governs the behavior of their ideals, namely the Cohen-Macaulay property. This article is focused on uncovering this ground truth about secant varieties.

Another way to know when the given equations generate a prime ideal that might be available is if the variety is arithmetically Cohen-Macaulay (aCM), i.e. if the depth of its coordinate ring is equal to the codimension. If the variety is aCM, one can determine if the given ideal agrees with the ideal of the variety by checking if it (a) cuts the variety out set-theoretically, and (b) is generically reduced. Inspired by recent evidence and classical results we offer the following:

Conjecture 0.1. *Every secant variety of a Segre product is arithmetically Cohen-Macaulay.*

In addition to the implicitization motivation, this conjecture is interesting because aCM varieties have many nice properties, such as being equi-dimensional, and being aCM says that the singularities of the variety are mild. Thus knowing whether a variety is aCM or not gives insight into how complicated that variety is. Evidence for Conjecture 0.1 includes the following cases where a secant variety of a Segre product is known to be aCM: Segre

varieties themselves, when the secant variety fills its ambient space or is a hypersurface, determinantal varieties, subspace varieties, as well as several other special cases.

Geramita made the symmetric version of Conjecture 0.1, (see [9, p55]), and Kanev proved special cases:

Theorem 0.2 (Kanev, [10]). *$\sigma_s(\nu_d\mathbb{P}^n)$ is aCM if either $d = 2$, or $n = 1$ or $s \leq 2$.*

The dimension, degree, and singular locus are also known. A local, non-symmetric version of Kanev's result is the following result obtained by the author together with Michalek and Zwiernik:

Theorem 0.3 (Michalek-Oeding-Zwiernik [12]). *$\sigma_2(\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_m})$ is covered by open normal toric varieties. In particular $\sigma_2(\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_m})$ is locally Cohen-Macaulay.*

Theorem 0.4 below pushed the computational boundaries and confirms Conjecture 0.1 in a new case. The inspiration came from B. Sturmfels's "algebraic fitness session" at the Simon's Institute, Fall 2014, and the following results: It is known classically that $2\times2\times2\times2$ tensors are defective in rank 3. Bocci and Chiantini showed that $2 \times 2 \times 2 \times 2 \times 2$ tensors are not identifiable in rank 5 — the generic tensor of that format has exactly 2 decompositions [3]. Further, Bocci-Chiantini-Ottaviani showed that for ≥ 6 factors, the Segre is almost always k-identifiable, This generated interest in finding the equations for the boundary case of 5 binary factors of rank 5.

Theorem* 0.4 (Oeding-Sam [13]). *The affine cone of $\sigma_5(\mathrm{Seg}(\mathbb{P}^{1\times5}))$ is a complete intersection of a degree 6 and a degree 16 equation.*

The star refers to the use of careful numerical, sometimes probabilistic computations in the proofs. In particular, $\sigma_5(\mathrm{Seg}(\mathbb{P}^{1\times5}))$ is aCM. These computations took approximately *two weeks of human/computer time.*

An adaptation of Weyman's geometric technique. Landsberg and Weyman applied a partial desingularization together with a mapping cone argument to show the following.

Theorem 0.5 (Landsberg–Weyman [11]). *Suppose $X := \sigma_r(\mathrm{Seg}(\mathbb{P}^{r-1\times d}))$ is aCM, with "a resolution by small partitions." If $n_i \geq r-1$ for all $1 \leq i \leq d$, then $\sigma_r(\mathrm{Seg}(\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_d}))$ is aCM and its ideal is generated by those inherited from X and the $(r+1)\times(r+1)$-minors of flattenings.*

This begs the question about the cases of $\sigma_r(\mathrm{Seg}(\mathbb{P}^{n_1}\times\cdots\times\mathbb{P}^{n_d}))$ when $n_i < r-1$ for some i. The concept of resolutions via small partitions (which appeared as a technical curiosity in [11]. An R-module M has a resolution by small partitions if the Schur modules which occur in the G-equivariant resolution are indexed by partitions that fit inside prescribed sized boxes. More specifically, suppose $A_i' \subseteq A_i$ for $1 \leq i \leq n$ and let $S_\pi A$ denote the Schur module associated to the multi-partition π. A G-variety Y has an (s_j)-small resolution if every module $S_\pi A$ occurring in the resolution has the property

$$\text{for each } j \text{ the first part of } \pi^j \text{ is not greater than } s_j.$$

Let $\widehat{a_j} := \frac{a_1\cdots a_n}{a_j}$, $\widehat{r_j} := \frac{r_1\cdots r_n}{r_j}$, $G = \mathrm{GL}(A_1) \times \cdots \times \mathrm{GL}(A_n)$ and $G' = \mathrm{GL}(A_1') \times \cdots \times \mathrm{GL}(A_n')$.

Theorem 0.6 (Oeding, (adapted from [11])). *If a G'-variety Y is an aCM with an resolution that is $(\widehat{r_j} - r_j)$-small for every j for which $0 < r_j < a_j$, then $\overline{G.Y}$ is aCM.*

Moreover one obtains a (not necessarily minimal) resolution of $\overline{G.Y}$ that is (s_j)-small with

$$s_j = \max_\pi \begin{cases} \widehat{a_j} - r_j, & \text{if } r_j < a_j \\ \widehat{a_j} - \widehat{r_j} + \pi^j_1, & \text{if } r_j = a_j, \end{cases}$$

where the max *is taken over all multi-partitions π occurring in the resolution of $\mathbb{C}[Y]$.*

Applying the geometric technique to Strassen's degree 9 hypersurface one obtains the following.

Proposition 0.7.

$\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^2)$	*is aCM, deg. 9 hypersurface*	*[Strassen]*
$\mathrm{GL}(4).\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^2)$	*is aCM and* codim *3 in*	$\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^3)$.
$\mathrm{GL}(4)^{\times 2}.\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^2)$	*is aCM and* codim *4 in*	$\sigma_4(\mathbb{P}^2 \times \mathbb{P}^3 \times \mathbb{P}^3)$.
$\mathrm{GL}(4)^{\times 3}.\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^2)$	*is aCM and* codim *5 in*	$\sigma_4(\mathbb{P}^3 \times \mathbb{P}^3 \times \mathbb{P}^3)$.
$\mathrm{GL}(4).\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^3)$	*is aCM and* codim *1 in*	$\sigma_4(\mathbb{P}^2 \times \mathbb{P}^3 \times \mathbb{P}^3)$.
$\mathrm{GL}(4)^{\times 2}.\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^3)$	*is aCM and* codim *2 in*	$\sigma_4(\mathbb{P}^3 \times \mathbb{P}^3 \times \mathbb{P}^3)$.
$\mathrm{GL}(4).\sigma_4(\mathbb{P}^2 \times \mathbb{P}^3 \times \mathbb{P}^3)$	*has* codim *1 in*	$\sigma_4(\mathbb{P}^3 \times \mathbb{P}^3 \times \mathbb{P}^3)$.

It remains to determine if one may lift the aCM property further. If possible, this will complete a major step forward, since in particular it would solve the salmon conjecture [1, 2, 5, 6].

Let $R = \mathbb{C}[A \otimes B \otimes C]$ and $G = \mathrm{GL}(A) \times \mathrm{GL}(B) \times \mathrm{GL}(C)$. Using Galetto's HighestWeights package [7] in Macaulay2, we determined the G-module structure of the minimal free resolution of $\sigma_4(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^3)$, and verified that it is aCM with a resolution by small partitions (we omit the details for space reasons). Daleo and Hauenstein have also numerically verified that this variety is Cohen-Macaulay [4]. After checking that the ideal is generically reduced, one obtains the following.

Theorem 0.8. *The secant variety $\sigma_4(\mathrm{Seg}(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^3))$ is arithmetically Cohen-Macaulay. Its prime ideal is minimally generated by the 10 degree 6 Landsberg-Manivel equations, and the 20 degree 9 equations inherited from Strassen's equation.*

Applying the adaptation of the Landsberg–Weyman inheritance result, and the G-module structure one has:

Theorem 0.9. *The secant variety $\sigma_4(\mathrm{Seg}(\mathbb{P}^2 \times \mathbb{P}^2 \times \mathbb{P}^n))$ is arithmetically Cohen-Macaulay for all $n \geq 0$. Its prime ideal is minimally generated by $\binom{n+2}{2}$ degree 6 (Landsberg-Manivel) equations (nontrivial when $n \geq 3$), and $\binom{n+3}{3}$ degree 9 (Strassen) equations (nontrivial when $n \geq 2$).*

Special cases of Conjecture 0.1 are also quite interesting. In the case $s = 2$ [15] that resolved the Garcia-Stillman-Sturmfels Conjecture [8] that the 3×3 minors of flattenings minimally define the ideal of σ_2, and it is known that the variety is locally Cohen-Macaulay [12], so it is expected that the aCM property should also hold, and it would be a nice addition to the story. Moreover, the connection to cumulants found in [16] and [12] simplified much of the computational efforts.

If in the case $s = 3$ one can prove that σ_3 is aCM for any number of factors, this would improve Yang Qi's recent result that σ_3 is defined set-theoretically by Strassen's degree 4 equations [14] together with flattenings by showing that his result holds even ideal-theoretically.

In addition it is anticipated that insights discovered in a case-by-case study will also prove useful in special cases that currently command much attention outside of pure mathematics. The matrix multiplication tensor for 3×3-matrices, denoted MM_3, lies in $\sigma_s(\mathbb{P}^8\times\mathbb{P}^8\times\mathbb{P}^8)\subset\mathbb{P}^{728}$ for some $14\leq s\leq 21$. It would be a major result in Computational Complexity to determine s exactly. Settling the case $s = 4$ and 3 tensor factors of Conjecture 0.1 would resolve the salmon conjecture for Phylogenetics. If one knew that they were arithmetically Cohen-Macaulay, then this would allow more tools from Commutative Algebra to be used in their study.

References

[1] E. Allman, *Open problem: Determine the ideal defining* $Sec_4(\mathbb{P}^3\times\mathbb{P}^3\times\mathbb{P}^3)$, http://www.dms.uaf.edu/ eallman/salmonPrize.pdf, 2010.

[2] D. J. Bates and L. Oeding, *Toward a salmon conjecture*, Exp. Math. **20** (2011), no. 3, 358–370, `arXiv:1009.6181`.

[3] C. Bocci and L. Chiantini, *On the identifiability of binary Segre products*, J. Algebraic Geom. **22** (2013), no. 1, 1–11.

[4] N. S. Daleo and J. D. Hauenstein, *Numerically deciding the arithmetically Cohen-Macaulayness of a projective scheme*, Journal of Symbolic Computation **72** (2016), 128–146.

[5] S. Friedland, *On tensors of border rank l in* $\mathbb{C}^{m\times n\times l}$, Linear Algebra Appl. **438** (2013), no. 2, 713–737.

[6] S. Friedland and E. Gross, *A proof of the set-theoretic version of the salmon conjecture*, J. Algebra **356** (2012), 374–379.

[7] F. Galetto, *HighestWeights: a package for free resolutions and modules with a semisimple lie group action*, preprint (2014), `http://www.mast.queensu.ca/~galetto/research.htm`.

[8] L. Garcia, M. Stillman, and B. Sturmfels, *Algebraic geometry of Bayesian networks*, J. Symbolic Comput. **39** (2005), no. 3-4, 331–355.

[9] A. V. Geramita, *Inverse systems of fat points: Waring's problem, secant varieties of Veronese varieties and parameter spaces for Gorenstein ideals*, The Curves Seminar at Queen's, Vol. X (Kingston, ON, 1995), Queen's Papers in Pure and Appl. Math., vol. 102, Queen's Univ., Kingston, ON, 1996, pp. 2–114.

[10] V. Kanev, *Chordal varieties of Veronese varieties and catalecticant matrices*, J. Math. Sci. (New York) **94** (1999), no. 1, 1114–1125, Algebraic geometry, 9.

[11] J.M. Landsberg and J. Weyman, *On the ideals and singularities of secant varieties of Segre varieties*, Bull. Lond. Math. Soc. **39** (2007), no. 4, 685–697.

[12] M. Michalek, L. Oeding, and P. Zwiernik, *Secant cumulants and toric geometry*, International Mathematics Research Notices **2015** (2015), no. 12, 4019–4063, `arXiv:1212.1515`.

[13] L. Oeding and S. V Sam, *Equations for the fifth secant variety of Segre products of projective spaces*, Experimental Mathematics **25** (2016), 94–99, `arXiv:1502.00203`.

[14] Y. Qi, *Equations for the third secant variety of the Segre product of n projective spaces*, preprint (2013), `arXiv:1311.2566`.

[15] C. Raicu, *Secant varieties of Segre-Veronese varieties*, Algebra Number Theory **6-8** (2012), 1817–1868.

[16] B. Sturmfels and P. Zwiernik, *Binary cumulant varieties*, Ann. Comb. **17** (2013), no. 1, 229–250, `arXiv:1103.0153`.

Auburn University

E-mail address: `oeding@auburn.edu`

GENERATING (CO)HOMOLOGICAL INTERACTIONS WITHIN AT-MODEL CONTEXT

PEDRO REAL

Abstract. Working with coefficients in a field and starting from an AT-model (Algebraic Topological Model) of a finite cell complex X, an algorithmic framework for computing a new family of non-functorial correspondences between (co)homology classes of X is designed. These correspondences, called (co)homology interactions, are induced by elementary relations between cells of X of the kind "to be in the (co)boundary of".

Roughly speaking, (co)homology information with coefficients in a field could be defined as the set of processed and structured linear algebraic data related to its (co)homology classes and theirs relations between them. We talk about homology and cohomology information (with coefficients in a field) as a whole due to the fact that they are dual algebraic concepts and we assume that they provide the same measurable information quantities with regards the same object. A simple example of (co)homology information is provided by the numerical topological invariants called Betti numbers. If X is a cell complex embedded in $\mathbf{R}^3$, Betti numbers β_0, β_1 and β_2 respectively measure the number of different connected components, (co)homological tunnels and cavities of X. Nevertheless, (co)homology information of X is not reduced in general to that provided by Betti numbers. For example, a torus T and a three-dimensional sphere with two handles S have the same Betti numbers but they are not (co)homologically equivalents. The two tunnels of T are related to its cavity in a much more "stronger" way than the tunnels of S are with regards to the corresponding cavity. In order to advance in a coherent "(co)homology information theory" and understanding homology as a kind of homotopy, a first step to design appropriate mathematical models allowing us not only to anticipate a correct computation at the corresponding homology or homotopy level, but also to easily channel and interpret useful information from one level to the other. As homotopy level, we propose to work with *primal-dual abstract cell complexes (pACC-model for short)* or, in other words, finite CW-complexes modeled as partially bi-ordered sets $(X, \leq, \geq)$ with regards to the relationships "to be in the (co)boundary of". As model for computing and interpreting advanced (co)homology information, we work with **chain-integral complexes**, that is, finitely generated chain complexes (C, d, cd) with regards two different differential operators d and cd.

Using an analogy with signal processing, the first step needed for modulating a coherent and qualitative information theory at homology "frequency-band" is the construction of a flexible representation under the corresponding format of chain-integral complex. Algebraic-Topological models (or AT-models for short) ([3, 4]) will be used for this task. In this paper, we focus our attention to convenient "homology-frequency modulations" in terms of AT-models such that combinatorial signals can also be modulated under these algebraic schemes. This coherent modulation process is described by a kind of operation between (co)homology

classes belonging to a basis of the (co)homology of X as graded vector space: *an "AT-model-dependant" (co)homology interaction*. This (co)homology information must be differentiated from the one provided by classical (co)homology operations which are functorially defined. (Co)homology operations, like Steenrod squares and powers, cup-product in (co)homology or high-order cohomology operations (see [1, 2] for an algorithmic treatment) are determined at (co)chain level via universal maps for any finite simplicial complex X. A (co)homology interaction is defined at (co)chain level by suitably transferring combinatorial signals of X (related to the relations between cells "to be in the (co)boundary of") into a concrete AT-model setting, homologically modeling X. If the corresponding homologically modulated signal determines a non-trivial interaction, there is no guarantee that this interaction persists at homotopy or homeomorphic level.

In the sequel, we try to briefly describe the method for generating (co)homology interactions within the context of enriched AT-models and a first approximation to the problem of fixing a "normalized" system of (co)homological coordinates for $Betti(X)$.

1. Chain-integral complexes

From now on, let F be the field of coefficients. Let us define a χ-complex (C, d, ϕ)

Definition 1.1. A chain-integral complex (or simply, χ-complex) (C, d, ϕ) is a graded vector space $C = \{C_p\}_{p=0}^{n}$ endowed with two linear maps: a differential operator $d : C_* \to C_{*-1}$, and an integral or codifferential operator $\phi : C_* \to C_{*+1}$, satisfying the global nilpotency properties $d_* \circ d_{*+1} = 0$ and $\phi_{*+1} \circ \phi_* = 0$. An χ-complex (C, d, ϕ) is *pure differential* (resp. *pure integral*) if the condition $d = d \circ \phi \circ d$, called homology condition (resp. the condition $\phi = \phi \circ d \circ \phi$, called *cohomology* condition) is satisfied. A χ-complex (C, d, ϕ) that is both, pure differential and pure integral, is called *homology χ complex.*

A χ-complex arises naturally from a finite cell complex structure X, where each C_q is generated by its q-dimensional cells, its differential (resp. integral) operator $\partial_q(c)$ (resp. $\delta_q(c)$) is a linear combination of $(q-1)$-dimensional (resp. $(q+1)$-dimensional) cells in the boundary (resp. coboundary) of c with the coefficients reflecting incidence numbers and orientation. The χ-complex canonically associated to X is denoted by $(C(X), \partial, \delta)$. Since (differential) homology and (integral) cohomology are invariants saving the same amount of information of the topological space (polyhedron) which is the union of all the cells of X, it is also called the χ-homology of the topological space.

Definition 1.2. A χ-map $f : (C, d, \phi) \to (C, d', \phi')$ is a map inducing well-defined maps at differential and integral homology level. In other words, the following maps make sense: (a) $[f]_d : H(C, d) \to H(C', d')$ (passing f to homology with regards to d); (b) $[f]_\phi : H(C, \phi) \to H(C', \phi')$ (passing f to homology with regards to ϕ)

In general, the map f don't necessarily need to satisfy the commutativity relations $d'f = fd$ nor $\phi' f = f\phi$. If one of these relations is satisfied, the map f is called strong χ-map.

In an analogous way, we define a notion of χ-homology equivalence as a pair of χ-maps f and g between two χ-complexes such that the compositions fg and gf are equal to the respective identity maps at (integral or differential) homology level.

Using the χ-language, an AT-model for a finite cell complex X is a homology χ-complex of the kind $(C(X), \partial, \phi)$, where $\phi \neq \delta$ and χ-homologically equivalent to $(C(X), \partial, \delta)$.

Definition 1.3. Let (C, d, ϕ) an χ-complex. Let $\pi : C_* \to C_*$ be the linear map, (called *flow of* (C, d, ϕ)), defined by $\pi = id_C - d \circ \phi - \phi \circ d$. The χ-complex $\pi(C, d, \phi) = (\pi(C), d|_{\pi(C)}, \phi|_{\pi(C)})$ is the *harmonic complex associated to* (C, d, ϕ). If (C, d, ϕ) is a pure differential or a pure integral χ-complex, then $\pi^2 = \pi \circ \pi = \pi$ and $\pi(C) = \{x \in C | x = \pi(x)\}$.

Lemma 1.4. *A χ-complex (C, d, ϕ) is χ-homologically equivalent to its harmonic complex $\pi(C, d, \phi)$.*

In fact, three different χ-homology equivalences can be constructed between these last two complexes: (a) the pair $(\pi, incl)$; (b) the pair $(id - d\phi, id - \phi d)$; (c) the pair $(id - \phi d, id - d\phi)$. Let us note that the maps $id - d\phi$ and $id - \phi d$ are not strong χ-maps.

2. Computing (co)homology interactions

We are now able to construct a homology χ-complex $(C(X), \partial, \phi)$ for a finite cell complex X which is enriched with a discrete gradient vector field guaranteeing that the integral homology of it can be fixed using a combinatorial basis of cells of X.

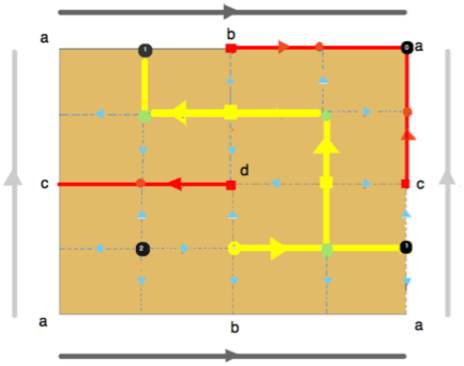


FIGURE 1. An enriched combinatorial AT-model $AT = (C(T), \partial, \phi)$ for a subdivided torus T. Some homology interactions are $CoInt(AT, db \leq abcd) = [ac]_\phi + [ab]_\phi \leq [abcd]_\phi$ or $CoInt(AT, cd \leq abcd) = 0$.

The output of the Algorithm 1 is a homology χ-complex $AT = (C(X), \partial, \phi)$, a combinatorial basis $\{b_1, \ldots, b_t\}$ for $H(X, \phi)$ given by $\mathcal{J}_n^\phi$ and a combinatorial integral operator $\tilde{\phi}$ associated to ϕ determined by the set of ordered pairs GVF_n^ϕ. Let us emphasize the use of non-strong χ-maps, like $id - d\phi$ and $id - \phi d$, in the construction of the integral operators ϕ and $\tilde{\phi}$.

Starting with these input data and fixing a cohomology class $[b_s]_\phi = [(id - \phi d)(b_s)]_\partial$ (let us note that $\phi(b_s) = 0$), it is possible to design an algorithm for transferring an elementary connectivity relation between cells to cohomology level by means of $(C(X), \partial, \phi)$. Let us choose in X a connectivity relation $bd_k(b_s) = e \leq b_s$ of the kind "the cell e is in the boundary of b_s". The elementary "coboundary" relation can be analogously handled. Then, the following χ-homology interaction $CoInt(AT, (c \leq [b_s]))$ is defined by the correspondence "connecting" $[b_s]_\phi$ with $[(id - \phi d)(e)]_\phi$, if this last class is non null or zero otherwise.

Algorithm 1 Combinatorial AT-model

Input:

Filtration $\mathcal{F} = \{F_i\}_{i=0}^n$ *of the finite cell complex* (X, ∂) *described by an ordered sequence* $\{c_1, c_2, \ldots, c_n\}$ *of all the cells of* X. *The differential* $\partial|F_i : F_i \to F_i$ *is denoted by* ∂_i.

$\mathcal{H}_0^\partial \leftarrow \{c_1\}$; $\mathcal{K}_0^\phi \leftarrow \{c_1\}$; $\mathcal{J}_0^\phi \leftarrow \{c_1\}$; $\mathcal{GVF}_0 \leftarrow \emptyset$;
$\phi_0(c_1) \leftarrow 0$;
for $i = 2$ to m **do**
    $\bar{c}_i \leftarrow c_i + \phi_{i-1}\partial_i(c_i)$;
    $\mathcal{H}_i^\partial \leftarrow H_{i-1}^\partial \bigcup \{\bar{c}_i\}$; $\mathcal{K}_i^\phi \leftarrow \mathcal{K}_{i-1}^\phi \bigcup \{c_i\}$; $\mathcal{J}_i^\phi \leftarrow \mathcal{J}_{i-1}^\phi \bigcup \{c_i\}$; $\mathcal{GVF}_i^\phi \leftarrow GVF_{i-1}^\phi$
    if $\partial_i(\bar{c}_i)) = 0$ **then**
        for $j = 1$ to $i - 1$ **do**
            $\phi_i(c_j) \leftarrow \phi_{i-1}(c_j)$;
    if $\partial_i(\bar{c}_i)$ is a sum of the kind $\sum_{j=1}^r u_j \neq 0$ where each u_j is a different element of $\mathcal{K}_{i-1}^\phi$ and at least one of them belongs to $\mathcal{J}_{i-1}^\phi$ **then**
        Choose one of the summands $u_j \in \mathcal{J}_{i-1}^\phi$ and let k be the subindex j of the chosen one;
        $\tilde{\phi}(u_k) \leftarrow c_i$;
        $\tilde{\phi}(u) \leftarrow 0$ for every $u \in \mathcal{K}_{i-1} \setminus \{u_k\}$;
        for $j = 1$ to $i - 1$ **do**
            $\phi_i(c_j) \leftarrow (\phi_{i-1} + (id_{X_i} - \phi_{i-1}\partial_{i-1})\tilde{\phi}(id_{X_i} - \partial_{i-1}\phi_{i-1}))(c_j)$;
        $\mathcal{H}_i^\partial \leftarrow \mathcal{H}_i^\partial \setminus \{\bar{u_k}, \bar{c_i}\}$;
        $\mathcal{K}_i^\phi \leftarrow \mathcal{K}_i^\phi \setminus \{u_k\}$;
        $\mathcal{J}_i^\phi \leftarrow \mathcal{J}_i^\phi \setminus \{u_k, c_i\}$;
        $\mathcal{GVF}_i^\phi \leftarrow \mathcal{GVF}_i^\phi \bigcup \{(u_k, c_i)\}$;

As we have commented before, there is no guarantee that (co)homology interactions "persist" if we use another combinatorial AT-model. Of course, the enriched AT-models can be used for fast and efficiently computing cohomology operations like Steenrod squares or cup product in cohomology. To progress in defining at (co)chain or (co)cyclic level, classical or new "strong" (co)homology operations is crucial for "stabilizing" the non-null values of cohomology interactions for the $Betti(X)$ pACC picture.

References

[1] R. Gonzalez-Diaz, P. Real, *A combinatorial method for computing Steenrod squares*, J. Pure Applied Algebra, vol. 139, Issues 1-3, (1999) 89-108.

[2] R. Gonzalez-Diaz and P. Real,*Computation of cohomology operations on finite simplicial complexes*, Homology Homotopy Appl., 5 n.2 (2003) 83Ð93.

[3] H. Molina-Abril, P. Real, A. Nakamura,R. Klette, *Connectivity calculus of fractal polyhedrons*. Pattern Recognition, 48(4), (2015) 1150-1160.

[4] P. Pilarczyk, P. Real, *Computation of cubical homology, cohomology and (co)homological operations via chain contraction*, Adv. Comput. Math.) 41: (2015) 253-275.

Institute of Mathematics of Seville University
E-mail address: real@us.es

NUMERICAL SEMIGROUPS: SUITABLE SETS OF PSEUDO-FROBENIUS NUMBERS

AURELIANO M. ROBLES-PÉREZ

Abstract. Let PF be a given set of positive integers. We study the problem of knowing whether there exists a numerical semigroup S such that the set of pseudo-Frobenius numbers of S is equal to PF. Moreover, we describe an algorithm to compute explicitly all the numerical semigroups (if they exist) whose set of pseudo-Frobenius numbers is PF.

Introduction

Let S be a numerical semigroup, that is, a cofinite submonoid of $(\mathbb{N},+)$, where $\mathbb{N}$ stands for the set of nonnegative integers.

An integer x is the *Frobenius number* of S (respectively, a *pseudo-Frobenius number* of S) if $x \notin S$ and $x+s \in S$, for all $s \in \mathbb{N}\setminus\{0\}$ (respectively, for all $s \in S\setminus\{0\}$). We denote by $\mathrm{F}(S)$ the Frobenius number of S and by $\mathrm{PF}(S)$ the set of pseudo-Frobenius numbers of S. It is obvious that $\mathrm{F}(S)=\max(\mathrm{PF}(S))$. The cardinality of $\mathrm{PF}(S)$ is called the *type* of S and is denoted by $\mathrm{t}(S)$. Moreover, we will denote by $\mathrm{H}(S)(=\mathbb{N}\setminus S)$ the set of *gaps* of S, and by $\mathrm{small}(S)(=S\cap[0,F(S)+1])$ the set of *small elements* of S.

Let us observe that the concept of type is connected to other problems in commutative algebra. For instance, if $\mathbb{K}[x]$ is a polynomial ring in one variable over a field $\mathbb{K}$, we define the semigroup ring of S as the monomial subring $\mathbb{K}[S]:=\mathbb{K}[t^s \mid s\in S]$. Then $\mathrm{t}(S)$ coincides with the Cohen-Macaulay type of $\mathbb{K}[S]$.

Let n be a positive integer and let $PF=\{g_1,\ldots,g_n\}$ be a set of positive integers. Let us denote by $\mathcal{S}(PF)=\{S \mid S \text{ is a numerical semigroup with } \mathrm{PF}(S)=PF\}$. As an immediate consequence of the results in [1] and [4], we have that

(1) if $PF=\{f\}$, then $\mathcal{S}(PF)\neq\emptyset$ if and only if f is an odd integer greater than or equal to -1 (case corresponding to the symmetric numerical semigroups);

(2) if $PF=\left\{f,\frac{f}{2}\right\}$, then $\mathcal{S}(PF)\neq\emptyset$ if and only if f is an even positive integer (case corresponding to the pseudo-symmetric numerical semigroups).

Moreover, in [1] it was shown an efficient algorithm to fully compute $\mathcal{S}(\{f\})$ and $\mathcal{S}\left(\left\{f,\frac{f}{2}\right\}\right)$. Now, the following two questions arise naturally.

Q1. Find conditions on the set PF that ensure that $\mathcal{S}(PF)\neq\emptyset$.

Q2. Find an algorithm to compute $\mathcal{S}(PF)$.

Our aims are: (1) show a general algorithm for Q2 (see [3]); (2) give a exhaustive answer to Q1 if $PF=\{g_1,g_2\}$ (see [5]); (3) give partial results when $PF=\{g_1,g_2,g_3\}$.

Work supported by the projects FQM-343 (Junta de Andalucía) and MTM2014-55367-P (Ministerio de Economía y Competitividad and Fondo Europeo de Desarrollo Regional FEDER).

1. Forced integers

From now on, let $PF = \{g_1 < \cdots < g_n\}$. Then $\mathrm{F}(S) = g_n$ and $\mathrm{t}(S) = n$ if $S \in \mathcal{S}(PF)$.

We denote by $\mathcal{GF}(PF) = \bigcap_{S \in \mathcal{S}(PF)} \mathrm{H}(S)$ the set of *PF-forced gaps*. In a similar way, $\mathcal{EF}(PF) = \bigcap_{S \in \mathcal{S}(PF)} \mathrm{small}(S)$ is the set of *small PF-forced elements*. The union of the PF-forced gaps and small PF-forced elements will be the set of *PF-forced integers*. If $G \subseteq \mathcal{GF}(PF)$ and $E \subseteq \mathcal{EF}(PF)$, then we say that $v \in \{1, \ldots, g_n\} \setminus (G \cup E)$ is a *free integer for (G, E)*. Observe that, if $\mathcal{S}(PF) = \emptyset$, then $\mathcal{GF}(PF) = \mathcal{EF}(PF) = \mathbb{N}$. Thus, we have a simple, but crucial fact.

Proposition 1.1. *$\mathcal{S}(PF) \neq \emptyset$ if and only if $\mathcal{GF}(PF) \cap \mathcal{EF}(PF) = \emptyset$.*

1.1. **Starting forced gaps.** Let $\mathrm{m}(S) = \min(S \setminus \{0\})$ (the so-called *multiplicity* of S). From [4, Corollary 2], we have that $\mathrm{m}(S) \geq \mathrm{t}(S) + 1$. Therefore, $\{x \in \mathbb{N} \mid 1 \leq x \leq \mathrm{t}(S)\} \subseteq \mathrm{H}(S)$. Consequently, $\{1, \ldots, n\} \subseteq \mathcal{GF}(PF)$.

The partial order $\leq_S$ induced by the numerical semigroup S on the integers is defined as follows: $x \leq_S y$ if $y - x \in S$. The next result is well known and useful to get our purposes.

Lemma 1.2. [6, Lemma 2.19] *Let S be a numerical semigroup. Then*

1. $\mathrm{PF}(S) = \mathrm{Maximals}_{\leq_S}(\mathbb{Z} \setminus S)$;
2. *$x \in \mathbb{Z} \setminus S$ if and only if $f - x \in S$ for some $f \in \mathrm{PF}(S)$.*

The maximality of the pseudo-Frobenius numbers with respect to $\leq_S$ means that they are incomparable with respect to this ordering. This is the underlying idea of the next result. (If $A \subseteq \mathbb{N}$, then we denote by $\langle A \rangle$ the submonoid of $(\mathbb{N}, +)$ generated by A, that is, the set of non-negative integer linear combinations of elements of A.)

Lemma 1.3. *Let $\mathrm{PF}(S) = \{g_1 < \cdots < g_n\}$ with $n > 1$. If $2 \leq i \leq n$ and $g \in \langle \mathrm{PF}(S) \rangle$ with $g < g_i$, then $g_i - g \in \mathrm{H}(S)$. In particular, $\{g_i - g_j \mid i, j \in \{1, \ldots, n\}, i > j\} \subseteq \mathrm{H}(S)$.*

It is clear that $\mathrm{PF}(S) \subseteq \mathrm{H}(S)$ and that any positive divisor of a gap must be a gap also. Therefore, if we denote by $\mathrm{sfg}(PF)$ the set of positive divisors of $PF \cup \{1, \ldots, n\} \cup \{g_i - g \mid i \in \{2, \ldots, n\}, g \in \langle PF \rangle, g_i > g\}$, then $\mathrm{sfg}(PF) \subseteq \mathrm{H}(S)$ for all $S \in \mathcal{S}(PF)$. We will call *starting forced gap for PF* to any element of $\mathrm{sfg}(PF)$.

1.2. **Forced elements.** In this moment we can get forced elements in two ways. Firstly, we are going to compute the *big forced elements* ($\mathrm{bfe}(PF)$). We need the following result.

Lemma 1.4. *Let S be a numerical semigroup with $\mathrm{m}(S) = m$. Then, for each integer $i \in \{1, \ldots, m\}$, either $\mathrm{F}(S) - i \in S$ or $\mathrm{F}(S) - i \in \mathrm{PF}(S)$.*

Since we do not know the multiplicity of the semigroups we are looking for, in the above result, we can take m as the least positive integer that currently is not a forced gap.

Secondly, by using Lemma 1.2(2), we get the *elements forced by exclusion* ($\mathrm{efe}(PF)$).

Lemma 1.5. *Let FG be a set of forced gaps (with $PF \subseteq FG$).*

1. *Let $x \in FG$. Consider the set $H = \{h \in PF \mid h - x \geq 0 \text{ and } h - x \notin FG\}$. If $H = \{h\}$ for some positive integer h, then $h - x$ is a forced element.*
2. *Take $x \in \{1, \ldots, g_n - 1\} \setminus FG$. If there is no positive integer in $(-x + PF) \setminus FG$, then x is a forced element.*

If $\mathtt{fe}$ is a list of forced elements, then so is $\langle \mathtt{fe} \rangle \cap \{0, \ldots, g_n + 1\}$. Thus, we will take $\mathrm{fe}(PF) = \langle \mathrm{bfe}(PF) \cup \mathrm{efe}(PF) \rangle \cap \{0, \ldots, g_n + 1\} \subseteq \mathcal{EF}(PF)$.

1.3. **Futher forced gaps and elements.** Whenever we find a new forced element, it might produce new forced gaps. In effect, if e is an element and f is a gap, then $f - e$ is either negative or a gap. Thus, if $\mathtt{fg} \subseteq \mathcal{GF}(PF)$ and $\mathtt{fe} \subseteq \mathcal{EF}(PF)$, then $(\mathtt{fg} - e) \cap \mathbb{N} \subseteq \mathcal{GF}(PF)$ for every $e \in \mathtt{fe}$. Of course, new forced gaps might produce new forced elements, and so on.

1.4. **Stop conditions.** If they are not fulfilled, some conditions can be used to stop an algorithm, producing in this case $\mathcal{S}(PF) = \emptyset$.

By Lemmas 1.2 and 1.3, $g_k - (g_n - g_1) \geq 0$ for some $k \leq n - 1$. Thus, $g_1 \geq g_n - g_{n-1}$, condition with negligible computational cost which can avoid extra efforts to get $\mathcal{S}(PF) = \emptyset$.

From Lemma 1.2, we have that, for each x in $\mathrm{sfg}(PF) \setminus PF$, always there exists a positive integer in $(-x + PF) \setminus \mathrm{sfg}(PF)$.

By Proposition 1.1, at any step of the execution of an algorithm, the set of forced gaps cannot intersect the set of forced elements.

2. Computing forced integers

We present two procedures to compute forced integers. The first one is the fastest and appropriate to be used within a recursive function. The second one has a non-negligible cost in terms of time and its use in all the situations may not be the best option.

2.1. **Quick procedure.** With the contents of Section 1, we show an algorithm that computes a list containing forced gaps and forced elements.

Algorithm 2.1. Input: $\mathfrak{g}, \mathfrak{e}$ (sets of PF-forced gaps and PF-forced elements, respectively).
Output: $[\mathfrak{fg}, \mathfrak{fe}]$ (where $\mathfrak{fg} \supseteq \mathfrak{g}$ is a set of PF-forced gaps and $\mathfrak{fe} \supseteq \mathfrak{e}$ is a set of PF-forced elements) or `fail` (when some inconsistency is discovered).
repeat
Compute new gaps using Sections 1.1 and 1.3, storing them in $\mathfrak{fg}$;
Compute new elements using Sections 1.2 and 1.3, storing them in $\mathfrak{fe}$;
if some inconsistency arises (Section 1.4) **then return** `fail`;
until no new gaps or elements arise;
return $[\mathfrak{fg}, \mathfrak{fe}]$;

2.2. **Not so quick procedure.** Let G and E be sets of PF-forced gaps and PF-forced elements respectively, and let v be a free integer for (G, E). We say that v is *admissible for* (G, E) if Algorithm 2.1 does not return *fail* when applied to $(G, E \cup \{v\})$. Otherwise, v is a PF-forced gap and we say that is *non-admissible for* (G, E).

Algorithm 2.2. Input: $PF = \{g_1 < \ldots < g_n\}$, with $PF \neq \{f\}$ and $PF \neq \{f/2, f\}$.
Output: `fail` if some inconsistency is discovered; otherwise, returns $[\mathfrak{fg}, \mathfrak{fe}]$, where $\mathfrak{fg}$ and $\mathfrak{fe}$ are sets of forced gaps and forced elements, respectively.
fints := Algorithm 2.1 to $(\mathrm{sfg}(PF), \emptyset)$;
if fints = `fail` **then return** `fail`;
else if fints[1]$\cup$fints[2]$=\{0, \ldots, g_n\}$ **then return** fints;
newgaps := set of non-admissible integers for (fints[1], fints[2]);
return Algorithm 2.1 to (newgaps $\cup$ fints[1], fints[2]);

3. Question Q2: an algorithm to compute $\mathcal{S}(PF)$

Basically, the idea for an algorithm that computes $\mathcal{S}(PF)$ is to use a recursive function to construct a tree whose nodes are labeled by pairs (X, Y), where X is a list of forced gaps and Y is a list of forced elements. Moreover, the complement of $X \cup Y$ in the set $U = \{1, \ldots, g_n\}$ is a list of free integers. Nodes with an empty set of free integers are the leafs in our tree.

A node (X, Y) such that there exists a numerical semigroup $S \in \mathcal{S}(PF)$ for which $X \subseteq \mathrm{H}(S)$ and $Y \subseteq \mathrm{small}(S)$ is said to be PF-*feasible*. In other case we have a *dead node*.

Let $\mathcal{L}$ be a list of free integers for some set of forced integers. Then, for each integer $v \in \mathcal{L}$, we compute all numerical semigroups containing v and the forced elements, and having the forced gaps as gaps. We proceed recursively and use backtracking when a semigroup or a dead node is found. When we revisit the node, we then suppose that v is a gap and continue with the next free integer.

This algorithm has been implemented in the GAP package numericalsgps (see [2]).

4. Question Q1: particular cases

We have a full answer to question Q1 when the type is equal to two (see [5]).

Theorem 4.1. *Let g_1, g_2 be two positive integers such that $\frac{g_2}{2} < g_1 < g_2$. Let $\theta(g_1, g_2)$ be equal to $2g_1 - g_2$ if g_2 is odd, and equal to $g_1 - \frac{g_2}{2}$ if g_2 is even. Then there exists $S \in \mathcal{S}(\{g_1, g_2\})$ if and only if $\theta(g_1, g_2)$ does not divide any element of the set $\{g_1, g_2, g_2 - g_1\}$.*

Observe that there are differences in terms of the parity of g_n when type is equal to two. We have a similar situation for type equal to three.

Lemma 4.2. *If $\mathcal{S}(\{g_1, g_2, g_3 = g_1 + g_2\}) \neq \emptyset$, then g_3 is odd.*

The general problem is open. In fact, we only have partial results for type equal three.

Proposition 4.3. *Let α, β, g be positive integers.*

(1) *$\mathcal{S}(\{g, \alpha g, \beta g\}) \neq \emptyset$ if and only if g is odd, $\alpha = 2$ and $\beta = 3$.*
(2) *If $(\alpha + 1)g + \beta$ is odd and $\alpha > \beta$, then $\mathcal{S}(\{g, \alpha g + \beta, (\alpha + 1)g + \beta\}) \neq \emptyset$.*

Proposition 4.4. *If $g_1 + g_2 \geq \frac{3g_3+1}{2}$ and g_3 is odd, then $\mathcal{S}(\{g_1, g_2, g_3\}) \neq \emptyset$.*

References

[1] V. Blanco, J. C. Rosales, *The tree of irreducible numerical semigroups with fixed Frobenius number*, Forum Math. **25** (2013), 1249–1261.

[2] M. Delgado, P. A. García-Sánchez and J. Morais, "NumericalSgps", *a GAP package for numerical semigroups, Version 1.0.1*; 2015. Available via http://www.gap-system.org/.

[3] M. Delgado, P. A. García-Sánchez, and A. M. Robles-Pérez, *Numerical semigroups with a given set of pseudo-Frobenius numbers*, LMS J. Comput. Math. **19(1)** (2016), 186–205.

[4] R. Fröberg, C. Gottlieb, R. Häggkvist, *On numerical semigroups*, Semigroup Forum **35** (1987), 63–83.

[5] A. M. Robles-Pérez and J. C. Rosales, *The genus, the Frobenius number, and the pseudo-Frobenius numbers of numerical semigroups with type two*. To appear in Proc. Roy. Soc. Edinburgh Sect. A.

[6] J. C. Rosales and P. A. García-Sánchez, *Numerical semigroups*, Developments in Mathematics, vol. **20**, Springer, New York, 2009.

Department of Applied Mathematics, University of Granada
E-mail address: arobles@ugr.es

SIMPLICIAL EFFECTIVE HOMOTOPY

ANA ROMERO AND FRANCIS SERGERAERT

Abstract. The first invariants of Algebraic Topology are the homology and homotopy groups. The success of *effective* homology leads to study whether analogous methods could be applied to the homotopy groups. Because of the strong differences between their respective definitions, *effective* homotopy requires more sophisticated tools, less algebraic, more topological. This text explains the general framework of effective homotopy.

Introduction

The comparison between homology and homotopy groups starts as follows. The homotopy groups can be easily and naturally defined, but their calculation in general is hard. This was suddenly understood in 1931 when Heinz Hopf [3] proved $\pi_3(S^2) = \mathbb{Z}$, pointing out a surprising *internal* property of the 2-sphere in dimension... 3! This was only the initial step of a long story far from being finished: the calculation of the homotopy groups remains today one of the major problems of mathematics. On the contrary, the definition of the homology groups is not so easy, but their calculation is relatively simple, for example $H_n(S^2) = 0$ for every $n > 2$.

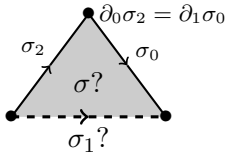
This amazing and strange duality has already been systematically studied [4]. Taking account of the methods of *effective homology* [9] and their success in some computations of homology groups [1], it is therefore necessary to study what *effective homotopy* could be.

The general framework of effective homotopy is described in this text. The homotopy groups are... groups, some algebraic objects, but their definition has a topological basis. In the category of topological spaces, the definition of the homotopy groups is natural and simple. The qualifier *effective* meaning here something which can be successfully processed on a computer, a *combinatorial* definition of the homotopy groups is necessary. It was really a *tour de force* by Daniel Kan [2] to obtain in the now standard simplicial framework the right combinatorial definition of the homotopy groups.

Kan's combinatorial definition of the homotopy groups is valid only for the simplicial sets satisfying an extension condition called the *Kan extension condition*, and the combinatorial spaces satisfying this condition are called *Kan spaces*; this is explained in Section 1. Kan's definition of the homotopy groups of his spaces is given in Section 2. These definitions are only "theoretical"; Section 3 defines the notion of *Effective Homotopy*, giving an explicit translation of Kan's definitions into a computational framework. Section 4 explains these computational definitions produce *effective* translations of Serre's exact sequences, so becoming *effective* tools allowing a user to *compute* homotopy groups. Finally, Section 5 is a brief report about a first computer implementation due to the first author.

1. Kan spaces

All our spaces are *simplicial sets* [6], a powerful generalization of the simplicial complexes, the details of which are not here really necessary. A *Kan space* X satisfies an extension condition which is easily understood in dimension 2. Let σ_2 and σ_0 be two 1-simplices (edges) of X satisfying the condition $\partial_0\sigma_2 = \partial_1\sigma_0$, in other words, the origin of σ_0 is the extremity of σ_2:



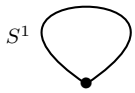
The Kan extension is satisfied for this configuration if there exists a 2-simplex σ such that the relations $\partial_0\sigma = \sigma_0$ and $\partial_2\sigma = \sigma_2$ hold; then the edge $\sigma_1 = \partial_1\sigma$ is a *composition* of the edges σ_2 and σ_0, not necessarily uniquely defined.

There is an obvious generalization of this extension condition to an arbitrary dimension n consisting in giving all the faces of a *hypothetical* n-simplex except one, all these pseudo-faces satisfying the necessary adjacency conditions; then the Kan condition is satisfied with respect to this configuration if an n-simplex does exist filling in the given collection of $(n-1)$-simplices. The last face which was missing is in a sense a composition of the given faces.

A simplicial set satisfying this extension condition for every n and for every coherent configuration of $(n-1)$-simplices is called a *Kan space*.

2. Homotopy groups of Kan spaces

The *origin* of an edge in a simplicial complex is always different from its *extremity*. In the more general setting of simplicial sets, the origin and the extremity of an edge can be the same, this edge then modelling a circle, a 1-sphere. More generally an n-simplex can have all its faces collapsed on a unique vertex of a simplicial set, modelling an n-sphere.



An *n-sphere* σ of a based Kan space $(X, *)$ is an n-simplex every face of which $\partial_i\sigma$, $0 \leq i \leq n$, is collapsed on the base point: $\partial_i\sigma = *$.

The Kan extension condition makes it possible to copy in this environment of Kan simplicial sets all the standard definitions of homotopy groups. In particular a *homotopy class* of dimension n is represented by an n-sphere modulo an appropriate *homotopy relation*. All these homotopy classes are then organized for $n \geq 1$ as a group $\pi_n(X, *)$, abelian as soon as $n \geq 2$.

If the space $(X, *)$ is "reasonable", see Serre's paper [10], then these homotopy groups are abelian groups of finite type, and their calculation has become one of the major problems of Algebraic Topology.

3. Effective Homotopy

We work now in a *computational* environment. A simplicial set is modelled as the type X of its simplices. A face operator ∂, included in the definition of the type, can compute a relevant *i-face* $\partial_i\sigma$ when $0 \leq i \leq \dim\sigma$. The number of simplices of a given dimension is not necessarily finite. The simplicial set X is *Kan* if an extra operator k is provided able to answer positively any extension problem as described in Section 1.

An abelian group of finite type can be coded as a finite sequence $(d_1, \ldots, d_g)$ of non-negative integers satisfying the *divisor condition*: d_i must divide d_{i+1}. The last elements of this sequence can be 0. The sequence $(d_1, \ldots, d_g)$ represents the group $\mathbb{Z}/d_1 \oplus \cdots \oplus \mathbb{Z}/d_g$. The group is finite if and only if $d_g > 0$. The sequence $(6, 0)$ represents the group $\mathbb{Z}/6 \oplus \mathbb{Z}$.

A simply connected Kan simplicial set X becomes a simplicial set *with effective homotopy* if an algorithm $EH : n \mapsto (G_n, f_n, g_n, h_n)$ is provided, valid for every integer $n \geq 2$, the following properties being satisfied.

- G_n is a group $(d_1, \ldots, d_g)$ depending on n.
- The component f_n is a functional object $S_n(X) \to G_n$ computing for every n-sphere σ of X an element $(a_1, \ldots, a_g) \in G_n$, where $0 \leq a_i < d_i$ when $d_i > 0$.
- The component g_n is a functional object $G_n \to S_n(X)$.
- The component h_n is a functional object $S^0_n(X) \to X_{n+1}$.

The group G_n is the standard canonical representation of the homotopy group $\pi_n(X)$. The component f_n *computes* the *homotopy class* $f_n(\sigma)$ of an n-sphere, some element of G_n. If two spheres are homotopic, their images by f_n are equal. In the reverse direction, the component g_n returns some (*effective*) n-sphere $g_n(a)$ representing some element $a \in G_n$. In particular, the relation $f_n \circ g_n = \mathrm{id}$ is satisfied. Finally, if f_n claims the homotopy class $f_n(\sigma)$ of the n-sphere σ is null, then $h_n(\sigma)$ is a *certificate* of this property, some $(n+1)$-simplex of X *effectively* describing a homotopy between σ and the trivial sphere $*$, the base point.

4. Exact Sequences of Homotopy Groups

The standard *exact sequences* of homotopy groups connect the homotopy groups of three spaces strongly related to each other. For example, if $F \to E \to B$ is a *fibration* of base space B, fibre space F and total space E, then the Serre exact sequence of the corresponding homotopy groups is:

$$\cdots \xrightarrow{j} \pi_{n+1}B \xrightarrow{\partial} \pi_n F \xrightarrow{i} \pi_n E \xrightarrow{j} \pi_n B \xrightarrow{\partial} \pi_{n-1}F \xrightarrow{i} \cdots$$

giving for example the short exact sequence:

$$0 \to \operatorname{coker}\partial \xrightarrow{i} \pi_n E \xrightarrow{j} \ker\partial \to 0 \tag{1}$$

As usual in this context, if $\ker\partial$ and $\operatorname{coker}\partial$ are known, a hard *extension problem* can remain pending to determine the unknown group $\pi_n E$. Typically, if $\ker\partial$ and $\operatorname{coker}\partial$ are $\mathbb{Z}/2$, is the central group $\mathbb{Z}/2 \oplus \mathbb{Z}/2$ or $\mathbb{Z}/4$?

The main theorem of *Effective Homotopy* [8] states that, if the base space B and the fibre space F are spaces *with effective homotopy*, then the extra data that are provided are sufficient to solve this extension problem. And the space E can be in turn provided with a structure of space with effective homotopy, allowing the user to use now E for example as a fibre space of another fibration, with the same role.

The same sorts of result are available for the other usual exact sequences of homotopy groups.

5. A concrete Implementation

Effective homotopy is not only a theoretical result. A first implementation has been written down, as an extra module [7] of the Kenzo program [1]. The best solution to handle the abelian groups of finite type in this context is not obvious. The minimal free resolutions corresponding to these groups have been chosen; for example the group (2) "=" $\mathbb{Z}/2$ is represented by the resolution $0 \to \mathbb{Z} \stackrel{\times 2}{\to} \mathbb{Z} \to 0$, for our group is the cokernel of the multiplication by 2.

An interesting process then makes it possible to determine a looked-for group extension. If $0 \to G_1 \to G \to G_2 \to 0$ is a short exact sequence with G_1 and G_2 given, then the extension G is determined by a *characteristic class* $\chi \in H^2(G_2, G_1)$. If $0 \to G_i'' \stackrel{\mu_i}{\to} G_i' \to G_i \to 0$ are free resolutions for G_i, $i = 1, 2$, then a correspondance between characteristic classes χ and integer matrices $\mu_{2,1}^{\chi} : G_2'' \to G_1'$ can be defined producing the resolution:

$$0 \to G_1'' \oplus G_2'' \xrightarrow{m} G_1' \oplus G_2' \to G \to 0$$

where m is the matrix $\begin{bmatrix} \mu_1 & \mu_{2,1}^{\chi} \\ 0 & \mu_2 \end{bmatrix}$. The key algorithm of the effective homotopy is then:

"Effective Homotopy" $\mapsto \chi \mapsto \mu_{2,1}^{\chi} \mapsto$ resolution of $G \mapsto$ minimal resolution of G

See [7] for details.

References

[1] Xavier Dousson, Julio Rubio, Francis Sergeraert and Yvon Siret. *The Kenzo program.* `http://www-fourier.ujf-grenoble.fr/~sergerar/Kenzo/`

[2] Daniel M. Kan. *A combinatorial definition of homotopy groups.* Annals of Mathematics. 1958, vol. 67, pp. 282-312.

[3] Heinz Hopf (1931), *Über die Abbildungen der dreidimensionalen Sphäre auf die Kugelfläche*, Mathematische Annalen, 1931, vol.104, pp.637-665.

[4] Peter Hilton. *Homotopy theory and duality.* Gordon and Breach, 1965.

[5] Stanley O. Kochman, *Stable Homotopy Groups of Spheres, a Computer-Assisted Approach*, Springer, LNM #1423.

[6] J. Peter May. *Simplicial objects in algebraic topology.* Van Nostrand, 1967.

[7] Ana Romero, Julio Rubio, Francis Sergeraert. *An implementation of effective homotopy of fibrations.* Preprint.

[8] A. Romero and F. Sergeraert. *Effective homotopy of fibrations.* Applicable Algebra in Engineering, Communication and Computing, 2012, vol.23, pp.85-100.

[9] Julio Rubio, Francis Sergeraert. *Constructive Algebraic Topology.* Bulletin des Sciences Mathématiques, 2002, vol. 126, pp. 389-412.

[10] Jean-Pierre Serre. *Groupes d'homotopie et classes de groupes abéliens.* Annals of Mathematics, 1953, vol. 58, pp. 258-294.

Universidad de La Rioja, Departamento de Matemáticas y Computación, Logroño, Spain.
E-mail address: `Ana.Romero@unirioja.es`

Université Grenoble-Alpes, Institut Fourier, Grenoble, France.
E-mail address: `Francis.Sergeraert@ujfgrenoble.fr`

VERIFICACIÓN DE LA EFICIENCIA DE CÓDIGOS DS-LDPC APLICADOS A LA PROTECCIÓN DE MEMORIAS DE ALTA VELOCIDAD

ROSARIO RUBIO AND M. PILAR VÉLEZ

Abstract. In this paper we prove, through symbolic methods, a conjecture on error detection codes raised in the field of microelectronics. This result simplifies the process of detection of errors in DS-LDPC codes when the number of errors of the received word is at most 5. Instead of checking N cycles, where N is the length of the code, we prove that checking 3 consecutive ones is enough. This result had been studied for the case of 2 errors and extended afterwords to the case of 4 errors by trying to check exhaustively each possible combination without success in all cases. The proof we present is an alternative one that involves all cases up to five errors in codes applied to memories and solves all of them.

Introducción

La fiabilidad de los componentes electrónicos es un problema creciente debido a la miniaturización de estos. El control de los *soft errors* que se producen cuando una partícula impacta sobre el circuito y altera su funcionamiento es uno de los retos a resolver para mejorar la fiabilidad de los circuitos [1]. Las memorias incorporan códigos de detección y corrección de errores para evitar que esto afecte al funcionamiento del sistema [2]. Para mejorar la velocidad del decodificador se ha explorado la posibilidad de utilizar las primeras fases de la decodificación para detectar errores de modo que si no hay errores la decodificación se termina antes de tiempo. Por ahora la técnica se ha probado experimentalmente para los códigos basados en conjuntos de diferencias perfectas, códigos DS-LDPC (*Difference Set Low Density Parity Check*) [4, 5]. Los códigos basados en conjuntos de diferencias perfectas DS-LDPC son códigos binarios, cíclicos y lineales.

La probabilidad de error en memorias protegidas por códigos DS-LDPC es muy baja [6]. Además, en la mayoría de escenarios reales, las palabras erróneas sufren un número limitado y bajo de *bit flips* (cambios de valor en un bit). Por tanto, este trabajo se centra en la detección de errores que afectan hasta cinco bits. La detección de errores en una palabra en estos códigos se basa en la verificación de ciertas relaciones, denominadas *check sums*, en todos sus ciclos. Una vez definidas las *check sums* y su utilidad, resulta sencillo comprobar que cuando una palabra está afectada por un número impar de errores, se detecta en el primer ciclo. Por lo tanto se analizan aquí los casos de 2 y 4 errores.

En [4, 5] los autores conjeturan que evaluar las *check sums* en 3 ciclos consecutivos es suficiente para detectar 2 o 4 errores en los códigos DS-LDPC que se usan en memorias (es decir, con longitud de código $N = 73, 273, 1057$). Además, los mismos autores, tratan

de verificar de forma exhaustiva la conjetura generando todas las $\binom{N}{i}$, $i = 2, 4$ posibles combinaciones de dos y cuatro errores, pero no lo consiguen completamente. Para 2 errores y longitudes de código $N = 73, 273, 1057$ y para 4 errores y longitudes $N = 73, 273$, consiguen verificar todas las combinaciones posibles. Sin embargo, para 4 errores y $N = 1057$ consiguen llegar hasta mil millones de las $\binom{1057}{4} = 51.911.764.520$ combinaciones posibles.

En este trabajo, se resuelve de forma simbólica, con ayuda del sistema de cálculo simbólico Maple, la conjetura de los tres ciclos para 2 y 4 errores en códigos DS-LDPC de longitudes $N = 21, 73, 273, 1057$.

1. Códigos DS-LDPC en memorias

Un código DS-LDPC, C, de longitud N es un subespacio vectorial k-dimensional de $\mathbb{F}_2^N$ sobre $\mathbb{F}_2$ que está caracterizado por las *check sums*, un conjunto de relaciones dadas por polinomios generados por un conjunto de diferencias perfectas.

Definición 1.1. Sea $P = \{l_0, \ldots, l_q\} \subset \mathbb{Z}^+ \cup \{0\}$ donde $0 \leq l_0 < \cdots < l_q \leq N-1$, $q = 2^s$, $N = q(q+1)+1$. P es un *conjunto de diferencias perfectas* si las diferencias $l_j - l_k$ módulo N, con $j \neq k$, son todas diferentes.

Nótese que la definición anterior es equivalente a decir que las anteriores diferencias toman una, y sólo una vez, los valores del conjunto $\{1, 2, \ldots, N-1\}$ módulo N.

Sea $x \in C$ un mensaje codificado. Debido a causas externas, x puede sufrir uno o varios *bit flips* dando lugar a una palabra errónea. Si y es la palabra recibida, el número de *bit flips* coincide con el número de 1 del vector error $e = y - x$, que se denota por $e = e_0 e_1 \cdots e_{N-1}$.

Dado un conjunto de diferencias perfectas P con $l_0 = 0$, se define el polinomio:
$w_0 = X^{N-1} + X^{N-1-l_1} + X^{N-1-l_2} + \cdots + X^{N-1-l_q}$

y la *check sum* asociada a w_0 evaluada en e como: $A_0 = \sum_{j=0}^{q} e_{N-1-l_j} \mod 2$

Obsérvese que si $y \in C$, entonces $A_0 = 0 \mod 2$, y por la linealidad de esta definición $A_0 = \sum_{j=0}^{q} y_{N-1-l_j} \mod 2$. Pero en la literatura se utiliza la notación anterior con el vector error [3].

Sumando $l_1 \mod N$ a los exponentes de w_0 se genera w_1:
$w_1 = X^{l_1-1} + X^{N-1} + X^{N-1-(l_2-l_1)} + \cdots + X^{N-1-(l_q-l_1)}$

Siguiendo el mismo procedimiento se genera el conjunto de $q+1$ polinomios,

$w_k = X^{l_k-1} + X^{l_k-l_1-1} + \cdots + X^{l_k-l_{k-1}-1} + X^{N-1} + X^{N-1-(l_{k+1}-l_k)} + \cdots + X^{N-1-(l_q-l_k)}$
con $0 \leq k \leq q$ y sus *check sums* asociadas.

Si $\mathcal{E}_j$ es un ciclo de e, se denota por $w_k(\mathcal{E}_j)$ la *check sum* asociada a w_k en $\mathcal{E}j$. Entonces

$$y \in C \Longleftrightarrow w_k(\mathcal{E}_j) = 0 \mod 2, \forall\ 0 \leq k \leq q,\ \forall\ 1 \leq j \leq N \tag{1}$$

2. Teorema principal

El teorema principal afirma que para códigos con 2 o 4 errores la condición (1) se reduce a evaluar sólo tres ciclos consecutivos. Para su demostración se introducen las ecuaciones lineales en diferencias perfectas. Esto es, ecuaciones lineales con coeficientes en $\mathbb{Z}_N$ cuyas soluciones son diferencias de elementos en P. Además se desarrolla un algoritmo en Maple que calcula el conjunto de soluciones de una ecuación lineal en diferencias perfectas. Este algoritmo utiliza el algoritmo extendido de Euclides y se ejecuta sin problemas, pero se ralentiza al escribir las soluciones (el número de soluciones es como máximo $(q+1)^r(2q+1)$).

Teorema 2.1. *Sea C un código DS-LDPC de longitud $N = 21, 73, 273, 1057$. Sea $x \in C$ tal que la palabra recibida y contiene 2 o 4 errores. Entonces, la evaluación de las* check sums *del código en tres ciclos consecutivos de y es suficiente para detectar los errores.*

Esbozo de la demostración: Supongamos que los errores en y no se detectan en tres ciclos consecutivos, entonces se obtiene un sistema lineal en diferencias perfectas que describe la situación. La demostración consiste en comprobar que el sistema es incompatible.

Si y tiene 2 errores, utilizando que la ecuación $l_i - l_j = b,\ b \neq 0$ tiene solución única en P, llegamos a demostrar que el sistema no tiene solución.

Si y tiene 4 errores, se pueden dar tres situaciones diferentes. Emparejando las posiciones de los errores por pares, cada par puede aparecer en:

Caso 1: El mismo polinomio a lo largo de los tres ciclos.
En este caso el sistema está formado por dos subsistemas similares al de dos errores.

Caso 2: El mismo polinomio en dos de los tres ciclos y en el otro ciclo se mezclan.
Una sencilla manipulación del sistema lleva a que los 4 errores comparten el mismo polinomio en el ciclo en el que se intercambian. Y por lo tanto se reduce al Caso 1.

Caso 3: El mismo polinomio una única vez a lo largo de los tres ciclos.
Obtenemos el siguiente sistema *sparse* con restricciones:

$$\begin{cases} l_{f_{10}} - l_{c_{10}} = l_{f_{11}} - l_{c_{11}} - 1 \\ l_{f_{11}} - l_{c_{11}} = l_{f_{12}} - l_{c_{12}} - 1 \\ l_{f_{10}} - l_{c_{20}} = l_{f_{21}} - l_{c_{21}} - 1 \\ l_{f_{21}} - l_{c_{21}} = l_{f_{22}} - l_{c_{22}} - 1 \\ l_{f_{30}} - l_{c_{30}} = l_{f_{11}} - l_{c_{31}} - 1 \\ l_{f_{11}} - l_{c_{31}} = l_{f_{22}} - l_{c_{32}} - 1 \\ l_{f_{30}} - l_{c_{40}} = l_{f_{21}} - l_{c_{41}} - 1 \\ l_{f_{21}} - l_{c_{41}} = l_{f_{12}} - l_{c_{42}} - 1 \end{cases} \qquad \begin{bmatrix} l_{f_{10}} - l_{c_{10}} \neq N, l_{f_{11}} - l_{c_{11}} \neq N, \dots . \\ l_{c_{10}} \neq l_{c_{20}}, l_{c_{30}} \neq l_{c_{40}}, l_{c_{11}} \neq l_{c_{31}}, \dots \\ l_{f_{10}} \neq l_{f_{30}}, l_{f_{11}} \neq l_{f_{21}} y l_{f_{12}} \neq l_{f_{22}}. \end{bmatrix}$$

El objetivo es demostrar que no tiene solución. Para ello se ha diseñado un segundo algoritmo en Maple en el que, una vez reordenadas las ecuaciones, se combina la solución general de cada ecuación con las restricciones asociadas para ir eliminando paso a paso los posibles candidatos a solución. Ejecutando el algoritmo para $N = 21, 73, 273$ y 1057, se obtiene que el sistema no tiene solución.

Observaciones finales: Este trabajo es un ejemplo de colaboración interdisciplinar. El Grupo de Diseño Electrónico de la Universidad Antonio de Nebrija en una investigación sobre diseño de circuitos tolerantes a fallos encontraba reiteradamente, en simulaciones experimentales, que se podían detectar los errores en tres ciclos consecutivos. Intentaron comprobarlo con un análisis exhaustivo de todos los posibles casos de errores, pero no consiguieron completarlo. Pidieron nuestra colaboración para encontar una demostración de su conjetura.

Tras comprobar que en el Caso 3 con la manipulación de las ecuaciones no llegábamos a la incompatibildiad del sistema. Nuestro primer intento fue resolverlo por el método de Gauß y mediante bases de Gröbner utilizando los sistemas de álgebra computacional Maple y Cocoa y el sistema de cálculo numérico Matlab. En ningún caso se obtuvo un resultado debido al elevado número de variables y restricciones. Finalmente queda evidenciado que mediante una preparación simbólica de las ecuaciones se consigue ayudar a la máquina y a así probar la conjetura que en principio se abordó comprobando millones de casos.

References

[1] Baumann R. C.: Radiation-induced soft errors in advanced semiconductor technologies, IEEE Trans. On Device and Materials Reliability, Vol. 5, No. 3, pp. 301-316 (2005).
[2] Chen C. L., Hsiao M. Y.: Error-correcting codes for semiconductor memory applications: a state-of-the-art review", IBM Journal of Research and Development 28(2), pp. 124-134 (1984).
[3] Lin S., Costello D.J.: Error control coding. Prentice Hall (2004).
[4] Liu S.: A new methodology to systemize the design of fault-tolerant circuits based on system knowledge. PhD thesis, Universidad Antonio de Nebrija (2011).
[5] Liu S., Reviriego P., Maestro J.A.: Efficient majority logic fault detection with difference-set codes for memory applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems 20, 148–156 (2011).
[6] Reviriego P., Maestro J.A., Baeg S., Wen S., Wong R.: Protection of memories suffering MCUs through the selection of the optimal interleaving distance. IEEE Transactions on Nuclear Science 57, No. 4, 2124–2128 (2010).

Escuela Politécnica Superior, Universidad Antonio de Nebrija
E-mail address: mrubio@nebrija.es
E-mail address: pvelez@nebrija.es

GRÖBNER BASES FOR $I_1(XY)$

JOYDIP SAHA, INDRANATH SENGUPTA, AND GAURAB TRIPATHI

Abstract. Let K be a field and X, Y denote matrices such that, the entries of X are either indeterminates over K or 0 and the entries of Y are indeterminates over K which are different from those appearing in X. We consider ideals of the form $I_1(XY)$, which is the ideal generated by the 1×1 minors of the matrix XY. Our aim is to compute Gröbner bases for ideals of the form $I_1(XY)$, where Y is a generic column matrix and X is either generic or symmetric or anti-symmetric.

Introduction

Let K be a field and $\{x_{ij};\, 1 \leq i \leq m,\, 1 \leq j \leq n\}$, $\{y_j;\, 1 \leq j \leq n\}$ be indeterminates over K. Let $R = K[x_{ij}]$ and $S = K[x_{ij}, y_j]$ denote the polynomial algebras over K. Let X denote an $m \times n$ matrix such that its entries belong to the ideal $\langle \{x_{ij};\, 1 \leq i \leq m,\, 1 \leq j \leq n\} \rangle$. Let $Y = (y_j)_{n \times 1}$ be the generic $n \times 1$ column matrix.

Ideals generated by minors of a matrix appear in various contexts. Suppose that $X = (x_{ij})_{m \times n}$ is the generic matrix over K. The ring $R = K[x_{ij}]$ can be seen as the coordinate ring of the vector space of $m \times n$ matrices with entries in K. Let $\mathcal{I}_r \subseteq S$ denote the vanishing ideal of the affine subvariety of matrices of rank at most r, where $r < \min(m, n)$. Sturmfels [6] proved that, the set of $(r+1) \times (r+1)$ minors of X is the reduced Gröbner basis of the ideal $\mathcal{I}_r$ with respect to the lexicographic order $x_{1,n} > x_{1,n-1} > \cdots > x_{1,1} > x_{2,n} > x_{2,n-1} > \cdots > x_{2,1} > \cdots > x_{m,n} > x_{m,n-1} > \cdots > x_{m,1}$.

Ideals of the form $I + J$, where I and J are polynomial ideals generated by the minors of matrices with entries in the polynomial ring appear in various contexts, see [2], [3], [5], [1]. In order to study their syzygies and Betti numbers, the *Iterated Mapping Cone* is a standard technique which has been used by many authors, see [4], [2], [3]. However, in order to use this technique, one has to understand the successive colon ideals between I and J, which in general are not easy to compute. It is often helpful if Gröbner bases for I or J are known.

Let $I_1(XY)$ denote the ideal generated by the 1×1 minors or the entries of the $m \times 1$ matrix XY. Ideals of the form $I_1(XY)$ demand special attention because they occur as one of the summands in $I + J$ in several geometric considerations like linkage and generic residual intersection of polynomial ideals, especially in the context of syzygies. In a similar vein, Bruns-Kustin-Miller [1] resolved the ideal $I_1(XY) + I_{\min(m,n)}(X)$, where X is a generic $m \times n$ matrix and Y is a generic $n \times 1$ matrix. Johnson-McLoud [5] proved certain properties

The first author thanks UGC for the Senior Research Fellowship.

The second author is the corresponding author, who is supported by the the research project IP/IITGN/MATH/IS/201415-13.

The third author thanks CSIR for the Senior Research Fellowship.

for the ideals of the form $I_1(XY) + I_2(X)$, where X is a generic symmetric matrix and Y is either generic or generic alternating.

Let us assume that $m = n$. Our aim in this article is to compute Gröbner bases for the ideal $\mathcal{I} = I_1(XY) = \langle g_1, g_2, \ldots, g_n \rangle$, under the following conditions:

(1) X is generic;
(2) X is generic symmetric;
(3) X is generic antisymmetric.

1. Results

Theorem 1.1 (**Generic**). *Let $S = K[x_{ij},\, y_j \mid 1 \leq i, j \leq n]$ denote the polynomial K-algebra. Suppose that*

$$X \quad = \quad (x_{ij})_{n\times n} \quad = \quad \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} \end{pmatrix}$$

and $Y = (y_j)_{n\times 1}$ are generic. Let $\mathcal{I} = I_1(XY) = \langle g_1, g_2, \cdots, g_n \rangle$, where $g_i = \sum_{j=1}^{n} x_{ij}y_j$. The set $\{g_1, \cdots, g_n\}$ forms a Gröbner basis for the ideal $\mathcal{I}$, with respect to any monomial order which satisfies

(1) $x_{11} > x_{22} > \cdots > x_{nn}$;
(2) $x_{ij}, y_j < x_{nn}$ *for every* $1 \leq i \neq j \leq n$.

Theorem 1.2 (**Symmetric**). *Let $S = K[x_{ij},\, y_j \mid 1 \leq i \leq j \leq n]$ denote the polynomial K-algebra. Suppose that*

$$X \quad = \quad \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{12} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{nn} \end{pmatrix}$$

is generic symmetric and $Y = (y_j)_{n\times 1}$ is generic. Let $\mathcal{I} = I_1(XY) = \langle g_1, g_2, \cdots, g_n \rangle$, where $g_1 = \sum_{j=1}^{n} x_{1j}y_j$, $g_n = (\sum_{1\leq k\leq n} x_{ki}y_k)$ and $g_i = (\sum_{1\leq k<i} x_{ki}y_k) + (\sum_{i\leq k\leq n} x_{ik}y_k)$ for $1 < i < n$. The set $\{g_1, \cdots, g_n\}$ forms a Gröbner basis for the ideal $\mathcal{I}$, with respect to any monomial order which satisfies

(1) $x_{11} > x_{22} > \cdots > x_{nn}$;
(2) $x_{ij}, y_j < x_{nn}$ *for every* $1 \leq i < j \leq n$.

Theorem 1.3 (**Anti-symmetric**). *Let $S = K[x_{ij},\, y_j \mid 1 \leq i < j \leq n]$ denote the polynomial K-algebra. Suppose that*

$$X \quad = \quad \begin{pmatrix} 0 & x_{12} & \cdots & x_{1n} \\ -x_{12} & 0 & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ -x_{1n} & -x_{2n} & \cdots & 0 \end{pmatrix}$$

is generic anti-symmetric and $Y = (y_j)_{n\times 1}$ *is generic. Let* $\mathcal{I} = I_1(XY) = \langle g_1, g_2, \cdots, g_n\rangle$, *where*

$$g_1 = \sum_{i=2}^{n} x_{1i}y_i, \quad g_n = -\left(\sum_{1\le k<n} x_{ki}y_k\right)$$

and

$$g_i = -\left(\sum_{1\le k<i} x_{ki}y_k\right) + \left(\sum_{i<k\le n} x_{ik}y_k\right) \quad \text{for} \quad 1 < i < n.$$

Consider any monomial order satisfying $x_{12} > x_{23} > \cdots > x_{(n-1)n} > x_{13} > \cdots > x_{(n-2)n} > x_{14} > \cdots > x_{1n} > y_1 > \cdots > y_n$. *Let* $S_{(i_2,i_1)}$ *denote the S-polynomial* $S(g_{i_2}, g_{i_1})$. *The S-polynomial* $S_{(i_k,i_{k-1},\cdots,i_i)} = S(g_{i_k}, S_{(i_{k-1},i_{k-2},\cdots,i_i)})$ *is defined iteratively. Then, with respect to the monomial order defined above,*

(i) $S = \{g_1, \cdots g_n, S_{(3,1)}, S_{(5,3,1)}, \cdots, S_{((2[\frac{n}{2}]-1),\cdots,3,1)}\}$ *forms a Gröbner basis, if* $n \ge 4$;
(ii) $S = \{g_1, \cdots g_n\}$ *forms a Gröbner basis if* $n = 2, 3$.

2. Sketch of proofs

The proofs are direct applications of Buchberger's algorithm. The structures of the ideals are quite special, although they do not help us much, as far as the computations are concerned. The subsequent work on Betti numbers of these ideals uses the linearity of the generators in two sets of variables more effectively. It throws more light on the structure of these ideals as well.

Definition 2.1. Let $T \subset R$ be a set of monomials. We define

$$\text{supp}(T) = \{(i,j,0) \mid x_{ij}|m \quad \text{for some} \quad m \in T\} \cup \{(0,0,k) \mid y_k|m \quad \text{for some} \quad m \in T\}.$$

If $T = \{m\}$, then we write supp(m) instead of supp$(\{m\})$.

Theorem 2.2. *Let* $S = \{h_1, \cdots h_r\} \subset R$ *and* $I = \langle h_1, \cdots h_r\rangle$ *be an ideal in* R. *Suppose that* $\text{supp}(\text{Lt}(h_i)) \cap \text{supp}(\text{Lt}(h_j)) = \phi$, *for all* $1 \le i \ne j \le r$, *with respect to some monomial order. Then* $S(h_i, h_j) \longrightarrow 0$ *modulo* S.

The cases of Generic and Generic Symmetric follow from the above theorem and Buchberger's algorithm. However, the case of Generic Anti-symmetric is not so straightforward. The following are the main theorems leading to a proof in the case of Generic Anti-symmetric.

Theorem 2.3. $S_{(1,(2k-1),\cdots,3,1))} \longrightarrow 0$ *modulo* $S_k = \{g_1, \cdots, g_n, S_{(3,1)}, \cdots S_{((2k-1),\cdots,3,1)}\}$, *for each* $2 \le k \le [\frac{n}{2}]$.

Theorem 2.4. $S_{((2i)(2k-1),\cdots,3,1))} \longrightarrow 0$ *modulo* $S_k = \{g_1, \cdots, g_n, S_{(3,1)}, \cdots S_{((2k-1),\cdots,3,1)}\}$ *for each* $1 \le i \le k$ *and* $2 \le k \le [\frac{n}{2}]$.

References

[1] W. Bruns, A.R. Kustin, M. Miller, The Resolution of the Generic Residual Intersection of a Complete Intersection, *Journal of Algebra* 128 (1990) 214-239.

[2] P. Gimenez, I. Sengupta and H. Srinivasan, Minimal free resolution for certain affine monomial curves. In: *Commutative Algebra and its Connections to Geometry* (*PASI 2009*), A. Corso and C. Polini Eds, Contemp. Math. 555 (Amer.Math. Soc., 2011) 87–95.

[3] P. Gimenez, I. Sengupta and H. Srinivasan, Minimal graded free resolution for monomial curves defined by arithmetic sequences, *Journal of Algebra* 388 (2013) 294-310.

[4] J. Herzog, Y. Takayama, Resolutions by mapping cones, *Homology Homotopy Appl.* 4(2002) 277–294.

[5] M.R., Johnson, J. McLoud-Mann, On equations defining Veronese Rings, *Arch. Math. (Basel)* 86(3)(2006) 205-210.

[6] B. Sturmfels, Gröbner bases and Stanley decompositions of determinantal rings, *Math. Zeitschrift* 205(1990) 137 - 144.

Department of Mathematics, RKM Vivekananda University, Belur Math, Howrah 711202, India.
E-mail address: saha.joydip56@gmail.com

Discipline of Mathematics, IIT Gandhinagar, VGEC Campus, Visat-Gandhinagar Highway, Chandkheda, Ahmedabad, Gujarat 382424, INDIA.
E-mail address: indranathsg@iitgn.ac.in

Department of Mathematics, Jadavpur University, Kolkata, WB 700 032, India.
E-mail address: gelatinx@gmail.com

ALGEBRAIC ASPECTS OF RADICAL PARAMETRIZATIONS

J. RAFAEL SENDRA, DAVID SEVILLA, AND CARLOS VILLARINO

Abstract. We develop an algebraic foundation for working with radical parametrizations of varieties. As a motivation we show partial results on how to reparametrize a radical parametrization to make it rational, when it is unirational.

Introduction

It is well known that in many applications dealing with geometric objects, parametric representations are very useful. In this article we continue the exploration of radical parametrizations initiated in [3] and [4]. Here we introduce a framework to manipulate these parametrizations in a rational way by means of rational auxiliary varieties and maps. This allows us to apply results of algebraic geometry to derive conclusions on the radical parametrization and its image. In short, to translate radical statements into rational ones.

In Section 1 we make the intuitive definition of radical parametrization into a concrete object within the realm of field extensions, and develop the basic tools to manipulate it. Section 2 contains some results, among other things, on the problem of reparametrizing a radical parametrization to make it rational, when possible.

1. Basic concepts

A radical parametrization is, intuitively speaking, a tuple $(x_1(\bar{t}),\dots,x_r(\bar{t}))$ of functions of $\bar{t}=(t_1,\dots,t_n)$ which are constructed by rational operations and roots of any index. Formally, a radical parametrization is a tuple of elements of a radical extension of the field $\mathbb{C}(\bar{t})$ of rational functions in $\bar{t}$. From now on we assume that $t_1,\dots,t_n$ are algebraically independent over the field $\mathbb{C}$ of the complex numbers.

Definition 1.1. A *radical tower over* $\mathbb{C}(\bar{t})$ is a tower of field extensions $\mathbb{F}_m \supset \cdots \supset \mathbb{F}_0 = \mathbb{C}(\bar{t})$ such that $\mathbb{F}_i = \mathbb{F}_{i-1}(\delta_i)$ with $\delta_i^{e_i} = \alpha_i \in \mathbb{F}_{i-1}$, $e_i \in \mathbb{N}$. In particular, $\mathbb{C}(\bar{t})$ is a radical tower over itself. A *radical parametrization* is a tuple $(x_1(\bar{t}),\dots,x_r(\bar{t}))$ of elements of the last field $\mathbb{F}_m$ of some radical tower over $\mathbb{C}(\bar{t})$, and such that their Jacobian has rank n.

Remark 1.2. The Jacobian is defined by extension of the canonical derivations $\frac{\partial}{\partial t_i}$ from $\mathbb{C}(\bar{t})$ to $\mathbb{F}_m$ (see [6, II, Section 17, Cor. 2] for details).

The following example relates the usual way of writing roots with Definition 1.1.

The authors are partially supported by the Spanish *Ministerio de Economía y Competitividad* under Project MTM2014-54141-P, and by Junta de Extremadura and FEDER funds. The first and third authors are members of the research group ASYNACS (Ref. CCEE2011/R34). The second author is member of the research group GADAC (Ref. FQM024).

Example 1.3. From the expresion $(x(t),y(t)) = \left(t + \frac{1-\sqrt[4]{t^3+2t}}{\sqrt{t^2-\sqrt[3]{t-1}}}, \frac{\sqrt[4]{5\sqrt[3]{t-1}+1}}{t^3+5}\right)$, let us generate a radical parametrization. We consider the tower

$$\mathbb{T} \equiv \left[\begin{array}{l} \mathbb{F}_0 \subset \mathbb{F}_1 := \mathbb{F}_0(\delta_1) \subset \mathbb{F}_2 := \mathbb{F}_1(\delta_2) \subset \mathbb{F}_3 := \mathbb{F}_2(\delta_3) \subset \mathbb{F}_4 := \mathbb{F}_3(\delta_4), \\ \text{where } \delta_1^3 = t-1,\ \delta_2^2 = t^2-\delta_1, \delta_3^4 = 5\delta_1+1,\ \delta_4^4 = t^3+2t \end{array} \right].$$

Then, one possible radical parametrization is (here the roots denote real positive values)

$$\mathcal{P} = \left\{ \left(t + \tfrac{1-\delta_4}{\delta_2}, \tfrac{\delta_3}{t^3+5}\right), \mathbb{T}, \delta_1(2) = 1, \delta_2(2) = -\sqrt{3}, \delta_3(2) = i\sqrt[4]{6}, \delta_4(2) = \sqrt[4]{12} \right\}$$

Definition 1.4. Let $\mathcal{P} = (x_1(\bar{t}), \ldots, x_r(\bar{t}))$ be a radical parametrization where

$$x_i(\bar{t}) = \frac{x_{iN}(\bar{t}, \delta_1, \ldots, \delta_m)}{x_{iD}(\bar{t}, \delta_1, \ldots, \delta_m)},\ \delta_i^{e_i} = \alpha_i \text{ and } \alpha_1 = \frac{\alpha_{1N}(\bar{t})}{\alpha_{1D}(\bar{t})},\ \ldots,\ \alpha_m = \frac{\alpha_{mN}(\bar{t}, \delta_1, \ldots, \delta_{m-1})}{\alpha_{mD}(\bar{t}, \delta_1, \ldots, \delta_{m-1})}.$$

We construct the *incidence variety $\mathcal{B}_\mathcal{P}$ associated to this representation of $\mathcal{P}$* as follows (note that it depends not only on $\mathcal{P}$ but also the way we write it): in the ring with coefficients in $\mathbb{C}$ over the variables $\overline{T}, \Delta_1, \ldots, \Delta_m, X_1, \ldots, X_r, Z$ we define the polynomials:

$$\begin{aligned}
&E_1 = (\Delta_1)^{e_1} \cdot \alpha_{1D}(\overline{T}) - \alpha_{1N}(\overline{T}),\\
&E_i = (\Delta_i)^{e_i} \cdot \alpha_{iD}(\overline{T}, \Delta_1, \ldots, \Delta_{i-1}) - \alpha_{iN}(\overline{T}, \Delta_1, \ldots, \Delta_{i-1}), \qquad i = 2, \ldots, m,\\
&G_j = X_j \cdot x_{jD}(\overline{T}, \Delta_1, \ldots, \Delta_m) - x_{jN}(\overline{T}, \Delta_1, \ldots, \Delta_m), \qquad j = 1, \ldots, r.\\
&G_Z = Z \cdot \operatorname{lcm}(x_{1D}, \ldots, x_{rD}) \cdot \operatorname{lcm}(\alpha_{1D}, \cdots, \alpha_{mD}) - 1.
\end{aligned}$$

Then we define $\mathcal{B}_\mathcal{P}$ as the zeroset of $\{E_1, \ldots, E_m, G_1, \ldots, G_r, G_Z\}$ in $\mathbb{C}^{n+m+r+1}$.

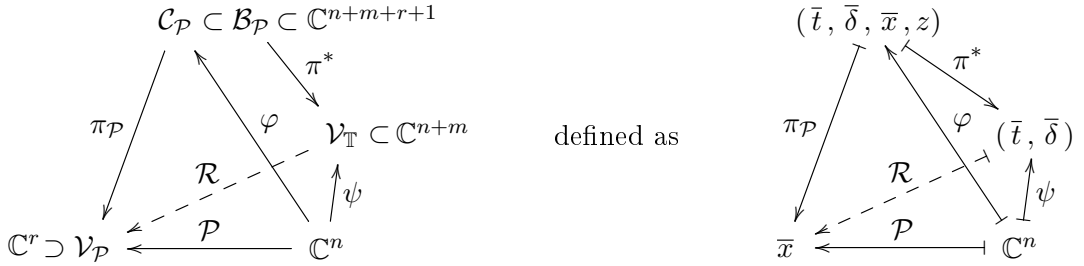
The variety $\mathcal{B}_\mathcal{P}$ contains points related to the conjugates of the δ_i, since the defining polynomials E_i do not discriminate those. By fixing the branches of the δ_i, for instance as in Example 1.3, one particular component of $\mathcal{B}_\mathcal{P}$ is determined, as we will see next.

Let $D = \{\bar{t} \in \mathbb{C}^n : \text{for some } i,\ \delta_i(\bar{t}) = 0 \text{ or undefined}\}$ and let $\psi : \mathbb{C}^n \setminus D \to \mathbb{C}^{n+m}$ be defined as $\psi(\bar{t}) = (\bar{t}, \delta_1(\bar{t}), \ldots, \delta_m(\bar{t}))$. Let $\pi^\star$ be the projection from $\mathcal{B}_\mathcal{P}$ onto its $n+m$ first coordinates; it is injective, thus $(\pi^\star)^{-1}$ is well defined in $D' = \mathbb{C}^{n+m} \setminus V(x_{1D} \cdots x_{rD})$. Finally let $\varphi = (\pi^\star)^{-1} \circ \psi$ with domain $D'' = (\mathbb{C}^n \setminus D) \cap \{\bar{t} : \psi(\bar{t}) \in D'\}$.

Definition 1.5. We define $\mathcal{C}_\mathcal{P} = \overline{\operatorname{Im}(\varphi)}$. The *radical variety* associated to the radical parametrization $\mathcal{P}$ is $\mathcal{V}_\mathcal{P} = \overline{\pi_\mathcal{P}(\mathcal{C}_\mathcal{P})}$ where $\pi_P(\overline{T}, \overline{\Delta}, \overline{X}, Z) = (\overline{X})$.

Remark 1.6. $\mathcal{V}_\mathcal{P}$ coincides with the intuitive $\overline{\operatorname{Im}\mathcal{P}}$, since they have the same topological dimension. In other words, $\mathcal{V}_\mathcal{P}$ does not depend on the construction above. One can recover the equations of $\mathcal{V}_\mathcal{P}$ from those of $\mathcal{B}_\mathcal{P}$ by eliminating all the variables except $\overline{X}$.

The next diagram shows the elements defined so far; $\mathcal{V}_\mathbb{T}$ and $\mathcal{R}$ are defined in Section 2.



Example 1.7. Here we show that $\mathcal{B}_\mathcal{P}$ and its projection on the $\overline{x}$ can be reducible. Consider $\mathcal{P} = (\sqrt[4]{t^2}, t)$ where $\sqrt[4]{1} = 1$. Take the tower $\mathbb{C}(t) \subset \mathbb{C}(\sqrt[4]{t^2})$ (an extension of degree 2), then $\mathcal{B}_\mathcal{P} = V(\Delta^4 - T^2, X - \Delta, Y - T, Z - 1)$, which has two components; they are projected on $V(X^4 - Y^2)$ which is the union of the parabolas $X^2 = \pm Y$. The image of $\mathcal{P}$ is half of the parabola $X^2 = Y$, and $\mathcal{V}_\mathcal{P}$ is the whole parabola. The "conjugate parametrizations" $(i^k \sqrt[4]{t^2}, t)$, $k = 1, 2, 3$, cover the other three half parabolas.

The main properties of the varieties defined above are summarized in the next theorem.

Theorem 1.8. *Let $\mathcal{P}$ be a radical parametrization.*

(1) *Every component of $\mathcal{B}_\mathcal{P}$ has dimension n, and the number of components is $\leq \prod e_i$.*
(2) *For every $\bar{t}_0$ in the domain of φ, the point $\varphi(\bar{t}_0) \in \mathcal{B}_\mathcal{P}$ is nonsingular.*
(3) *$\operatorname{Im}\varphi$ is contained in only one component of $\mathcal{B}_\mathcal{P}$, thus $\mathcal{C}_\mathcal{P}$ is irreducible.*
(4) *$\mathcal{V}_\mathcal{P}$ is irreducible and has dimension n.*

Sketch of proof. (1) By inspection of the defining equations of $\mathcal{B}_\mathcal{P}$, the fibres of $(\overline{T}, \overline{\Delta}, \overline{X}, Z) \mapsto (\overline{T})$ are finite and of cardinality $\leq \prod e_i$. (2) Apply Theorem 9 of [1] (p.492). (3) Every pair of points in the image can be connected by a path of regular points. But if two were in different algebraic components, any path must contain points in the intersection of the components, which are singular. (4) $\mathcal{V}_\mathcal{P}$ is the closure of the projection of $\mathcal{C}_\mathcal{P}$. □

2. Results

Definition 2.1. The tracing index of a parametrization $\mathcal{P}$ is defined as the degree of the map π_P, that is, the generic cardinal of $\pi_\mathcal{P}^{-1}(p)$ for $p \in \mathcal{V}_\mathcal{P}$.

Theorem 2.2. *If the tracing index of $\mathcal{P}$ is m, then there is a dense subset of $\operatorname{Im}\mathcal{P}$ where all the fibres have cardinal m.*

Example 2.3. Consider $\mathcal{P} = (t^2, \sqrt{t^2+1})$. Then $\mathcal{B}_\mathcal{P} = V(\Delta^2 - (T^2+1), X - T^2, Y - \Delta, Z - 1)$ (irreducible, thus equal to $\mathcal{C}_\mathcal{P}$) and $\mathcal{V}_\mathcal{P} = V(Y^2 - (X+1))$. Given a point $(a, b) \in \mathcal{V}_\mathcal{P}$, it has two preimages in $\mathcal{C}_\mathcal{P}$, given by $T = \pm\sqrt{a}$ (except for $a = 0$). Therefore the tracing index of $\mathcal{P}$ is two. On the other hand, $(1, \sqrt{2}) \in \mathcal{V}_\mathcal{P}$ has the parametrization preimages $t = \pm 1$, but $(1, -\sqrt{2}) \in \mathcal{V}_\mathcal{P}$ has no preimage (our chosen branch produces positive roots of positive real numbers). Indeed, only half of the points of $\mathcal{V}_\mathcal{P}$ are covered by $\mathcal{P}$.

Definition 2.4. We define the *tower variety* as $\mathcal{V}_\mathbb{T} = \overline{\operatorname{Im}\psi}$. The map $\mathcal{P}$ lifts to $\mathcal{R}\colon \mathcal{V}_\mathbb{T} \to \mathcal{V}_\mathcal{P}$.

$\mathcal{V}_\mathbb{T}$ has dimension n and its defining equations are the E_i in the definition of $\mathcal{B}_\mathcal{P}$. The importance of $\mathcal{V}_\mathbb{T}$ and $\mathcal{R}$ resides in the fact that they encode rationally the information of the radical parametrization. For example, for dimension one, the existence of $\mathcal{R}$ implies that $\operatorname{genus}(\mathcal{V}_\mathcal{P}) \leq \operatorname{genus}(\mathcal{V}_\mathbb{T})$. In particular, if $\mathcal{V}_\mathbb{T}$ is rational, then $\mathcal{V}_\mathcal{P}$ is rational. Note that this means the following: given a radical tower of fields whose associated $\mathcal{V}_\mathbb{T}$ has genus zero, any radical parametrization from that tower will give rise to a rational curve.

Theorem 2.5. *If the tracing index of $\mathcal{P}$ is 1, then $\mathcal{V}_\mathcal{P}$ and $\mathcal{V}_\mathbb{T}$ are birationally equivalent. Therefore, $\mathcal{V}_\mathcal{P}$ is rational if and only $\mathcal{V}_\mathbb{T}$ is rational.*

Next, we explore the algorithmic aspects of these ideas. In particular, we are interested in reparametrizing $\mathcal{P}$ rationally if possible. In the case of curves, from the previous theorem we

have that, if $\mathcal{P}$ is injective and $\mathcal{V}_\mathcal{P}$ is rational, $\mathcal{V}_\mathbb{T}$ is rational too; a rational parametrization of $\mathcal{V}_\mathbb{T}$ provides a rational parametrization of $\mathcal{V}_\mathcal{P}$. More generally:

Theorem 2.6. *If $\mathcal{V}_\mathbb{T}$ is unirational, then $\mathcal{V}_\mathcal{P}$ is unirational. Furthermore, if $\mathcal{T}(\bar{t})$ is a rational parametrization of $\mathcal{V}_\mathbb{T}$ then $\mathcal{P}(\pi_\mathbb{T} \circ \mathcal{T}(\bar{t}))$ is a rational parametrization of $\mathcal{V}_\mathcal{P}$ where $\pi_\mathbb{T}(\overline{T}, \overline{\Delta}) = (\overline{T})$.*

Assume that an algorithm that decides the existence of rational parametrizations and computes them is available; see for instance [5] for curves and [2] for surfaces. Call this algorithm RatParamAlg. Then, the last two theorems provide a reparametrization algorithm.

Reparametrization Algorithm
Input: A radical parametrization $\mathcal{P}$ given as in Def. 1.4
Output: One of: a reparametrization of $\mathcal{P}$ that makes it rational; or "$\mathcal{V}_\mathcal{P}$ cannot be parametrized rationally"; or "No answer".

(1) Compute $\mathcal{V}_\mathbb{T}$ and apply RatParamAlg to it.
(2) If $\mathcal{V}_\mathbb{T}$ has a rational parametrization $\mathcal{T}(\bar{t})$ RETURN $\pi_\mathbb{T}(\mathcal{T}(\bar{t}))$.
(3) If $\mathcal{V}_\mathbb{T}$ cannot be parametrized rationally, compute the tracing index m of $\mathcal{P}$. If $m = 1$, RETURN "$\mathcal{V}_\mathcal{P}$ cannot be parametrized rationally" ELSE RETURN "No answer".

Example 2.7. Let $\mathcal{P} = (\sqrt[3]{t}, \sqrt{1 - \sqrt[3]{t^2}})$. We have $\mathcal{V}_\mathbb{T} = V(\Delta_1^3 - t, \Delta_2^2 - (1 - \Delta_1^2))$, that can be parametrized by $\mathcal{T}(t) = \left(\left(\frac{2t}{1+t^2}\right)^3, \frac{2t}{1+t^2}, \frac{1-t^2}{1+t^2}\right)$. Then we obtain the rational reparametrization $\mathcal{P}\left(\left(\frac{2t}{1+t^2}\right)^3\right) = \left(\frac{2t}{1+t^2}, \frac{1-t^2}{1+t^2}\right)$.

This can be applied to the computation of certain radical integrals. For example, $\int R(\sqrt[3]{t}, \sqrt{1 - \sqrt[3]{t^2}})\, dt$, where R is a rational function, can always be converted by $u = \left(\frac{2t}{1+t^2}\right)^3$ into the integral of a rational function in u.

References

[1] Cox D., Little J., O'Shea D. (2006). *Ideals, varieties, and algorithms* (3rd Ed.). Series *Undergraduate Texts in Mathematics*. Springer, New York.
[2] Schicho J. (1998). *Rational parametrization of surfaces*. J. Symbolic Comput., 26(1), pp. 1–29.
[3] Sendra J.R., Sevilla D. (2011). *Radical parametrizations of algebraic curves by adjoint curves*. J. Symbolic Comput. 46(9), pp. 1030–1038.
[4] Sendra J.R., Sevilla D. (2013). *First steps towards radical parametrization of algebraic surfaces*. Comput. Aided Geom. Design 30(4), pp. 374–388.
[5] Sendra J.R., Winkler F., Pérez-Díaz S. (2008). *Rational algebraic curves*. Volume 22 of *Algorithms and Computation in Mathematics*. Springer, Berlin.
[6] Zariski O., Samuel P. (1965). *Commutative algebra*, Vol. 1. Springer-Verlag, New York-Heidelberg-Berlin.

Grupo ASYNACS. Dpto. de Física y Matemáticas, Universidad de Alcalá. Ap. Correos 20, E-28871 Alcalá de Henares, Madrid
E-mail address: {rafael.sendra, carlos.villarino}@uah.es

Dpto. de Matemáticas, C. U. de Mérida, Universidad de Extremadura. Av. Santa Teresa de Jornet 38, E-06800 Mérida, Badajoz
E-mail address: sevillad@unex.es


GOBIERNO
DE ESPAÑA
MINISTERIO
DE ECONOMÍA
Y COMPETITIVIDAD


Real Sociedad
Matemática Española


aertic
fer


FOUNDATION
COMPOSITIO
MATHEMATICA


UNIVERSIDAD
DE LA RIOJA